PART 2

ENDPOINT SECURITY

An *endpoint* is any hardware device connected to a network. This includes stationary devices (desktop computers and printers), mobile devices (laptops, smartphones, and tablets), and specialized hardware (Internet of Things (IoT) devices). The modules in this second part identify the threats and attacks directed at endpoints and the security that should be applied to these devices and their applications.

MODULE 3
THREATS AND ATTACKS ON ENDPOINTS

MODULE 4
ENDPOINT AND APPLICATION DEVELOPMENT SECURITY

MODULE 5

MOBILE, EMBEDDED, AND SPECIALIZED DEVICE SECURITY

MODULE 3

THREATS AND ATTACKS ON ENDPOINTS

After completing this module, you should be able to do the following:

1. Identify the different types of attacks using malware
2. Define application attacks
3. Explain how threat actors use application attacks
4. Define adversarial artificial intelligence attacks

Front-Page Cybersecurity

Despite the skyrocketing number of ransomware attacks in the past five years, ransomware has been used for more than 30 years. The first known ransomware attack was initiated in 1989 by an AIDS researcher. He carried out his attack by distributing 20,000 floppy disks to other AIDS researchers in more than 90 countries with an accompanying message that the disks contained a program that analyzed an individual's risk of acquiring AIDS. However, the disk also contained ransomware that remained dormant until the computer had been turned on 90 times. At the next startup, the ransomware displayed a message demanding a payment of $189 (plus $378 for use of the program).

Today, ransomware attacks are running rampant. There were an estimated 184 million ransomware attacks in 2018. In 2019, attackers using ransomware turned their sights on government agencies, educational institutions, and even healthcare providers. But these incidents were not just expensive inconveniences; the disruption they caused put health, safety, and even lives at risk. Ransomware attacks on healthcare providers resulted in emergency patients being redirected to other hospitals, inaccessible medical records, cancelled surgical procedures, postponed lab tests, and suspended hospital admissions. Emergency responder and law enforcement agencies were also common victims so that emergency 911 services were interrupted, police were unable to access details about criminal histories or active warrants, and jail doors could not be opened.[1]

What is fueling the rapid rise of ransomware? Is it a growing sophistication of attackers? Misguided users? More powerful ransomware? The answer, according to some, is none of these but something entirely unexpected.

The rapid rise of ransomware is attributed to cyber insurance.

Cyber insurance started 20 years ago by Lloyd's of London, and today it is an $8 billion industry. About 80 percent of Lloyd's cyber insurance is written for U.S. entities. A cyber insurance policy is not cheap. The city of Houston is now taking out three $10 million policies from three different insurance companies for a total of $30 million in coverage. For this, Houston will pay $471,400 annually in premiums. Another Texas city, Fort Worth, has only a $5 million policy, but it costs $99,570 in annual premiums.

How can the blame be placed on cyber insurance agencies for the rapid growth of ransomware?

Following a ransomware attack, enterprises and government agencies need to get back to normal as quickly as possible. Every minute they are locked out of their computers because of ransomware, they are losing money or putting individuals at risk. Cyber insurers claim that it makes more financial sense to pay the ransom and get the key to unlock encrypted files so that organizations can get back to normal quickly.

However, paying the ransom is actually an advantage to the cyber insurance agencies. By paying the ransom, cyber insurance agencies hold down their overall costs. They do not have to pay for lost revenue due to downtime brought on by ransomware or pay for third-party security consultants to aid in the data recovery. But when insurers reward attackers by paying the ransom, they might actually be encouraging more ransomware attacks because attackers know they will be paid. In addition, an increase in the number of ransomware attacks could frighten more businesses and government agencies into buying cyber insurance policies.

But doesn't paying the ransom cost the cyber insurance agencies money? No. Cyber insurance is a very lucrative business. The "loss ratio" is an industry standard for comparing premiums paid for insurance (what comes in) against insurance claims (what goes out). For all property and casualty insurance, like auto insurance and homeowner insurance, that loss ratio is about 62 percent (or for every dollar of premiums, about 62 cents are paid out in claims). However, for cyber insurance, it is only 35 percent, meaning that the cyber insurance agencies pay 35 cents in claims for each dollar of premiums.

Some researchers claim that cyber insurance is increasing number of ransomware attacks. In fact, the chief technology officer for a well-known antivirus company has said, "Cyber insurance is what's keeping ransomware alive today."[2]

Throughout the years, different words have been used to describe network-connected hardware devices. Thirty years ago, when the TCP/IP protocol was becoming popular, the word *host* referred to any communicating device on the network (networks were made up of hosts). Twenty years ago, as servers became more popular, the word *client* was used (clients made requests to servers).

Today, a different word is commonly used when referring to network-connected hardware devices: *endpoints*. This change reflects the fact that devices that are connected to a network today are far more than a computing device with a keyboard and monitor. Instead, devices ranging from mobile smartphones and tablets to wearable fitness trackers, industrial control system sensors, automotive telematics units, and even personal drones are all network-connected hardware devices. The word endpoint has become an accurate description of today's end-user technology devices.

This change in terminology also reflects the increased risks that have multiplied—exponentially—with the increase of these new devices. Instead of protecting hosts or clients located inside a network security perimeter, today each endpoint is a target for attackers to attempt to steal or manipulate their data. And because the endpoints are connected to the network, a vulnerability on an endpoint can result in an attack that penetrates the network and infects all other endpoints. In short, today *every endpoint is a potential entry point*.

This module examines threats and attacks on endpoints. It begins by looking at attacks using various types of malware and then looks at application attacks. It concludes by examining adversarial artificial intelligence attacks.

ATTACKS USING MALWARE

✓ CERTIFICATION

1.2 Given a scenario, analyze potential indicators to determine the type of attack.

In a legal setting, a "computer contaminant" is defined as any set of computer instructions that is designed to modify, damage, destroy, record, or transmit information within a computer, computer system, or network without the intent or permission of the owner of the information, computer system, or network.[3] This legal definition is the basis for the definition of the cybersecurity word **malware** (*mal*icious soft*ware*), which is software that enters a computer system without the user's knowledge or consent and then performs an unwanted and harmful action. Malware is most often used as the general term that refers to a wide variety of damaging software programs.

Malware is continually evolving to avoid detection by improved security measures. According to one report, the number of new malware releases every month exceeds 20 million, and the total malware in existence is approaching 900 million instances.[4] Yet no standard has been established for the classification of the different types of malware so that like malware can be grouped together for study.

One attempt at classifying the diverse types of malware can be to examine the primary action that the malware performs and then group those together with similar actions. These malware actions used for groupings are imprison, launch, snoop, deceive, and evade.

NOTE 1

The popular vulnerability feed Mitre Common Vulnerabilities and Exposures (CVE) assigns a CVE ID number, brief description, and any pertinent references but does not try to group common vulnerabilities together. Likewise, the National Vulnerability Database (NVD) does not attempt to classify vulnerabilities. Both CVE and NVD are covered in Module 2.

Imprison

A prison is a building or location housing individuals who have been deprived of their freedom. Some types of malware attempt to take away the freedom of the user to do whatever they want on their computer. The types of malware that imprisons are ransomware and cryptomalware.

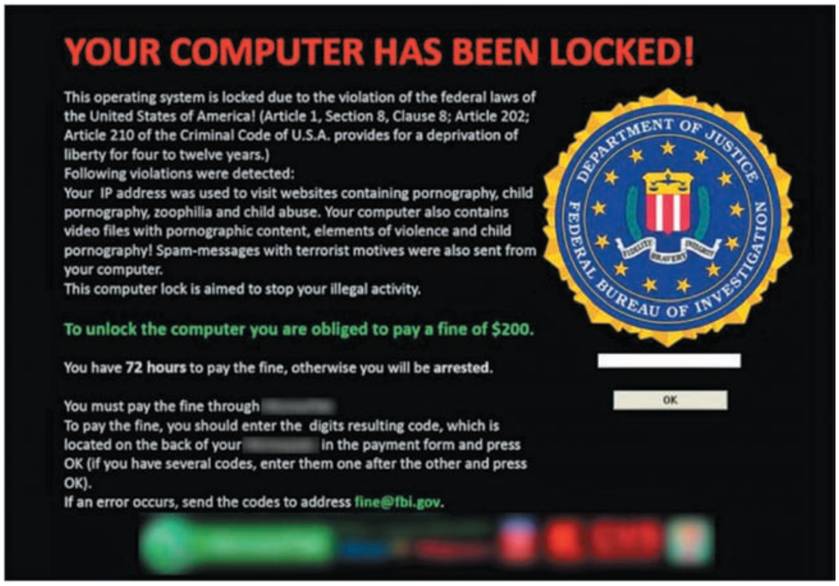
Ransomware

One of the fastest-growing types of malware is ransomware. **Ransomware** prevents a user's endpoint device from properly and fully functioning until a fee is paid; that is, it takes away a user's freedom from freely using their computer until the ransom is transacted. The ransomware embeds itself onto the computer in such a way that it cannot be bypassed, and even rebooting causes the ransomware to launch again.

NOTE 2

Some malware has more than one of these actions. However, in terms of classification, the *primary* action of the malware is used here.

Ransomware became widespread around 2010. This earliest ransomware displays a screen and prevents the user from accessing the computer's resources (called *blocker ransomware*). The screen contains instructions that pretends to be from a reputable third party, giving a "valid" reason for blocking the user's computer. One example is ransomware that purports to come from a law enforcement agency. This message, using official-looking imagery, states that the user performed an illegal action such as downloading pornography and must immediately pay a fine online by entering a credit card number. Figure 3-1 shows a blocker ransomware message.



YOUR COMPUTER HAS BEEN LOCKED!

This operating system is locked due to the violation of the federal laws of the United States of America! (Article 1, Section 8, Clause 8; Article 202; Article 210 of the Criminal Code of U.S.A. provides for a deprivation of liberty for four to twelve years.)
Following violations were detected:
Your IP address was used to visit websites containing pornography, child pornography, zoophilia and child abuse. Your computer also contains video files with pornographic content, elements of violence and child pornography! Spam-messages with terrorist motives were also sent from your computer.
This computer lock is aimed to stop your illegal activity.

To unlock the computer you are obliged to pay a fine of $200.

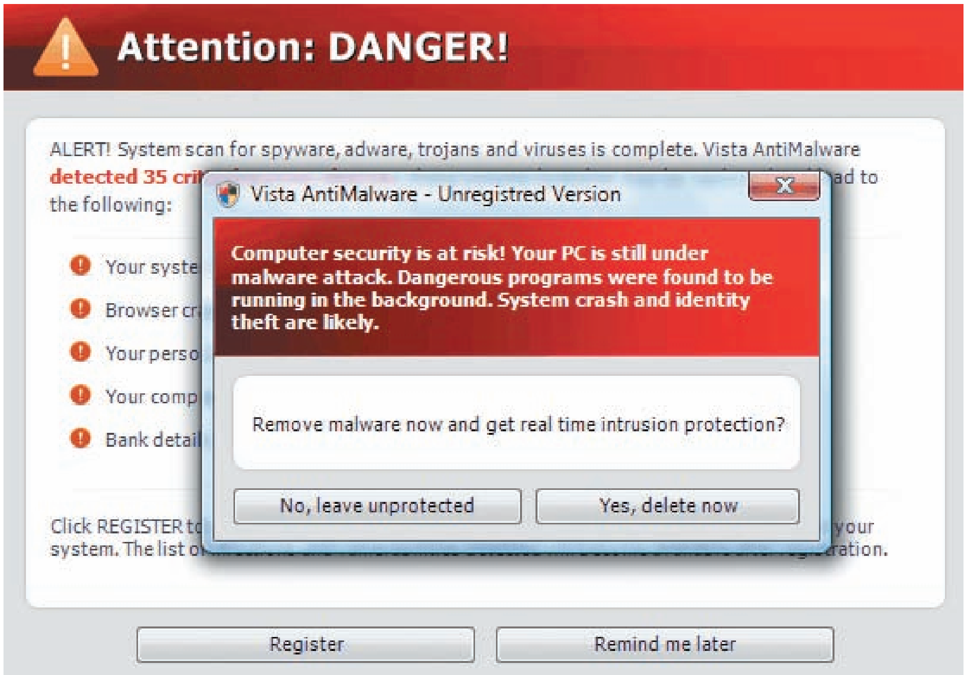
You have **72 hours** to pay the fine, otherwise you will be **arrested.**

You must pay the fine through [redacted]
To pay the fine, you should enter the digits resulting code, which is located on the back of your [redacted] in the payment form and press OK (if you have several codes, enter them one after the other and press OK).
If an error occurs, send the codes to address fine@fbi.gov.

Source: Symantec Security Response

Figure 3-1 Blocker ransomware message

Another variation of this type of ransomware pretends to come from a software vendor and displays a fictitious warning that a software license has expired or there is a problem with the computer such as imminent hard drive failure or—in a touch of irony—a malware infection. This ransomware variation tells users that they must immediately renew their license or purchase additional software online to fix a nonexistent problem. The ransomware example in Figure 3-2 uses color schemes and icons like those found on legitimate software.



Attention: DANGER!

ALERT! System scan for spyware, adware, trojans and viruses is complete. Vista AntiMalware detected 35 cri ad to the following:

- Your syste
- Browser cr
- Your perso
- Your comp
- Bank detail

Click REGISTER t your system. The list o ration.

Vista AntiMalware - Unregistred Version

Computer security is at risk! Your PC is still under malware attack. Dangerous programs were found to be running in the background. System crash and identity theft are likely.

Remove malware now and get real time intrusion protection?

Source: Used with permissions from Microsoft

Figure 3-2 Ransomware computer infection

NOTE 3

Researchers have developed an entirely new means of detecting a ransomware infection, and it works even if the type of ransomware is new and has never been seen before. When ransomware begins to encrypt files, the computer's temperature, power consumption, and voltage levels experience a surge. Sensors can detect these surges and immediately halt the encryption process, stopping ransomware dead in its tracks.

As ransomware became more widespread, the threat agents dropped the pretense that the ransomware was from a reputable third party. Instead, they simply blocked the user's computer and demanded a fee for its release. Ransomware attackers have determined what they consider the optimal price point for payment to unblock a computer: the amount must be small enough that most victims will begrudgingly pay to have their systems unblocked, but large enough that when thousands of victims pay up, the attackers can garner a handsome sum. For individuals, the ransom is usually around $500. However, for enterprises, the price has increased dramatically: the average ransom paid for one type of malware was more than $1.3 million, and the average ransom for all ransomware has increased by one-third.[5]

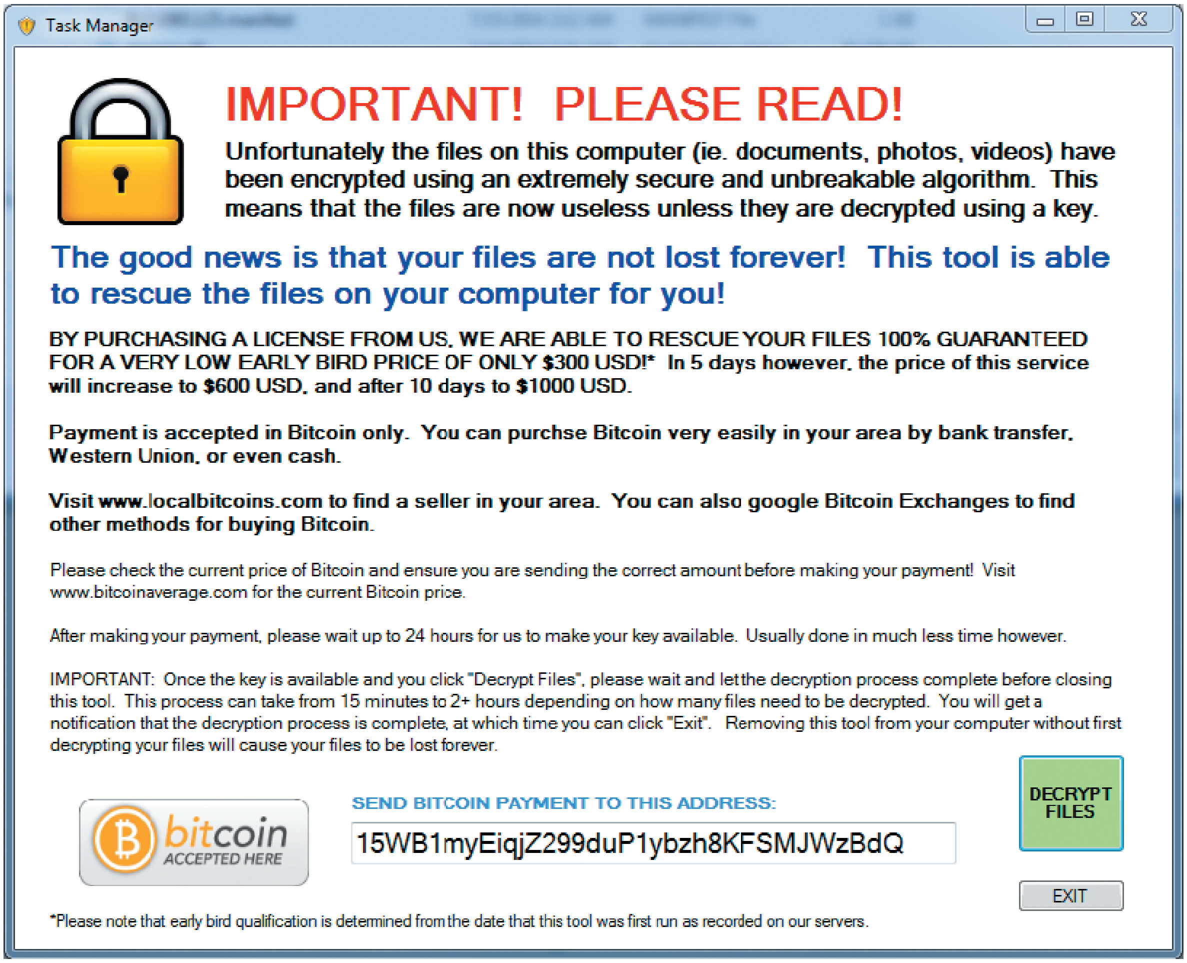
Ransomware continues to be a serious threat to users. Threat actors have now shifted their sights to state and local governments that typically have weaker security. In 2019, two-thirds of ransomware attacks targeted state and local governments;[6] to date, more than 350 of these governments have been the victims of successful attacks.[7]

NOTE 4

Originally, the FBI did not support paying a ransom in any circumstances. It said, "The FBI does not advocate paying a ransom, in part because it does not guarantee an organization will regain access to its data . . . Paying ransoms emboldens criminals to target other organizations and provides an alluring and lucrative enterprise to other criminals." But in late 2019, the FBI seemingly softened its stance by adding, "However, the FBI understands that when businesses are faced with an inability to function, executives will evaluate all options to protect their shareholders, employees, and customers."[8]

Cryptomalware

In recent years, a more malicious form of ransomware has arisen. Instead of just blocking users from accessing the computer, it encrypts all the files on the device so that none of them can be opened. This is called **cryptomalware**. A screen appears telling the victims that their files are now encrypted, and a fee must be paid to receive a key to unlock them. In addition, threat actors have increased the urgency for payment: the cost for the key to unlock the cryptomalware increases every few hours or days. On some occasions, the threat actors claim that a growing number of the encrypted user files will be deleted until the ransom is paid; if the ransom is not paid promptly, the key to unlock the files can never be purchased. Figure 3-3 shows a cryptomalware message.



IMPORTANT! PLEASE READ!

Unfortunately the files on this computer (ie. documents, photos, videos) have been encrypted using an extremely secure and unbreakable algorithm. This means that the files are now useless unless they are decrypted using a key.

The good news is that your files are not lost forever! This tool is able to rescue the files on your computer for you!

BY PURCHASING A LICENSE FROM US, WE ARE ABLE TO RESCUE YOUR FILES 100% GUARANTEED FOR A VERY LOW EARLY BIRD PRICE OF ONLY $300 USD!* In 5 days however, the price of this service will increase to $600 USD, and after 10 days to $1000 USD.

Payment is accepted in Bitcoin only. You can purchse Bitcoin very easily in your area by bank transfer, Western Union, or even cash.

Visit www.localbitcoins.com to find a seller in your area. You can also google Bitcoin Exchanges to find other methods for buying Bitcoin.

Please check the current price of Bitcoin and ensure you are sending the correct amount before making your payment! Visit www.bitcoinaverage.com for the current Bitcoin price.

After making your payment, please wait up to 24 hours for us to make your key available. Usually done in much less time however.

IMPORTANT: Once the key is available and you click "Decrypt Files", please wait and let the decryption process complete before closing this tool. This process can take from 15 minutes to 2+ hours depending on how many files need to be decrypted. You will get a notification that the decryption process is complete, at which time you can click "Exit". Removing this tool from your computer without first decrypting your files will cause your files to be lost forever.

SEND BITCOIN PAYMENT TO THIS ADDRESS:

15WB1myEiqjZ299duP1ybzh8KFSMJWzBdQ

*Please note that early bird qualification is determined from the date that this tool was first run as recorded on our servers.

Source: Bitcoin

Figure 3-3 Cryptomalware message

In addition to encrypting files on the user's local hard drive, new variants of cryptomalware encrypt all files on *any* network or attached device connected to that computer. This includes secondary hard disk drives, USB hard drives, network-attached storage devices, network servers, and even cloud-based data repositories. Thus, if a user's computer in an enterprise is infected with cryptomalware, potentially *all* files for the enterprise—and not just those on one computer—can be locked.

Launch

Another category of malware is that which infects a computer to launch attacks on other computers. This includes a virus, worm, and bot.

Virus

There are two types of viruses: a file-based virus and a fileless virus.

File-Based Virus A biological virus is composed of tiny bits of genetic material enclosed by a protective shell. By themselves, viruses are lifeless and inert as they wait for a favorable environment in which to reproduce. When a virus encounters a host cell, the virus attaches itself to the outer wall of the cell, enters inside, travels to the cell's genome, merges with its genes, and then tricks the host's genome into make copies of itself.

NOTE 5

With early cryptomalware attacks, threat actors only delivered the decryption key fewer than half of the times that a ransom was paid. However, this resulted in some victims not paying a ransom since the risk was high of not getting the key. Threat actors have since learned that there is more to gain in the long run of making the key available after a ransom is paid. Today, when victims pay the ransom, a decryption tool is delivered 99 percent of the time. However, the key only works about 96 percent of the time. This is because specific variants of ransomware have a tendency to corrupt data when it is encrypted.[9]

A *file-based virus* is remarkably similar to a biological virus. It is malicious computer code that is attached to a file. A very large number of file types can contain a virus, and Table 3-1 lists some of the 50 different Microsoft Windows file types that can be infected with a virus. Like its biological counterpart, a file-based virus reproduces itself on the same computer. Strictly speaking, a file-based virus replicates itself (or an evolved copy of itself) without any human intervention.

NOTE 6

When the host cell is infected by a virus, the virus takes over the operation of that cell, converting it into a virtual factory to make more copies of the virus. The host cell rapidly produces millions of identical copies of the original virus. Biologists often say that viruses exist only to make more viruses.

NOTE 7

One of the first viruses found on a microcomputer was written for the Apple II in 1982. Rich Skrenta, a ninth-grade student in Pittsburgh, wrote "Elk Cloner," which displayed his poem on the screen after every 50th use of the infected floppy disk. Unfortunately, the virus leaked out and found its way onto the computer used by Skrenta's math teacher. In 1984, the mathematician Dr. Frederick Cohen introduced the term *virus* based on a recommendation from his advisor, who came up with the name from reading science fiction novels.

Table 3-1 Windows file types that can be infected

File extension	Description
DOCX or XLSX	Microsoft Office user documents
EXE	Executable program file
MSI	Microsoft installer file
MSP	Windows installer patch file
SCR	Windows screen saver
CPL	Windows Control Panel file
MSC	Microsoft Management Console file
WSF	Windows script file
PS1	Windows PowerShell script

Early viruses were relatively straightforward in how they infected files. One basic type of infection is the *appender infection*. The virus first attaches or appends itself to the end of the infected file. It then inserts at the beginning of the file a *jump* instruction that points to the end of the file, which is the beginning of the virus code. When the program is launched, the jump instruction redirects control to the virus. Figure 3-4 shows how an appender infection works.

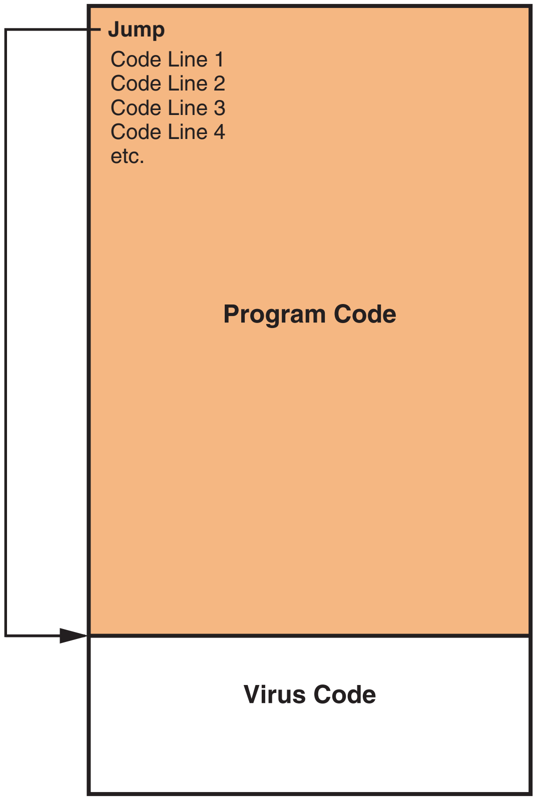


Figure 3-4 Appender infection

However, these types of viruses could be detected by virus scanners relatively easily. Later file-based viruses went to greater lengths to avoid detection; this type of virus is called an *armored file-based virus*. Some of the armored virus infection techniques include the *split infection* (it split the malicious code itself into several parts and then these parts are placed at random positions throughout the program code) and the *mutation* (the virus changes its internal code to one of a set number of predefined mutations whenever it is executed).

Each time the infected program is launched or the data file is opened—either by the user or the computer's operating system (OS)—the virus first unloads a payload to perform a malicious action (such as to corrupt or delete files, prevent programs from launching, steal data to be sent to another computer, cause a computer to crash repeatedly, or turn off the computer's security settings). Then the virus reproduces itself by inserting its code into another file, but only on the same computer. A virus can only replicate itself on the host computer where it is located; it cannot automatically spread to another computer by itself. Instead, it must rely on the actions of users to spread to other computers. Because viruses are attached to files, they are spread when a user transfers those files to other devices. For example, a user might send an infected file as an email attachment or copy an infected file to a USB flash drive and give the drive to another user. Once the virus reaches a new computer, it begins to infect it. Thus, a virus must have two carriers: a file to which it attaches and a human to transport it to other computers.

Fileless Virus A **fileless virus**, on the other hand, does not attach itself to a file. Instead, fileless viruses take advantage of native services and processes that are part of the OS to avoid detection and carry out its attacks. These native services used in a fileless virus are called *living-off-the-land binaries (LOLBins)*. For a computer running Microsoft Windows, some of commonly exploited LOLBins are listed in Table 3-2.

Table 3-2 Microsoft Windows common LOLBins

Name	Description
PowerShell	A cross-platform and open source task automation and configuration management framework
Windows Management Instrumentation (WMI)	A Microsoft standard for accessing management information about devices
.NET Framework	A free, cross-platform, open source developer platform for building different types of applications
Macro	A series of instructions that can be grouped together as a single command to automate a complex set of tasks or a repeated series of tasks, can be written by using a macro scripting language, such as Visual Basic for Applications (VBA), and is stored within the user document (such as in an Excel .xlsx workbook or Word .docx file)

Unlike a file-based virus, a fileless virus does not infect a file and wait for that file to be launched. Instead, the malicious code of a fileless virus is loaded directly in the computer's random access memory (RAM) through the LOLBins and then executed.

There are several advantages of a fileless virus over a file-based virus:

- *Easy to infect.* A fileless virus does not require that a specific type of file be stored on the computer's hard drive for the virus to infect. Instead, a common delivery method is through malicious webpages that the user visits. These pages silently send a script to the victim's web browser, which invokes a scripting language such as JavaScript. The browser passes instructions to a LOLBin such as PowerShell, which reads and executes the commands.

NOTE 8

Some armored viruses scan for the presence of files that security researchers typically use. If those files are present, the virus assumes it is being examined for weaknesses and automatically self-destructs by deleting itself.

NOTE 9

Several similarities between biological and computer viruses exist: both must enter their host passively (by relying on the action of an outside agent), both must be on the correct host (a horse virus cannot make a human sick, just as an Apple Mac virus cannot infect a Windows computer), both can only replicate when inside the host, both may remain dormant for a period of time, and both types of viruses replicate at the expense of the host.

NOTE 10

Microsoft Windows LOLBins are often categorized into binaries (programs that end in .EXE), libraries (.DLL), and scripts (.VBS). By some estimates 115 Windows LOLBins can be exploited by a fileless virus, while UNIX/Linux systems have 185 LOLBins.

- *Extensive control.* Several LOLBins have extensive control and authority on a computer. For example, PowerShell has full access to the core OS of a Windows computer, so it can undermine existing security features. PowerShell can also manipulate user accounts and password protection.
- *Persistent.* A program that is loaded into RAM for execution will terminate once the computer is shut down or rebooted. However, fileless viruses often write their script into the *Windows Registry*, which is a database that stores settings for the Windows OS and application programs. Each time the computer is restarted or on a set schedule, the script of the fileless virus is again launched.
- *Difficult to detect*. Files that are infected with a file-based virus can be scanned by an antivirus tool for detection. However, because a fileless virus loads into RAM, no telltale file can be scanned. Also, by using LOLBins, there is no evidence of other tools being used. And some LOLBins like PowerShell run in a section of system memory that cannot be queried or searched, making its activities virtually impossible to detect.
- *Difficult to defend against*. To fully defend against a fileless virus, it would be necessary to turn off all the potential LOLBins, which would cripple the OS and cause it to not properly function. Also, these LOLBins are loaded by default when the OS starts so that any attempt to turn selected LOLBins off would already be too late.

Worm

A second type of malware that has as its primary purpose to spread is a worm. A **worm** is a malicious program that uses a computer network to replicate. (Worms are sometimes called *network viruses*.) A worm is designed to enter a computer through the network and then take advantage of a vulnerability in an application or an OS on the host computer. Once the worm has exploited the vulnerability on one system, it immediately searches for another computer on the network that has the same vulnerability.

NOTE 11

One of the first wide-scale worm infections occurred in 1988. This worm exploited a misconfiguration in a program that allowed commands emailed to a remote system to be executed on that system. The worm also carried a payload containing a program that attempted to determine user passwords. Almost 6,000 computers, or 10 percent of the devices connected to the Internet at that time, were affected. The threat actor responsible was later convicted of federal crimes in connection with this incident.

Early worms were relatively benign and designed simply to spread quickly but not corrupt the systems they infected. These worms slowed down the network through which they were transmitted by replicating so quickly that they consumed all network resources. Today's worms can leave behind a payload on the systems they infect and cause harm, much like a virus. Actions that worms have performed include deleting files on the computer or allowing the computer to be remotely controlled by an attacker.

NOTE 12

Although viruses and worms are said to be automatically self-replicating, *where* they replicate is different. A virus self-replicates *on* the host computer but does not spread to other computers by itself. A worm self-replicates *between* computers (from one computer to another).

Bot

Another popular payload of malware is software that allows the infected computer to be placed under the remote control of an attacker for the purpose of launching attacks. This infected robot computer is known as a **bot** or *zombie*. When hundreds, thousands, or even millions of bot computers are gathered into a logical computer network, they create a *botnet* under the control of a *bot herder*.

NOTE 13

Due to the multitasking capabilities of modern computers, a computer can act as a bot while carrying out the tasks of its regular user. Users are completely unaware that their computer is being used for malicious activities.

Table 3-3 lists some of the attacks that can be generated through botnets.

Table 3-3 Uses of botnets

Type of attack	Description
Spamming	Botnets are widely recognized as the primary source of spam email. A botnet consisting of thousands of bots enables an attacker to send massive amounts of spam.
Spreading malware	Botnets can be used to spread malware and create new bots and botnets. Bots can download and execute a file sent by the attacker.
Ad fraud	Threat actors earn money by generating a high number of "clicks" on advertisements at targeted websites, using a bot to mimic the mouse clicks of a user.
Mining cryptocurrencies	Also called "cryptomining," this is a process in which transactions for various forms of cryptocurrency are verified, earning the "miner" a monetary reward. Botnets combine the resources of millions of bots for mining cryptocurrencies.

Infected bot computers receive instructions through a **command and control (C&C)** structure from the bot herders regarding which computers to attack and how. There are a variety of ways for this communication to occur, including the following:

- A bot can receive its instructions by automatically signing in to a bot-herding website where information has been placed that the bot knows how to interpret as commands.
- Bots can sign in to a third-party website; this has an advantage of the bot herder not needing to have a direct affiliation with that website.
- Commands can be sent via blogs, specially coded attack commands through posts on Twitter, or notes posted in Facebook.
- Bot herders are increasingly using a "dead drop" C&C mechanism by setting up a Google Gmail email account and then creating a draft email message that is never sent but contains commands the bot receives when it logs in to Gmail and reads the draft. Because the email message is never sent, there is no record of the commands. All Gmail transmissions are protected so that outsiders cannot view them.

Snoop

Another category of malware "snoops" or spies on its victims. The two common types of snooping malware are spyware and keyloggers.

Spyware

Spyware is tracking software that is deployed without the consent or control of the user. Spyware can secretly monitor users by collecting information without their approval through the computer's resources, including programs already installed on the computer, to collect and distribute personal or sensitive information. Table 3-4 lists different technologies used by spyware.

Table 3-4 Technologies used by spyware

Technology	Description	Impact
Automatic download software	Downloads and installs software without the user's interaction	Could install unauthorized applications
Passive tracking technologies	Gathers information about user activities without installing any software	Could collect private information such as websites a user has visited
System modifying software	Modifies or changes user configurations, such as the web browser home page or search page, default media player, or lower-level system functions	Changes configurations to settings that the user did not approve
Tracking software	Monitors user behavior or gathers information about the user, sometimes including personally identifiable or other sensitive information	Could collect personal information that can be shared widely or stolen, resulting in fraud or identity theft

CAUTION Not all spyware is necessarily malicious. For example, spyware monitoring tools can help parents keep track of the online activities of their children.

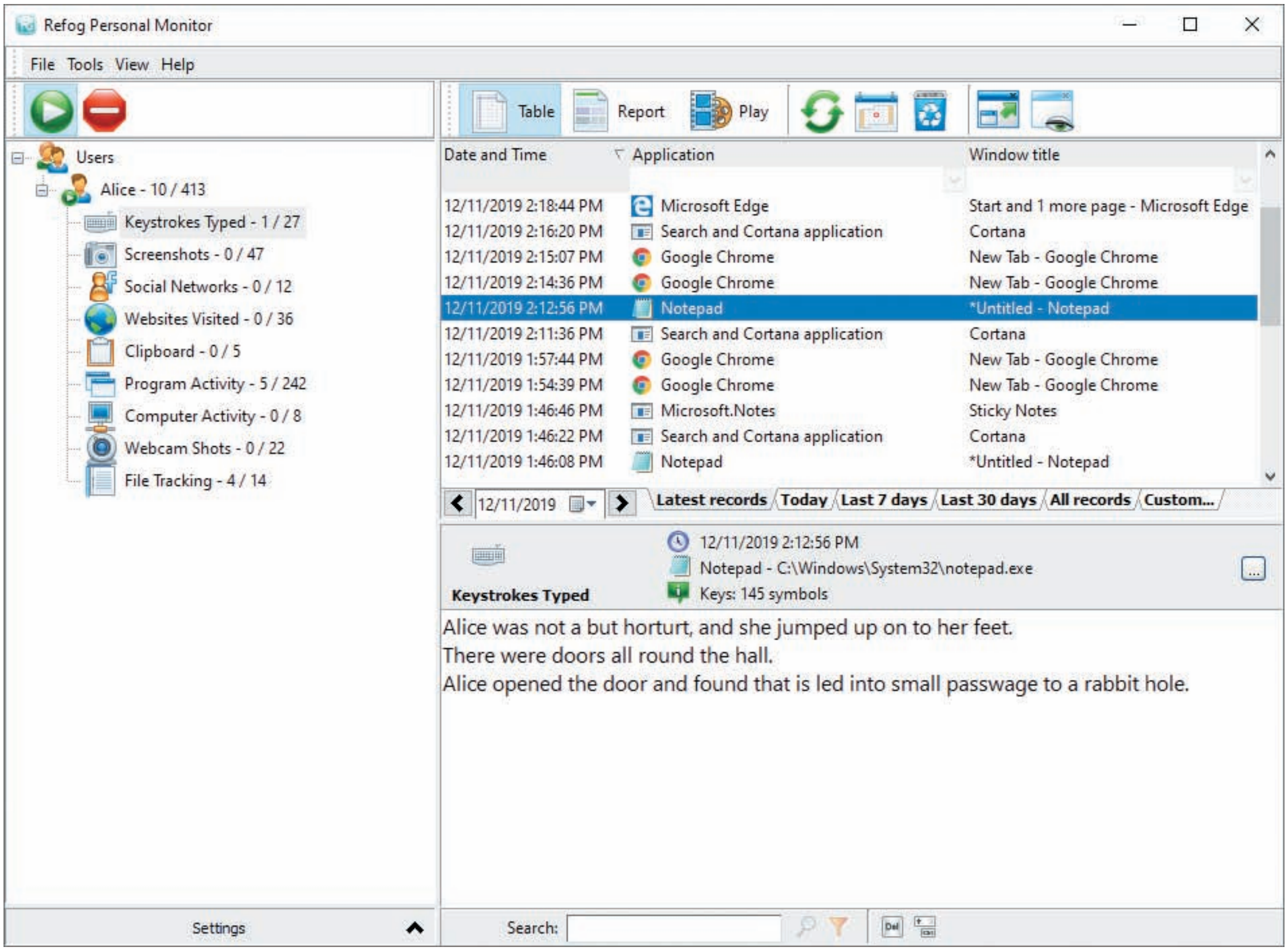
Keylogger

Another type of spying is done with a **keylogger** that silently captures and stores each keystroke that a user types on the computer's keyboard. The threat actor can then search the captured text for any useful information such as passwords, credit card numbers, or personal information. A keylogger can be a software program or a small hardware device.

Software keyloggers are programs installed on the computer that silently capture sensitive information. However, software keyloggers, which conceal themselves so that the user cannot detect them, go far beyond capturing a user's keystrokes. These programs can also capture everything on the user's screen and silently turn on the computer's web camera to record images of the user. A software keylogger is illustrated in Figure 3-5.

NOTE 14

An advantage of software keyloggers is that they do not require physical access to the user's computer because they can be installed remotely. They can routinely send captured information back to the attacker through the victim's own Internet connection.



Date and Time	Application	Window title
12/11/2019 2:18:44 PM	Microsoft Edge	Start and 1 more page - Microsoft Edge
12/11/2019 2:16:20 PM	Search and Cortana application	Cortana
12/11/2019 2:15:07 PM	Google Chrome	New Tab - Google Chrome
12/11/2019 2:14:36 PM	Google Chrome	New Tab - Google Chrome
12/11/2019 2:12:56 PM	Notepad	*Untitled - Notepad
12/11/2019 2:11:36 PM	Search and Cortana application	Cortana
12/11/2019 1:57:44 PM	Google Chrome	New Tab - Google Chrome
12/11/2019 1:54:39 PM	Google Chrome	New Tab - Google Chrome
12/11/2019 1:46:46 PM	Microsoft.Notes	Sticky Notes
12/11/2019 1:46:22 PM	Search and Cortana application	Cortana
12/11/2019 1:46:08 PM	Notepad	*Untitled - Notepad

12/11/2019 2:12:56 PM
Notepad - C:\Windows\System32\notepad.exe
Keys: 145 symbols

Alice was not a but horturt, and she jumped up on to her feet.
There were doors all round the hall.
Alice opened the door and found that is led into small passwage to a rabbit hole.

Refog

Figure 3-5 Software keylogger

For computers that are in a public location such as a library or computer lab but are "locked down" so that no software can be installed, a hardware keylogger can be used instead. These keyloggers are hardware devices inserted between the computer keyboard connection and USB port, as shown in Figure 3-6. Because the device resembles an ordinary keyboard plug and the computer keyboard USB port is often on the back of the computer, a hardware keylogger can easily go undetected. In addition, the device is beyond the reach of the computer's antimalware scanning software and thus raises no alarms. A disadvantage of a hardware keylogger is that the threat actor must install and then later return to physically remove the device in order to access the information it has stored, each time being careful not to be detected.

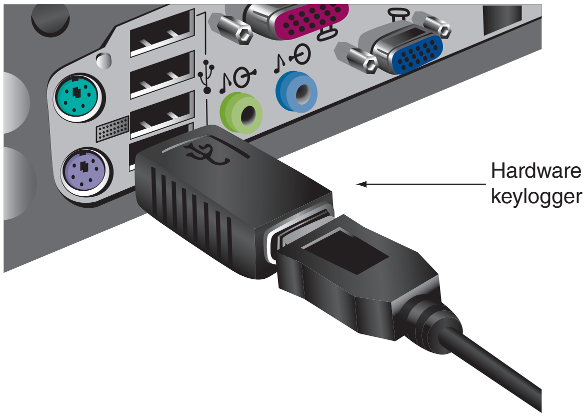


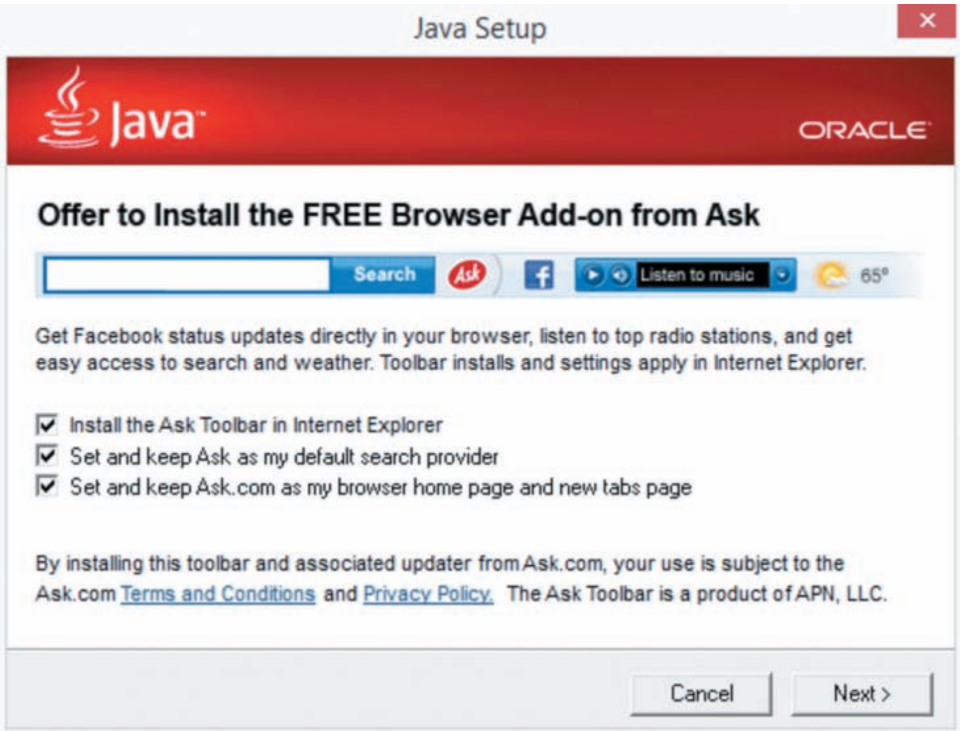
Figure 3-6 Hardware keylogger

Deceive

Some malware attempts to deceive the user and hide its true intentions. Software in this category includes potentially unwanted programs (PUPs), Trojans, and remote access Trojans (RATs).

Potentially Unwanted Program (PUP)

A broad category of software that is often more annoying than malicious is called **potentially unwanted programs (PUPs)**. A PUP is software that the user does not want on their computer. PUPs often become installed along with other programs and are the result of the user overlooking the default installation options on software downloads, as seen in Figure 3-7.



Offer to Install the FREE Browser Add-on from Ask

Get Facebook status updates directly in your browser, listen to top radio stations, and get easy access to search and weather. Toolbar installs and settings apply in Internet Explorer.

☑ Install the Ask Toolbar in Internet Explorer
☑ Set and keep Ask as my default search provider
☑ Set and keep Ask.com as my browser home page and new tabs page

By installing this toolbar and associated updater from Ask.com, your use is subject to the Ask.com Terms and Conditions and Privacy Policy. The Ask Toolbar is a product of APN, LLC.

Source: Oracle Corporation

Figure 3-7 Default installation options

PUPs may include software that is pre-installed on a new computer or smartphone and cannot be easily removed (if at all). Other examples of PUPs are advertising that obstructs content or interferes with web browsing, pop-up windows, pop-under windows, search engine hijacking, home page hijacking, toolbars with no value for the user, and settings that redirect to competitors' websites, alter search results, and replace ads on webpages.

NOTE 15

The term PUP was created by an Internet security company because marketing firms objected to having their products being called "spyware."

Trojan

According to ancient legend, the Greeks won the Trojan War by hiding soldiers in a large hollow wooden horse that was presented as a gift to the city of Troy. Once the horse was wheeled into the fortified city, the soldiers crept out of the horse during the night and attacked the unsuspecting defenders.

A computer **Trojan** is an executable program that masquerades as performing a benign activity but also does something malicious. For example, a user might download what is advertised as a calendar program, yet in addition to installing the calendar, it also installs malware that scans the system for credit card numbers and passwords, connects through the network to a remote system, and then transmits that information to the attacker.

Remote Access Trojan (RAT)

A special type of Trojan is a **remote access Trojan (RAT)**. A RAT has the basic functionality of a Trojan but also gives the threat agent unauthorized remote access to the victim's computer by using specially configured communication protocols. This creates an opening into the victim's computer, allowing the threat agent unrestricted access. The attacker can not only monitor what the user is doing but also change computer settings, browse and copy files, and even use the computer to access other computers connected on the network.

Evade

The final category of malware attempts to help malware or attacks evade detection. This includes backdoor, logic bomb, and rootkit.

Backdoor

A **backdoor** gives access to a computer, program, or service that circumvents any normal security protections. Backdoors installed on a computer allow the attacker to return later and bypass security settings.

Creating a legitimate backdoor is a common practice by developers, who may need to access a program or device on a regular basis, yet do not want to be hindered by continual requests for passwords or other security approvals. The intent is for the backdoor to be removed once the application is finalized. However, in some instances, backdoors have been left installed, and attackers have used them to bypass security.

Logic Bomb

A **logic bomb** is computer code that is typically added to a legitimate program but lies dormant and evades detection until a specific logical event triggers it. Once it is triggered, the program then deletes data or performs other malicious activities.

Logic bombs are difficult to detect before they are triggered. This is because logic bombs are often embedded in very large computer programs, some containing hundreds of thousands of lines of code, and a trusted employee can easily insert a few lines of computer code into a long program without anyone detecting it. In addition, these programs are not routinely scanned for containing malicious actions.

NOTE 16

Many logic bombs have been planted by disgruntled employees. For example, a Maryland government employee tried to destroy the contents of more than 4,000 servers by planting a logic bomb script that was scheduled to activate 90 days after he was terminated.

Rootkits

A **rootkit** is malware that can hide its presence and the presence of other malware on the computer. It does this by accessing "lower layers" of the operating system or even using undocumented functions to make alterations. This enables the rootkit and any accompanying software to become undetectable by the operating system and common antimalware scanning software that is designed to seek and find malware.

NOTE 17

The risks of rootkits are significantly diminished today due to protections built into operating systems. These protections include preventing unauthorized kernel drivers from loading, stopping modifications to certain kernel areas used by rootkits to hide, and preventing rootkits from modifying the bootloader program.

TWO RIGHTS & A WRONG

1. It is a common tactic for cryptomalware attackers to not send the decryption key after the ransom has been paid.
2. Fileless viruses take advantage of native services and processes that are part of the operating system (OS) to avoid detection and carry out its attacks, and these native services used in a fileless virus are called living-off-the-land binaries (LOLBins).
3. A remote access Trojan (RAT) can monitor what the user is doing, change computer settings, browse and copy files, and use the computer to access other computers connected on the network.

See Appendix B for the answer.

APPLICATION ATTACKS

CERTIFICATION

1.3 Given a scenario, analyze potential indicators associated with application attacks.

Attacks using malware typically add malicious software to an endpoint. Another category of attacks specifically targets software applications that are already installed and running on the device. These attacks look for vulnerabilities in the application or manipulate the application in order to compromise it. While on occasion threat actors target applications running on a user's endpoint like a personal computer, more often, their sights are set on compromising applications that can provide many more potential victims than a single computer user. The more common targets of attackers using application attacks are Internet web servers.

A web server provides services that are implemented as "web applications" through software applications running on the server. A typical web application infrastructure is shown in Figure 3-8. The client's web browser makes a request using the Hypertext Transport Protocol (HTTP) to a web server, which may be connected to one or more web application servers. These application servers run the specific "web apps," which in turn are directly connected to database servers on the internal network. Information from these database servers is retrieved and returned to the web server so that the information can be sent back to the user's web browser.

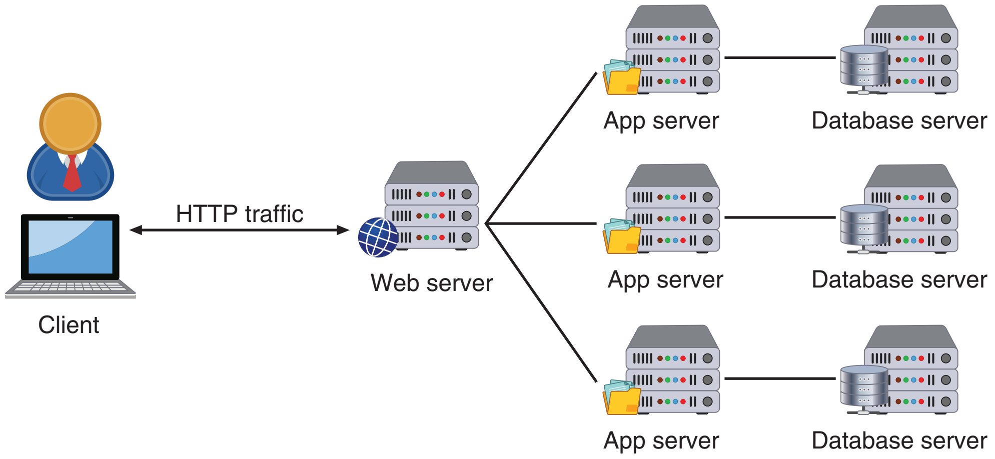


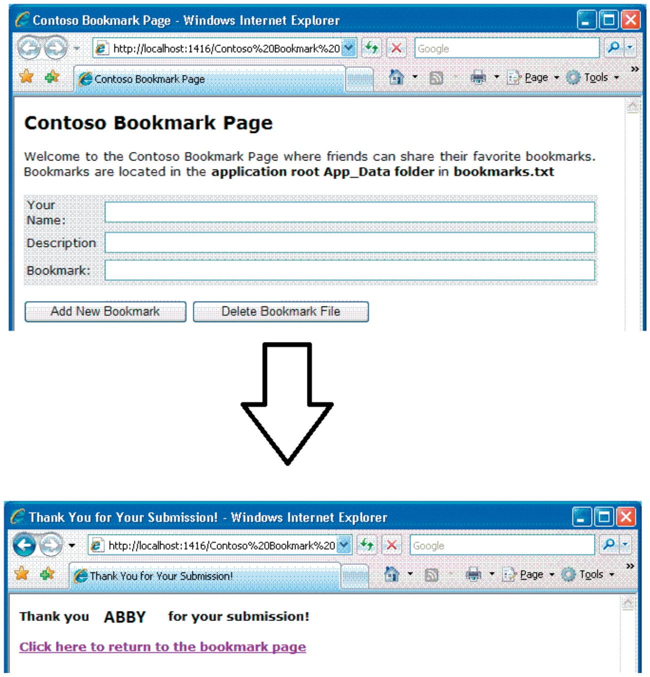
Figure 3-8 Web server application infrastructure

The multiple elements in a web application infrastructure provide multiple attack points: a single vulnerability could expose many other users who are accessing the web server. An attack could also compromise backend databases and app servers, and the connected network infrastructure.

Application attacks include scripting attacks, injection attacks, request forgery attacks, and replay attacks. These attacks typically target how the applications function. In addition, attacks directly focused on vulnerabilities in the software applications are common.

Scripting

Most web applications create dynamic content based on input from the user. Figure 3-9 illustrates a fictitious web application that allows friends to share their favorite bookmarks with each other online. Users can enter their name, a description, and the URL of the bookmark and then receive a personalized “Thank You” screen. In Figure 3-10, the code that generates the “Thank You” screen is illustrated.


Contoso Bookmark Page - Windows Internet Explorer
http://localhost:1416/Contoso%20Bookmark%20
Google
Contoso Bookmark Page
Page
Tools
Add New Bookmark
Delete Bookmark File
Thank You for Your Submission! - Windows Internet Explorer
http://localhost:1416/Contoso%20Bookmark%20
Google
Thank You for Your Submission!
Page
Tools

Contoso Bookmark Page

Welcome to the Contoso Bookmark Page where friends can share their favorite bookmarks. Bookmarks are located in the **application root App_Data folder** in **bookmarks.txt**

Your Name:

Description

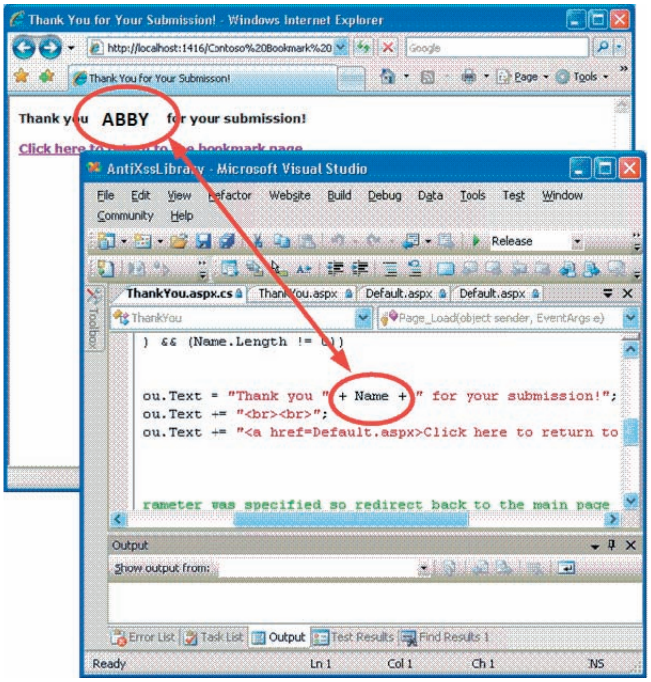
Bookmark:

Thank you ABBY for your submission!

Click here to return to the bookmark page

Source: Used with permissions from Microsoft

Figure 3-9 Bookmark page that accepts user input



Thank you ABBY for your submission!

Click here to return to the bookmark page

```
) && (Name.Length != 0))


ou.Text = "Thank you " + Name + " for your submission!";
ou.Text += "<br><br>";
ou.Text += "<a href=Default.aspx>Click here to return to


rameter was specified so redirect back to the main page
```

Source: Used with permissions from Microsoft

Figure 3-10 Input used in response

NOTE 18

The term *cross-site scripting* refers to an attack using scripting that originates on one site (the web server) to impact another site (the user’s computer).

In a **cross-site scripting (XSS)** attack, a website that accepts user input without validating it (called *sanitizing*) and uses that input in a response can be exploited. In the previous example, the input that the user enters for *Name* is not verified but instead is automatically added to a code segment that becomes part of an automated response. An attacker can take advantage of this in an XSS attack by tricking a valid website into feeding a malicious script to another user’s web browser, which will then execute it.

Injection

In addition to cross-site attacks on web server applications, attacks called **injections** also introduce new input to exploit a vulnerability. One of the most common injection attacks, called **SQL injection**, inserts statements to manipulate a database server. SQL stands for **Structured Query Language**, a language used to view and manipulate data that is stored in a relational database. SQL injection targets SQL servers by introducing malicious commands into them.

Consider a webpage that offers a solution for users who have forgotten their password. An online form asks users to enter their username, which is also their email address that is already on file. The submitted email address is compared to the stored email address, and if they match, a reset URL is emailed to that address.

If the email address entered by the user into the form is stored in the variable *$EMAIL*, then the underlying SQL statement to retrieve the stored email address from the database would be similar to this:

```
SELECT fieldlist FROM table WHERE field = '$EMAIL'
```

The *WHERE* clause is meant to limit the database query to only display information when the condition is considered true (that is, when the email address in *$EMAIL* matches an address in the database).

An attacker using an SQL injection attack would begin by first entering a fictitious email address on this webpage that included a single quotation mark as part of the data, such as *braden.thomas@fakemail.com'*. If the message *E-mail Address Unknown* is displayed, it indicates that user input is being properly filtered and an SQL attack cannot be rendered on the site. However, if the error message *Server Failure* is displayed, it means that the user input is not being filtered and all user input is sent directly to the database. This is because the *Server Failure* message is due to a syntax error created by the additional single quotation mark in the fictitious email address.

Armed with the knowledge that input is sent unfiltered to the database, the attacker knows that anything he enters as a username in the form would be sent to and then processed by the SQL database. Now, instead of entering a user name, the attacker would enter this command, which would let him view all the email addresses in the database: *whatever' or 'a'='a*. This command is stored in the variable *$EMAIL*. The expanded SQL statement would read

```
SELECT fieldlist FROM table WHERE field = 'whatever' or 'a'='a'
```

These values are the following:

- *'whatever'*. This can be anything meaningless.
- *or*. The SQL *or* means that as long as either of the conditions are true, the entire statement is true and will be executed.
- *'a'='a'*. This is a statement that will always be true.

Because *'a'='a'* is always true, the *WHERE* clause is also true. It is not limited as it was when searching for a single email address before it would become true. The result can be that *all* user email addresses will then be displayed.

By entering crafted SQL statements as user input, information from the database can be extracted or the existing data can be manipulated. SQL injection statements that can be entered and stored in *$EMAIL*, and their pending results are shown in Table 3-5.

Table 3-5 SQL injection statements

SQL injection statement	Result
'whatever' AND email IS NULL;	Determine the names of different fields in the database
'whatever' AND 1= (SELECT COUNT(*) FROM tabname);	Discover the name of the table
'whatever' OR full name LIKE '%Mia%';	Find specific users
'whatever'; DROP TABLE members;	Erase the database table
'whatever'; UPDATE members SET email = ' attacker-email@evil.net*' WHERE email = '* Mia@good.com*';*	Mail password to attacker's email account

In addition to using SQL to view and manipulate data that is stored in a relational database, other types of databases not using SQL (called *NoSQL databases*) are also used. One popular type of NoSQL database manipulates data using the **eXtensible Markup Language (XML)**. Like the markup language Hyper Text Markup Language (HTML) used for webpages, XML is not a processing language but instead is designed to store information. A NoSQL database that uses XML for data manipulation is also subject to an injection attack like SQL injection if the input is not sanitized. This is called an **XML injection**.

Request Forgery

Although some attacks have confusing names, that is not the case with the category of *request forgery*. As its name suggests, it is a request that has been fabricated. There are two types of request forgeries. These are a cross-site request forgery (CSRF) and a server-site request forgery (SSRF).

Cross-Site Request Forgery (CSRF)

A **cross-site request forgery (CSRF)** takes advantage of an authentication "token" that a website sends to a user's web browser. If a user is currently authenticated on a website and is then tricked into loading another webpage, the new page inherits the identity and privileges of the victim, who may then perform an undesired function on the attacker's behalf. Figure 3-11 illustrates a cross-site request forgery. Because a CSRF takes place on the client site, it is sometimes called a **client-side request forgery**.

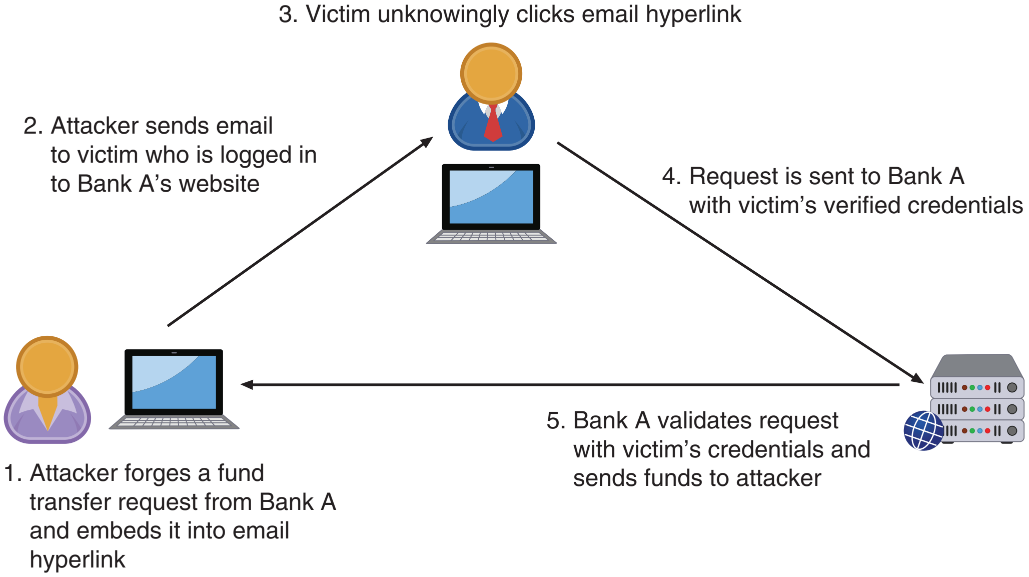

3. Victim unknowingly clicks email hyperlink
2. Attacker sends email to victim who is logged in to Bank A's website
4. Request is sent to Bank A with victim's verified credentials
5. Bank A validates request with victim's credentials and sends funds to attacker
1. Attacker forges a fund transfer request from Bank A and embeds it into email hyperlink

Figure 3-11 Cross-site request forgery

NOTE 19

In other words, in a CSRF attack a *request* to a website is not from the authentic user but is a *forgery* that involves *crossing sites.*

Server-Side Request Forgery (SSRF)

A **server-side request forgery (SSRF)** takes advantage of a trusting relationship between web servers (as opposed to a CSRF, which manipulates the trust from a user's browser to a server). SSRF attacks exploit how a web server processes external information received from another server. Some web applications are designed to read information from or write information to a specific URL. If an attacker can modify that target URL, they can potentially extract sensitive information from the application or inject untrusted input into it. Table 3-6 outlines the differences between a CSRF and a SSRF.

Table 3-6 CSRF and SSRF differences

Attack name	Attack target	Purpose of attack
CSRF	User	Force target to take action for attacker while pretending to be authorized user
SSRF	Web server	Gain access to sensitive data or inject harmful data

Replay

Whereas some attacks try to capture data sent between two users, a **replay** attack copies data and then uses it for an attack. Replay attacks are commonly used against digital identities—after intercepting and copying data, the threat actor retransmits selected and edited portions of the copied communications later to impersonate the legitimate user. Many digital identity replay attacks are between a user and an authentication server.

Attacks on Software

Other attacks are directly focused on vulnerabilities in the software applications. These include exploiting memory vulnerabilities, improper exception and error handling, and external software components.

Memory Vulnerabilities

Several attacks are directed at vulnerabilities associated with how a program uses RAM. These are often the result of poor techniques (or laziness) by the software developer.

Some memory-related attacks are called **resource exhaustion attacks** because they "deplete" parts of memory and thus interfere with the normal operation of the program in RAM. This may allow the threat actor access to the underlying OS in a way that it could be exploited by bypass security settings. An example is a **memory leak**. An application normally dynamically allocates memory, but due to a programming error, it may not free that memory when finished using it. An attacker can then take advantage of the unexpected program behavior resulting from a low memory condition.

Other memory-related attacks attempt to manipulate memory contents. Again, these are made possible by poor programming practices. These types of attacks include buffer overflow attacks and integer overflow attacks.

Buffer Overflow Consider a teacher working in his office who manually grades a lengthy written examination by marking incorrect answers with a red pen. Because he is frequently interrupted in his grading by students, the teacher places a ruler on the test question he is currently grading to indicate his "return point," or the point at which he should resume the grading. Suppose that two devious students enter his office as he is grading examinations. While one student distracts him, the second student silently slides the ruler down from question 4 to question 20. When the teacher returns to grading, he will resume at the wrong "return point" and not look at the answers for questions 4 through 19.

This scenario is similar to how a buffer overflow attacker attempts to compromise a computer. A storage buffer on a computer typically contains the memory location of the software program that was being executed when another function interrupted the process; that is, the storage buffer contains the "return address" where the computer's processor should resume once the new process has finished. Attackers can substitute their own "return address" in order to point to a different area in the computer's memory that contains their malware code.

A **buffer overflow attack** occurs when a process attempts to store data in RAM beyond the boundaries of a fixed-length storage buffer. This extra data overflows into the adjacent memory locations (a *buffer overflow*). Because the storage buffer typically contains the "return address" memory location of the software program being executed when another function interrupted the process, an attacker can overflow the buffer with a new address pointing to the attacker's malware code. A buffer overflow attack is shown in Figure 3-12.

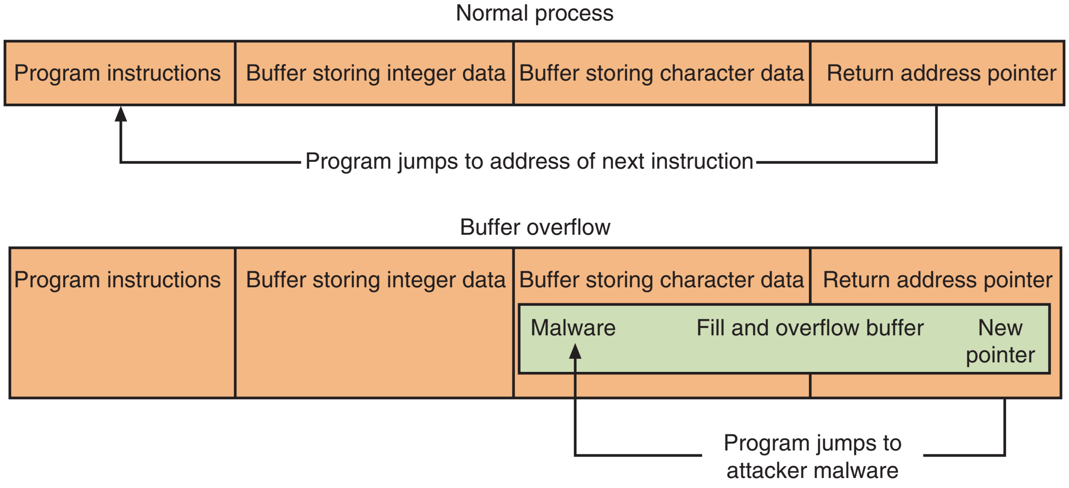

Normal process
Program instructions
Buffer storing integer data
Buffer storing character data
Return address pointer
Program jumps to address of next instruction
Buffer overflow
Program instructions
Buffer storing integer data
Buffer storing character data
Return address pointer
Malware
Fill and overflow buffer
New pointer
Program jumps to attacker malware

Figure 3-12 Buffer overflow attack

NOTE 20

The "return address" is not the only element that can be altered in a buffer overflow attack, but it is one of the most commonly altered elements.

Integer Overflow Consider a digital clock that can display the hours only as *1* to *12*. What happens when the time moves past *12:59*? The clock then "wraps around" to the lowest hour value of *1* again.

On a computer, an *integer overflow* is the condition that occurs when the result of an arithmetic operation—such as addition or multiplication—exceeds the maximum size of the integer type used to store it. When this integer overflow occurs, the interpreted value then wraps around from the maximum value to the minimum value. For example, an eight-bit signed integer has a maximum value of 127 and a minimum value of −128. If the value 127 is stored in a variable and 1 is added to it, the sum exceeds the maximum value for this integer type and wraps around to become −128.

In an **integer overflow attack**, an attacker changes the value of a variable to something outside the range that the programmer had intended by using an integer overflow. This type of attack could be used in the following situations:

- An attacker could use an integer overflow attack to create a buffer overflow. If an integer overflow could be introduced during the calculations for the length of a buffer when a copy is occurring, it could result in a buffer that is too small to hold the data. An attacker could then use this to create a buffer overflow attack.
- A program that calculates the total cost of items purchased would use the number of units sold times the cost per unit. If an integer overflow were introduced when tallying the number of items sold, it could result in a negative value and a resulting negative total cost, indicating that a refund is due the customer.
- A large positive value in a bank transfer could be wrapped around by an integer overflow attack to become a negative value, which could then reverse the flow of money: instead of adding this amount to the victim's account, it could withdraw that amount and later transfer it to the attacker's account.

NOTE 21

An extreme example of an integer overflow attack would be withdrawing $1 from an account that has a balance of 0, which could cause a new balance of $4,294,967,295!

Improper Exception Handling

Other attacks on software, like memory vulnerabilities, are the result of poor coding on the part of software developers. This is commonly the case when the program does not properly check for exceptions that may occur when the program is running.

Software that allows the user to enter data but has **improper input handling** features does not filter or validate user input to prevent a malicious action. For example, a webpage on a web server with improper input handling that asks for the user's email address could allow an attacker to instead enter a direct command that the server would then execute.

Other software may not properly trap an error condition and thus provides an attacker with underlying access to the system. This is known as incorrect **error handling**. Suppose an attacker enters a string of characters that is much longer than expected. Because the software has not been designed for this event, the program could crash or suddenly halt its execution and then display an underlying OS prompt, giving an attacker access to the computer.

Another improper exception handling situation is a NULL **pointer/object dereference**. (A *dereference* obtains from a pointer the address of a data item held in another location.) When an application dereferences a pointer that it expects to be valid but instead has a value of NULL, it typically will cause a program to crash or exit. A NULL pointer/object dereference can occur through a number of flaws, including simple programming omissions.

A NULL pointer/object dereference can also be the result of a race condition. A **race condition** in software occurs when two concurrent threads of execution access a shared resource simultaneously, resulting in unintended consequences. For example, in a program with two threads that have access to the same location in memory, Thread #1 stores the value *A* in that memory location. But since Thread #2 is also executing, it may overwrite the same memory location with the value *Z*. When Thread #1 retrieves the value stored, it is given Thread #2's *Z* instead of its own *A*. The software checks the state of a resource before using that resource, but the resource's state can change between the check and the use in a way that invalidates the results of the check. This is called a **time of check/time of use** race condition. This condition is often security-relevant: a threat actor who can influence the state of the resource between check and use can negatively impact a number of shared resources such as files, memory, or variables in multithreaded programs.

CAUTION Time of check/time of use often appears as time of check to time of use (TOCTTOU). A typo of TOCTTOU is TOCCTOU and has been used in some influential documents, so the typo is repeated fairly frequently.

Attacks on External Software Components

In addition to attacking the software directly, threat actors also target external software components. These include the following:

- *Application program interface (API)*. An *application program interface (API)* is a link provided by an OS, web browser, or other platform that allows a developer access to resources at a high level. An example of an API is when a user visits a website and the message "This site wants to know your location" appears. The website is attempting to the use geolocation API available in the web browser. APIs relieve the developer from the need to write code for specific hardware and software. Because APIs provide direct access to data and an entry point to an application's functions, they are attractive targets for attackers looking for vulnerabilities in the API in an **application program interface (API) attack**.
- *Device driver*. A *device driver* is software that controls and operates an external hardware device that is connected to a computer. Device drivers are specific to both the OS and the hardware device. Threat actors may attempt to alter a device driver for use in an attack (called **device driver manipulation**). An attacker may use **shimming**, or transparently adding a small coding library that intercepts calls made by the device and changes the parameters passed between the device and the device driver. This **refactoring** (changing the design of existing code) can be difficult to detect yet serves as a real threat.
- *Dynamic-link library (DLL)*. A *dynamic-link library (DLL)* is a repository of both code and data that can be used by more than one program at the same time. For example, in the Windows operating systems, the Comdlg32.DLL performs common dialog box related functions. Attackers use a technique called **DLL injection** for inserting code into a running process through a DLL to cause a program to function in a different way than intended.

NOTE 22

API vulnerabilities are particularly attractive because they can have a broad impact and may take a long time to discover. In 2018, Facebook found a vulnerability in its API code that made it possible for attackers to steal access tokens and take over the accounts of 30 million users. It took Facebook 14 months before they discovered the API vulnerability. It is predicted that by 2022, API abuses will become the most common type of web application attack resulting in a data breach.[10]

TWO RIGHTS & A WRONG

1. In an XSS attack, a website that accepts user input without sanitizing it and uses that input in a response can be exploited.
2. An SSRF takes advantage of a trusting relationship between a web browser and web servers.
3. A time of check/time of use is a vulnerability that causes a race condition.

See Appendix B for the answer.

ADVERSARIAL ARTIFICIAL INTELLIGENCE ATTACKS

CERTIFICATION

1.2 Given a scenario, analyze potential indicators to determine the type of attack.

Artificial intelligence is being used worldwide in a wide variety of applications, ranging from the mundane to the very sophisticated. Cybersecurity is likewise using these innovative technologies to enhance the detection of malicious behavior and advanced threats. However, there are significant vulnerabilities and risks with using these new tools. Understanding them includes knowing what the tools are and what they can do, how these tools are used in cybersecurity, and their potential risks.

What Are Artificial Intelligence (AI) and Machine Learning (ML)?

Consider the following scenario. Junaid's team leader asks him to fly to another branch of the company tomorrow morning to assist with its new direct marketing campaign. He agrees and books online an airplane reservation for early the next morning. As Junaid awakens, his smartphone buzzes. His phone alerts him about the weather at his destination and makes recommendations about the clothes to pack for his trip. It also tells him what time to leave for the airport based on local traffic and the length of time needed to pass through airport security. When Junaid pulls his car out of his driveway, he receives another message on his phone explaining that an accident is slowing traffic on the main road to the airport and directing him to take an alternate route. After his flight, Junaid arrives at the hotel and suddenly remembers that he volunteered to help plan a birthday party for his niece on Saturday. Using the smart speaker in his hotel room, he logs in to his account and tells the voice assistant what he needs. After a few minutes, he has arranged for the party invitations to be sent to select individuals in his contact list, ordered a birthday present based on what is popular with other children the same age as his niece, and has set a reminder for him to pick up the cake after he arrives back home. The next morning, Junaid arrives at the remote office to help with a direct marketing campaign. He begins by demonstrating to a college intern how to use the company's new smart assistant to segment their customers into groups to receive targeted messaging in order to increase the response rates.

Just a few years ago, this scenario would have been nothing more than science fiction. Today, however, it is commonplace and occurs multiple times every day in our work and personal lives. This is based on tools that provide genuine human-to-machine interaction.

The foundation behind this interaction is called *artificial intelligence (AI)*. Although definitions of AI vary, at its core, AI may be defined as technology that imitates human abilities. Although the practical use of AI has only appeared recently, it has a long history dating back to the first large-scale computers.

A recognized subset of AI is *machine learning (ML)*. Humans learn by direct commands of someone older and wiser, but this requires the other person to always be present. Humans also learn through experiences (such as touching a hot stove results in a painful burn). ML is defined as "teaching" a technology device to "learn" by itself without the continual instructions of a computer programmer. ML also involves learning through repeated experience: that is, if something attempted does not work, then it determines how it could be changed to make it work.

NOTE 23

The original goal of AI was to make computers more useful and more capable of independent reasoning. Most historians trace the birth of AI to a Dartmouth research project in 1956 that explored problem solving and symbolic methods. In the 1960s, the U.S. DoD became interested in this research and worked on training computers to imitate human reasoning. Some projects that came from the DoD were a street-mapping project in the 1970s and intelligent personal assistants in the early 2000s.

NOTE 24

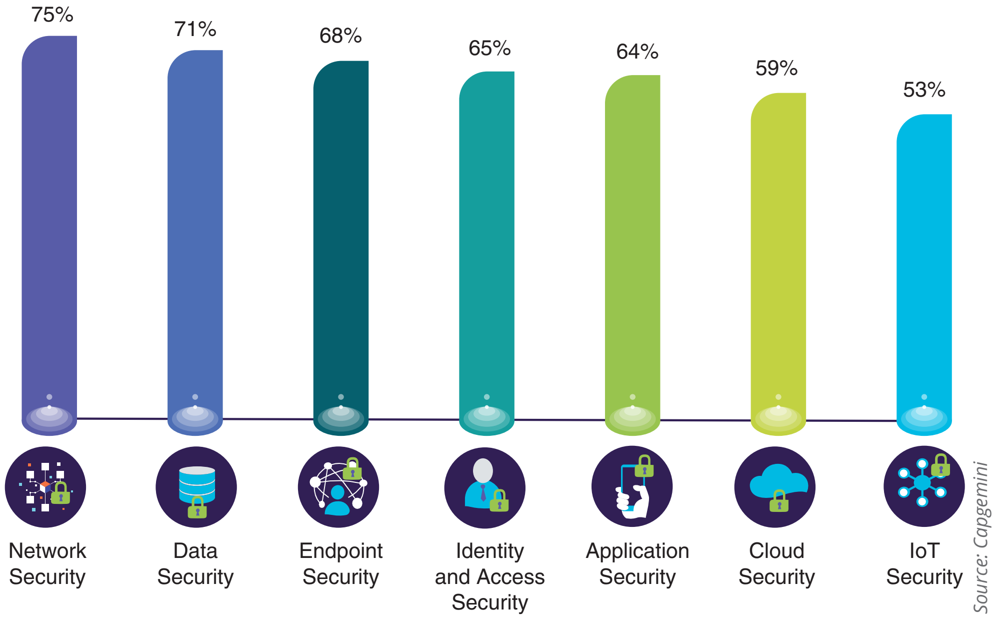
The relationship between AI and ML is AI applies ML to solve problems without being explicitly programmed what to do.

Uses in Cybersecurity

Cybersecurity AI allows organizations to detect, predict, and respond to cyberthreats in real time using ML. AI is already being used broadly in cybersecurity defenses. Virtually all email systems use some type of AI to block phishing attacks by examining obvious clues (such as the URL of the link that the victim is being tempted to click) but also subtle clues (such as the tense and voice of words in the email). AI using ML can analyze these to continually learn to distinguish and block phishing emails while allowing genuine emails to reach the user's inbox.

The prime advantages of using AI to combat threats are continual learning and greater speed in response. By relying on data from previous similar attacks, AI can predict and prevent future attacks. ML learning algorithms can quickly apply complex pattern recognition techniques to spot and thwart attacks much faster than humans can.

The use of AI in cybersecurity is widespread. About one in five organizations used cybersecurity AI before 2019, increasing to two out of three organizations planning to deploy it by the end of 2020. Telecommunications providers use cybersecurity AI more than any other sector: 80 percent of telecom companies said that they would not be able to respond to cyberattacks without using AI.[11] Figure 3-13 illustrates where AI cybersecurity is used in specific areas within an enterprise.



Source: Capgemini

Figure 3-13 How AI cybersecurity is used

Risks in Using AI and ML in Cybersecurity

Although the use of AI in cybersecurity is growing, there are risks associated with using AI and ML in cybersecurity. This is called **adversarial artificial intelligence**. The first risk is the **security of the ML algorithms**. Just as all hardware and software is subject to being infiltrated by threat actors, AI-powered cybersecurity applications and their devices likewise have vulnerabilities. These could be attacked and compromised, allowing threat actors to alter algorithms to ignore attacks, much like a rootkit can instruct an OS to ignore malicious actions.

Another risk is **tainted training data for machine learning**. Attackers can attempt to alter the training data that is used by ML in order to produce false negatives to cloak themselves.

CAUTION Another concern is that threat actors themselves will turn to using AI for attacks in order to circumvent defenses.

TWO RIGHTS & A WRONG

1. Artificial intelligence (AI) may be defined as technology that imitates human abilities.
2. AI is already being used broadly in cybersecurity defenses.
3. A recognized subset of ML is AI.

See Appendix B for the answer.

VM LAB You're now ready to complete the live, virtual machine labs for this module. The labs can be found each module in the MindTap.

SUMMARY

- The word “endpoint” is commonly used when referring to network-connected hardware devices. Devices that are connected to a network today include traditional desktop computers, mobile smartphones, tablets, wearable fitness trackers, and even personal drones. The word endpoint reflects the risks that have increased with using these devices. Instead of protecting devices located inside a network security perimeter, today each endpoint is a target for attackers to attempt to steal or manipulate their data. Every endpoint is a potential entry point for attackers.
- Malware (malicious software) is software that enters a computer system without the user’s knowledge or consent and then performs an unwanted and harmful action. There has been no standard established for the classification of the different types of malware so that like malware can be grouped together for study. One attempt at classifying the diverse types of malware can be to examine the primary action that the malware performs and then group those together with similar actions.
- Some types of malware attempt to take away the freedom of users to do whatever they want on their computer. These types of malware “imprison” the user. One of the fastest-growing types of malware is ransomware, which takes away users’ freedom from using their computer until the ransom is transacted. Early ransomware pretended to be from a reputable third-party, giving a “valid” reason for blocking the user’s computer. As ransomware became more widespread, the threat agents dropped the pretense that the ransomware was from a reputable third party. Instead, they simply blocked the user’s computer and demanded a fee for its release. In recent years, a more malicious form of ransomware has arisen. Instead of just blocking the user from accessing the computer, it encrypts all the files on the device so that none of them can be opened. This is called cryptomalware.
- Another category of malware infects a computer to then launch attacks on other computers. A file-based virus is remarkably similar to a biological virus. It is malicious computer code that is attached to a file. Like its biological counterpart, a file-based virus reproduces itself on the same computer. Early viruses were relatively straightforward in how they infected files. Later file-based viruses went to greater lengths to avoid detection; this type of virus is called an armored file-based virus. A fileless virus does not attach itself to a file but takes advantage of native services and processes that are part of the OS to avoid detection and carry out its attacks. These native services used in a fileless virus are called living-off-the-land binaries (LOLBins). There are several advantages of a fileless virus over a file-based virus. These include ease of infection, extensive control, persistence, difficulty in detection, and difficulty in defense. A worm is a malicious program that uses a computer network to replicate. It is designed to enter a computer through the network and then take advantage of a vulnerability in an application or an OS on the host computer. An infected computer that is under the remote control of an attacker for the purpose of launching attacks is called a bot or zombie. When large numbers of bot computers are gathered into a logical computer network, they create a botnet. Infected bot computers receive instructions through a command and control (C&C) structure.
- Another category of malware “snoops” or spies on its victims. Spyware is tracking software that is deployed without the consent or control of the user. Spyware typically secretly monitors users by collecting information without their approval using the computer’s resources, including programs already installed on the computer, to collect and distribute personal or sensitive information. A keylogger silently captures and stores each keystroke that a user types on the computer’s keyboard so that the attacker can later search the captured text for any useful information. A keylogger can be a software program or a small hardware device.
- Some malware attempts to deceive the user and hide its true intentions. A broad category of software that is often more annoying than malicious is called potentially unwanted programs (PUPs), or software that the user does not want on their computer. PUPs often become installed along with other programs when the user overlooks the default installation options on software downloads. A computer Trojan is an executable program that masquerades as performing a benign activity but also does something malicious. A special type of Trojan is a remote access Trojan (RAT). A RAT has the basic functionality of a Trojan but also gives the threat agent unauthorized remote access to the victim’s computer by using specially configured communication protocols.
- Some malware attempts to evade detection. A backdoor gives access to a computer, program, or service that circumvents any normal security protections. When installed on a computer, a backdoor allows the attacker to return later and bypass security settings. A logic bomb is computer code that is typically added to a legitimate program but lies dormant and evades detection until a specific logical event triggers it. Once it is triggered, the program then deletes data or performs other malicious activities. Logic bombs are difficult to detect before

they are triggered because they are often embedded in very large computer programs. A rootkit is malware that can hide its presence and the presence of other malware on the computer. It does this by accessing "lower layers" of the operating system or even using undocumented functions to make alterations.

- Another category of attacks specifically targets software applications that are already installed and running on the device. These attacks look for vulnerabilities in the application or manipulate the application in order to compromise it. In a cross-site scripting (XSS) attack, a website that accepts user input without validating (sanitizing) it and uses that input in a response can be exploited. Another type of attack called injections introduces new input to exploit a vulnerability. One of the most common injection attacks, called SQL injection, inserts statements to manipulate a database server. By entering crafted SQL statements as user input, information from the database can be extracted or the existing data can be manipulated. In addition to using SQL, other types of databases are used, such as those containing the eXtensible Markup Language (XML), which can also be used for data manipulation similar to an SQL injection. This is called an XML injection.
- A cross-site request forgery (CSRF) takes advantage of an authentication "token" that a website sends to a user's web browser. If a user is currently authenticated on a website and is then tricked into loading another webpage, the new page inherits the identity and privileges of the victim, who may then perform an undesired function on the attacker's behalf. A server-side request forgery (SSRF) takes advantage of a trusting relationship between web servers. SSRF attacks exploit how a web server processes external information received from another server. A replay attack copies data and then uses it for an attack.
- Several attacks are directed at vulnerabilities associated with how a program uses RAM. Some of these memory-related attacks are called resource exhaustion attacks because they "deplete" parts of memory and thus interfere with the normal operation of the program in RAM. This may allow the threat actor access to the underlying OS in a way that it could be exploited by bypassing security settings. A memory leak occurs when an application, instead of normally dynamically allocating memory, does not free that memory when finished using it. An attacker can then take advantage of the unexpected program behavior resulting from a low memory condition. A buffer overflow attack occurs when a process attempts to store data in RAM beyond the boundaries of a fixed-length storage buffer, and this extra data overflows into the adjacent memory locations. An attacker can overflow the buffer with a new address pointing to the attacker's malware code. An integer overflow attack occurs when an attacker changes the value of a variable to something outside the range that the programmer had intended by using an integer overflow.
- Software that allows the user to enter data but has improper input handling features does not filter or validate user input to prevent a malicious action. Other software may not properly trap an error condition and thus provides an attacker with underlying access to the system. This is known as incorrect error handling. When an application dereferences a pointer that it expects to be valid but instead has a value of NULL, it typically will cause a program to crash or exit. A NULL pointer/object dereference can occur through a number of flaws, including simple programming omissions. It can also be the result of a race condition, in which two concurrent threads of execution access a shared resource simultaneously.
- In an application program interface (API) attack, a threat actor looks for vulnerabilities in the API, a link provided by an OS, web browser, or other platform that allows a developer access to resources at a high level. A device driver is software that controls and operates an external hardware device that is connected to a computer and is specific to both the OS and the hardware device. Threat actors may attempt to alter a device driver for use in an attack (called device driver manipulation). A dynamic link library (DLL) is a repository of both code and data that can be used by more than one program at the same time. Attackers use a technique called DLL injection for inserting code into a running process through a DLL to cause a program to function in a way other than intended.
- Artificial intelligence (AI) is technology that imitates human abilities. Although the practical use of AI has only appeared recently, it has a long history dating back to the first large-scale computers. A recognized subset of AI is machine learning (ML). ML is defined as "teaching" a technology device to "learn" by itself without the continual instructions of a computer programmer and also to learn through repeated experience. Cybersecurity AI allows organizations to detect, predict, and respond to cyberthreats in real time using ML. There are risks associated with using AI and ML in cybersecurity called adversarial artificial intelligence. One risk is the security of the ML algorithms: just as all hardware and software is subject to being infiltrated by threat actors, AI-powered cybersecurity applications and their devices likewise have vulnerabilities. Another risk is tainted training data for machine learning. Attackers can attempt to alter the training data that is used by ML in order to produce false negatives to cloak themselves.

Key Terms

adversarial artificial intelligence
application program interface (API) attack
backdoor
bot
buffer overflow attack
client-side request forgery
command and control (C&C)
cross-site request forgery (CSRF)
cross-site scripting (XSS)
cryptomalware
device driver manipulation
DLL injection
error handling
eXtensible Markup Language (XML)
fileless virus
improper input handling
injections
integer overflow attack
keylogger
logic bomb
malware
memory leak
pointer/object dereference
potentially unwanted programs (PUPs)
race condition
ransomware
refactoring
remote access Trojan
replay
resource exhaustion attacks
rootkit
security of the ML algorithms
server-side request forgery (SSRF)
shimming
spyware
SQL injection
Structured Query Language
tainted training data for machine learning
time of check/time of use
Trojan
worm
XML injection

Review Questions

1. What word is the currently accepted term to refer to network-connected hardware devices?
  a. Host
  b. Endpoint
  c. Device
  d. Client

2. Which of the following is NOT a characteristic of malware?
  a. Deceive
  b. Launch
  c. Imprison
  d. Diffusion

3. Gabriel's sister called him about a message that suddenly appeared on her screen that says her software license has expired and she must immediately pay $500 to have it renewed before control of the computer will be returned to her. What type of malware has infected her computer?
  a. Persistent lockware
  b. Blocking ransomware
  c. Cryptomalware
  d. Impede-ware

4. Marius's team leader has just texted him that an employee, who violated company policy by bringing in a file on her USB flash drive, has just reported that her computer is suddenly locked up with cryptomalware. Why would Marius consider this a dangerous situation?
  a. It sets a precedent by encouraging other employees to violate company policy.
  b. Cryptomalware can encrypt all files on any network that is connected to the employee's computer.
  c. The organization may be forced to pay up to $500 for the ransom.
  d. The employee would have to wait at least an hour before her computer could be restored.

5. Which type of malware relies on LOLBins?
  a. PUP
  b. File-based virus
  c. Fileless virus
  d. Bot

6. Which of the following is known as a network virus?
  a. TAR
  b. Worm
  c. Remote exploitation virus (REV)
  d. C&C

7. Josh is researching the different types of attacks that can be generated through a botnet. Which of the following would NOT be something distributed by a botnet?
  a. LOLBins
  b. Spam
  c. Malware
  d. Ad fraud

8. Which of the following is NOT a means by which a bot communicates with a C&C device?
  a. Signing in to a website the bot herder operates
  b. Signing in to a third-party website
  c. Email
  d. Command sent through Twitter posts

9. Randall's roommate is complaining to him about all of the software that came pre-installed on his new computer. He doesn't want the software because it slows down the computer. What type of software is this?

- **a.** Spyware
- **b.** BOT
- **c.** PUP
- **d.** Keylogger

10. What is the difference between a Trojan and a RAT?

- **a.** There is no difference.
- **b.** A RAT gives the attacker unauthorized remote access to the victim's computer.
- **c.** A Trojan can carry malware while a RAT cannot.
- **d.** A RAT can infect only a smartphone and not a computer.

11. Which of these would NOT be considered the result of a logic bomb?

- **a.** Send an email to Rowan's inbox each Monday morning with the agenda of that week's department meeting.
- **b.** If the company's stock price drops below $50, then credit Oscar's retirement account with one additional year of retirement credit.
- **c.** Erase the hard drives of all the servers 90 days after Alfredo's name is removed from the list of current employees.
- **d.** Delete all human resource records regarding Augustine one month after he leaves the company.

12. Which of the following attacks is based on a website accepting user input without sanitizing it?

- **a.** RSS
- **b.** XSS
- **c.** SQLS
- **d.** SSXRS

13. Which of the following attacks is based on the principle that when a user is currently authenticated on a website and then loads another webpage, the new page inherits the identity and privileges of the first website?

- **a.** SSFR
- **b.** DLLS
- **c.** CSRF
- **d.** DRCR

14. Which of the following manipulates the trusting relationship between web servers?

- **a.** SSRF
- **b.** CSRF
- **c.** EXMAL
- **d.** SCSI

15. Which type of memory vulnerability attack manipulates the "return address" of the memory location of a software program?

- **a.** Shim overflow attack
- **b.** Factor overflow attack
- **c.** Integer overflow attack
- **d.** Buffer overflow attack

16. What race condition can result in a NULL pointer/object dereference?

- **a.** Conflict race condition
- **b.** Value-based race condition
- **c.** Thread race condition
- **d.** Time of check/time of use race condition

17. Which of the following attacks targets the external software component that is a repository of both code and data?

- **a.** Application program interface (API) attack
- **b.** Device driver manipulation attack
- **c.** Dynamic-link library (DLL) injection attack
- **d.** OS REG attack

18. What term refers to changing the design of existing code?

- **a.** Library manipulation
- **b.** Shimming
- **c.** Refactoring
- **d.** Design driver manipulation

19. Which of the following is technology that imitates human abilities?

- **a.** AI
- **b.** ML
- **c.** RC
- **d.** XLS

20. Which statement regarding a keylogger is NOT true?

- **a.** Software keyloggers can be designed to send captured information automatically back to the attacker through the Internet.
- **b.** Hardware keyloggers are installed between the keyboard connector and computer keyboard USB port.
- **c.** Software keyloggers are generally easy to detect.
- **d.** Keyloggers can be used to capture passwords, credit card numbers, or personal information.

Hands-On Projects

Project 3-1: Analyze File and URL for File-Based Viruses Using VirusTotal—Part 1

Time Required: 25 minutes

Objective: Given a scenario, analyze potential indicators to determine the type of attack.

Description: VirusTotal is a free online service that analyzes files and URLs to identify potential malware. VirusTotal combines 70 antivirus scanners and URL/domain blacklisting services along with other tools to identify malware. A wide range of files can be submitted to VirusTotal for examination, such as user data files and documents, executable programs, PDFs, and images. One of the uses of VirusTotal is to provide a "second opinion" on a file or URL that may have been flagged as suspicious by other scanning software. In this project, you use VirusTotal to scan a file and a URL.

1. First view several viruses from 20 years ago and observe their benign but annoying impact. Open your web browser and enter the URL **archive.org/details/malwaremuseum&tab=collection** (if you are no longer able to access the site through the web address, use a search engine to search for "Malware Museum").
2. All of the viruses have been rendered ineffective and will not harm a computer. Click several of the viruses and notice what they do.
3. When finished, close your web browser.
4. Use Microsoft Word to create a document that contains the preceding paragraph description about VirusTotal. Save the document as **VirusTotal.docx**.
5. Exit Word.
6. Open your web browser and enter the URL **www.virustotal.com** (if you are no longer able to access the site through the web address, use a search engine to search for "Virus Total").
7. If necessary, click the **File** tab.
8. Click **Choose File**.
9. Navigate to the location of **VirusTotal.docx** and click **Open**.
10. Click **Confirm upload**.
11. Wait until the upload and analysis are completed.
12. Scroll through the list of antivirus (AV) vendors that have been polled regarding this file. A green checkmark means no malware was detected.
13. Click the **DETAILS** tab and read through the analysis.
14. Use your browser's Back button to return to the VirusTotal home page.
15. Now you will analyze a website. Click **URL**.
16. Enter the URL of your school, place of employment, or another site with which you are familiar.
17. Wait until the analysis is completed.
18. Click the **DETAILS** tab and read through the analysis.
19. Click Scroll through the list of vendor analysis. Do any of these sites indicate **Unrated site** or **Malware site**?
20. How could VirusTotal be useful to users? How could it be useful to security researchers? Could it also be used by attackers to test their own malware before distributing it to ensure that it does not trigger an AV alert? What should be the protections against this?
21. Close all windows.

Project 3-2: Analyze Virus File Using VirusTotal—Part 2

Time Required: 20 minutes

Objective: Given a scenario, analyze potential indicators to determine the type of attack.

Description: What happens when VirusTotal detects a file-based virus? In this project, you will download a file that has a "signature" of a file-based virus into a sandbox in order to upload it to VirusTotal.

NOTE 25

None of the actions in this project will harm the underlying computer.

1. Open your web browser.
2. Enter the URL **www.eicar.org/?page_id=3950**.
3. Scroll down to **Download area using the standard protocol http**.
4. Click **eicar.com** to start the download.
5. Your antimalware software on your personal computer should immediately flag this file as malicious and not allow you to download it. Because it cannot (and should not) be downloaded on your regular computer, you will instead want to use the Windows Sandbox or VMware sandbox you created in Module 1.

NOTE 26

Refer to Project 1-3 and Project 1-4 of Module 1 for creating these sandboxes.

6. If you are using the Windows Sandbox, click **Start**, scroll down to Windows Sandbox, and then click **Windows Sandbox**.
7. First you will turn off the security protections in Windows Sandbox. Click **Start** and then **Windows Security**.
8. Click the three horizontal lines at the left of the screen to display the menu options.
9. Click **App & browser control**.
10. For each of the categories, click the **Off** button to turn off security. Remember this will only impact the security within the Windows Sandbox and will have no impact on the underlying computer.
11. Open Internet Explorer in the Windows Sandbox.

NOTE 27

Be sure to use the web browser in the Windows Sandbox and not the web browser in the underlying computer.

12. Enter the URL **www.eicar.org/?page_id=3950**.
13. Scroll down to **Download area using the standard protocol http**.
14. Click **eicar.com** to start the download.
15. The antimalware software within Windows Sandbox will now allow the file to be downloaded into the Sandbox.
16. Open another tab on the Internet Explorer web browser in the Windows Sandbox, and enter the URL **www.virustotal.com** (if you are no longer able to access the site through the web address, use a search engine to search for "Virus Total").
17. If necessary, click the **File** tab.
18. Click **Choose File**.
19. Navigate to the location of **eicar.com** and click **Open**.
20. Click **Confirm upload**.
21. Wait until the upload and analysis are completed.
22. Scroll through the list of AV vendors that have been polled regarding this file. A green checkmark means no malware was detected.
23. Click the **DETAILS** tab and read through the analysis.
24. Close the Windows Sandbox. This will delete the **eicar.com** file and reset the security settings to normal.

Project 3-3: Explore Ransomware Sites

Time Required: 15 minutes

Objective: Given a scenario, analyze potential indicators to determine the type of attack.

Description: A variety of sites provide information about ransomware along with tools for counteracting some types of infection. In this project, you explore different ransomware sites.

1. Open your web browser and enter the URL **www.nomoreransom.org** (if you are not able to access this site open a search engine and search for "**Nomoreransom.org**").
2. Click the **No** button.
3. Read through the Prevention Advice. Do you think it is helpful?
4. Click **Crypto Sheriff**. How could this be useful to a user who has suffered a ransomware infection?
5. Click **Ransomware: Q&A**. Read through the information. Which statements would you agree with? Which statements would you disagree with?

6. Click **Decryption Tools**. This contains a list of different tools that may help restore a computer that has been infected by a specific type of ransomware.
7. Click one of the tools and then click **Download** to download. Note that these tools change frequently based on the latest types of ransomware that is circulating.
8. Run the program to understand how these decryption tools function. Note that you will not be able to complete the process because there are no encrypted files on the computer. Close the program.
9. Now visit another site that provides ransomware information and tools. Open your web browser and enter the URL **id-ransomware.malwarehunterteam.com**.
10. What features does this site provide?
11. How could these sites be useful?
12. Close all windows.

Project 3-4: Use a Software Keylogger

CAUTION The purpose of this activity is to provide information regarding how these programs function in order that adequate defenses can be designed and implemented. These programs should never be used in a malicious fashion against another user.

Time Required: 25 minutes

Objective: Given a scenario, analyze potential indicators to determine the type of attack.

Description: A keylogger program captures everything that a user enters on a computer keyboard. In this project, you download and use a software keylogger.

1. Open your web browser and enter the URL **refog.com** (if you are no longer able to access the program through the URL, use a search engine to search for "Refog Keylogger").
2. Click **Features** to see the features of the product.
3. Click **Home**.
4. Click **Download**.
5. Click **Create an account** and enter the requested information.
6. Click **Download**.
7. When the file finishes downloading, run the installation program. Note that you may have to enter the password on the previous page to extract the files.
8. When prompted with **I'm going to use this software to monitor:** select **My own computer**.
9. Click **Hide program icon from Windows tray**. Click **Next**.
10. Click **I Agree**.
11. Click **Select All** and then **Next**.
12. Create a login and password for the online dashboard. Click **Activate**.
13. You will receive a message that the subscription has expired. Click **Yes** to install in offline mode.
14. Click **Install**.
15. Click **Restart Now**.
16. After the computer has restarted, use the keystroke combination **Ctrl + Alt + Shift + K** to launch Refog Keylogger.
17. Click **Tools** and then click **Settings**.
18. Note the default settings regarding what is captured.
19. Click **Back to log**.
20. Minimize Refog Keylogger.
21. Use your computer normally by opening a web browser to surf to a website. Open Microsoft Word and type several sentences. Open and close several programs on the computer.
22. Maximize Keylogger and note the information that was captured.
23. In the left pane, click through the different items that were captured.
24. Under Settings, click **Websites Visited**.
25. Under Websites Visited, click **Make website screenshots**.
26. Click **Apply**.

27. Open a web browser and surf to multiple websites.
28. Under Users, click **Websites visited**. Note the screen captures of the different sites.
29. What type of information would a software keylogger provide to a threat actor? How could it be used against the victim?
30. Click **File** and then **Exit** to close Keylogger.
31. You may uninstall Keylogger if you wish.
32. Close all windows.

Case Projects

Case Project 3-1: Biological and File-Based Viruses

The word virus comes from Latin, meaning a slimy liquid, poison, or poisonous secretion. In late Middle English, it was used for the venom of a snake. The word later evolved from the discharge to the substances within the body that caused the infectious diseases that produced the discharge. In 1799, Edward Jenner published his discovery that the cowpox virus could actually be used as a vaccine against smallpox. As biological science continued to advance, the word "virus" became even more specific when referring to tiny infectious agents—even smaller than bacteria—that replicate in living cells. This new field of virology exploded in the 1930s, when electronic microscopes allowed scientists to see viruses for the first time. Since then, scientists have continued to identify and name new biological viruses. Combating viruses by developing vaccines has many parallels to how malicious file-based viruses are identified and removed from a computer. Using the Internet, research these two types of viruses and find the similarities between combating biological and computer viruses. Write a one-to-two-paragraph summary of your research.

Case Project 3-2: Living-off-the-Land Binaries (LOLBins)

Fileless viruses take advantage of native services and processes that are part of the OS to avoid detection and carry out their attacks. These native services used in a fileless virus are called living-off-the-land binaries (LOLBins). Use the Internet to research fileless viruses and LOLBins. When did fileless viruses first appear? How do they compare with file-based viruses? What are the defenses against fileless viruses? Write a one-page paper on your research.

Case Project 3-3: Infamous Logic Bombs

Search the Internet for examples of logic bombs. Select four logic bombs and write a report about them. Who was responsible? When did the bombs go off? What was the damage? What was the penalty for the person responsible? Did the organization make any changes after the attack? How can they be prevented?

Case Project 3-4: Cybersecurity AI

The use of AI in cybersecurity is growing rapidly. Use the Internet to research the latest developments in cybersecurity AI. How does it work? What platforms are using it? What are some examples of it? How is it being improved? How can adversarial AI attacks be defended against? Write a one-page paper of what you have learned.

Case Project 3-5: Information Security Community Site Activity

The Information Security Community Site is an online companion to this textbook. It contains a wide variety of tools, information, discussion boards, and other features to assist learners. In order to gain the most benefit from the site, you will need to set up a free account.

Go to **community.cengage.com/infosec2**. Search the blogs on the topic "Ransomware." What did you learn? What were your biggest surprises? What did you already know? How could you use this information in your first security job?

Case Project 3-6: North Ridge Security

North Ridge Security provides security consulting and assurance services. You have recently been hired as an intern to assist them.

North Ridge is preparing a presentation to the monthly meeting of IT programmers and has asked you to do research on attacks on software.

1. Create a PowerPoint presentation on memory leaks, buffer overflow, integer overflow, pointer/object dereference, and attacks using API, device drivers, and DLLs. Your presentation should be at least nine slides in length.
2. As a follow-up to your presentation, you have been asked to write a one-page report on race conditions. Use the Internet to research race conditions and how they can best be addressed.

References

1. "The state of ransomware in the US: Report and statistics 2019," *Emisoft*, Dec. 12, 2019, accessed May 6, 2020, https://blog.emsisoft.com/en/34822/the-state-of-ransomware-in-the-us-report-and-statistics-2019/.
2. Dudley, Renee, "The extortion economy: How insurance companies are fueling a rise in ransomware attacks," *ProPublica*, Aug. 27, 2019, accessed Oct. 27, 2019, www.propublica.org/article/the-extortion-economy-how-insurance-companies-are-fueling-a-rise-in-ransomware-attacks.
3. "Definition of computer contaminant," *Law Insider*, accessed May 9, 2020, www.lawinsider.com/dictionary/computer-contaminant.
4. "McAfee Labs Threats Report," Dec. 2018, accessed Apr. 21, 2019, www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterly-threats-dec-2018.pdf.
5. "Ransomware payments up 33% as Maze and Sodinokibi proliferate in Q1 2020," *Coveware*, accessed May 5, 2020, www.coveware.com/blog/q1-2020-ransomware-marketplace-report.
6. Soare, Bianca, "This year in ransomware payouts (2019 edition)," *Heimdal Security*, Dec. 11, 2019, accessed May 9, 2020, https://heimdalsecurity.com/blog/ransomware-payouts/.
7. "Ransomware attacks map," *Statescoop,* accessed May 9, 2020, https://statescoop.com/ransomware-map/.
8. "High-impact ransomware attacks threaten U.S. businesses and organizations," *Public Service Announcement Federal Bureau of Investigation*, Oct 2, 2019, accessed May 9, 2020, www.ic3.gov/media/2019/191002.aspx.
9. "Ransomware payments up 33% as Maze and Sodinokibi proliferate in Q1 2020," *Coveware*, accessed May 5, 2020, www.coveware.com/blog/q1-2020-ransomware-marketplace-report.
10. Zumerle, Dioisio, D'Hoinne, Jeremy, and O'Neill, Mark, "How to build an effective API security strategy*,"* *Gartner Research*, Dec. 8, 2017, accessed May 12, 2020, www.gartner.com/en/documents/3834704.
11. "Reinventing cybersecurity with artificial intelligence: The new frontier in digital security," *Capgemini Research Institute,* accessed May 13, 2020, www.capgemini.com/wp-content/uploads/2019/07/AI-in-Cybersecurity_Report_20190711_V06.pdf.

MODULE 4

ENDPOINT AND APPLICATION DEVELOPMENT SECURITY

After completing this module, you should be able to do the following:

1. Describe different threat intelligence sources
2. List the steps for securing an endpoint
3. Explain how to create and deploy SecDevOps

Front-Page Cybersecurity

Suppose you were to uncover a zero-day vulnerability that nobody else knew about. What could you do with it?

You could be a broker and sell your knowledge of the vulnerability to a willing buyer. Governments often serve as buyers of zero-day vulnerabilities to protect their citizens or attack their enemies. Other willing buyers are third-party "acquisition platforms" who buy and then resell the vulnerability information to a government or a software developer. Very willing buyers are also threat actors who would craft their own attacks based on the vulnerability.

However, a more responsible option would be for you to privately disclose the vulnerability to the developer of the software, allowing the developer to fix it before an attacker uncovers and takes advantage of it. Not all zero-day vulnerabilities are discovered by individuals. Many organizations have internal security teams who look for vulnerabilities in their own code and in programs from other developers. When they find a problem, they privately contact the software developer with the information so the developer can patch it.

What happens if developers drag their feet and do not patch the vulnerability within a reasonable period of time? Delaying can keep the door open for a threat actor to find the zero-day vulnerability and exploit it.

Faced with the conundrum of trying to give developers time to create a patch and the public's right to protection from attacks, organizations have come up with "disclosure deadlines," or the time that the software developer is given to patch a vulnerability before it is publicly disclosed. These disclosure deadlines generally range from 45 days to 120 days.

Google started Project Zero in 2014 to look for zero-day vulnerabilities in its code and code from other software developers. Many developers responded quickly to a private alert of a zero-day vulnerability. As soon as the patch was available, Project Zero released information about it. However, not all developers moved quickly: in some instances, it took six months for the software developers to issue a patch.

Google's Project Zero set a disclosure deadline of 90 days, after which it would go public with information about the vulnerability. Going public would alert the user base to be aware of the vulnerability and possibly stop using the software while shaming the developer into a faster response. The next year, Project Zero added a 14-day grace period: after the 90-day deadline, a developer could request two additional weeks to make a patch available. Beyond 14 days, however, Google would go public with the information. Project Zero said that by 2019, 97.7 percent of vulnerabilities uncovered were fixed within 90 days.

However, in early 2020, Project Zero announced a trial tweak to their disclosure deadline. The limit is still 90 days. But Project Zero sometimes found that software developers were rushing to fix the zero-day vulnerability to meet the 90-day deadline, and this sometimes resulted in a rushed and flawed patch that introduced another vulnerability into the code. With Project Zero's new trial tweak, software developers will now have the full 90 days to create a patch for the vulnerability, and Project Zero will say nothing until the end of the 90 days. Even if the developer can fix the vulnerability in just 20 days, Google will still say nothing until 90 days have elapsed. However, the developer and Google can make a mutual disclosure before the 90-day deadline if both sides agree on the early disclosure.

Why the change? Project Zero says that faster patch development, while still important, is not the exclusive goal anymore. Instead, thorough patch development—not sticking a bandage on it but actually finding the underlying problem and correcting it—and improved patch adoption are also important. In the words of Project Zero, "End user security doesn't improve when a bug is found, and it doesn't improve when a bug is fixed. It improves once the end user is aware of the bug and typically patches their device."

Attacks using malware, application attacks, and adversarial artificial intelligence attacks against endpoints continue around the clock every day. What defenses can be used to ward off these attacks?

First, it is important to access threat intelligence sources in order to be aware of the latest types of attacks and how to defend against them. With that information at hand, securing devices through boot integrity, protecting endpoints, and hardening endpoints can then effectively be used. But deploying these defenses is still considered "after the fact." A growing chorus of security professionals today are demanding that it is the responsibility of the software developers to create and deploy software using secure application development coding techniques.

In this module, you will learn about securing endpoint devices and creating and deploying secure applications. But first you will explore the sources of threat intelligence information.

THREAT INTELLIGENCE SOURCES

✓ CERTIFICATION

1.5 Explain different threat actors, vectors, and intelligence sources.

At one time, organizations were reluctant to share information about attacks on their networks and endpoints, often because they were concerned about "bad publicity" that might arise from this disclosure. Today that is no longer the case. Organizations are pooling their experiences and knowledge gained about the latest attacks with the broader security community. Sharing this type of information has become an important aid to help other organizations shore up their defenses.

One type of shared information is the evidence of an attack. Most organizations monitor their networking environment to determine what normally occurs. They use this data to create a database of *key risk indicators (KRIs)*. A KRI is a metric of the upper and lower bounds of specific indicators of normal network activity. These indicators may include the total network logs per second, number of failed remote logins, network bandwidth, and outbound email traffic. A KRI exceeding its normal bounds could be (but is not always) an **indicator of compromise (IOC)**. An IOC shows that a malicious activity is occurring but is still in the early stages of an attack.

Making IOC information available to others can prove to be of high value as it may indicate a common attack that other organizations are also experiencing or will soon experience. This information aids others in their **predictive analysis** or discovering an attack before it occurs.

NOTE 1

Like radar that shows the enemy approaching, predictive analysis helps determine when and where attacks may occur.

Threat intelligence sources fall into several categories, as do the sources from which threat intelligence information can be gathered.

Categories of Sources

The two categories of threat intelligence sources are open source and closed source.

Open Source Information

The phrase **open source** has its roots in the computer industry. Initially, it referred to software for which the *source code* was open for anyone to examine. Over time, "open source" was used to refer to anything that could be freely used without restrictions, such as an *open source film* or *open source curriculum*.

NOTE 2

The phrase "open source" came out of a strategy session in 1998 by about a half dozen Linux software developers who wanted to take advantage of an announcement by Netscape that it was planning to give away the source code of its browser. The developers wanted to persuade the corporate world about the superiority of an open software development process but found that the phrase "free software" that had been used previously carried a stigma of being inferior. After a brainstorming session, the label "open source" was agreed upon.

Open source threat intelligence information that is freely available, often called open source intelligence (OSINT), has become a vital resource. This information is often collected and then disseminated through **public information sharing centers**. A typical sharing center is the U.S. Department of Homeland Security (DHS) Cyber Information Sharing and Collaboration Program (CISCP). The CISCP "enables actionable, relevant, and timely unclassified information exchange through trusted public-private partnerships across all critical infrastructure sectors." With the DHS serving as the coordinator, the CISCP enables its members (called "partners") to not only share threat and vulnerability information but also take advantage of the DHS's cyber resources. Some of the CISCP services include the following:

- *Analyst-to-analyst technical exchanges*. Partners can share and receive information on threat actor tactics, techniques, and procedures (TTPs) and emerging trends.
- *CISCP analytical products*. A portal can be accessed through which partners can receive analysis of products and threats.
- *Cross industry orchestration*. Partners can share lessons learned and their expertise with peers across common sectors.
- *Digital malware analysis*. Suspected malware can be submitted to be analyzed and then used to generate malware analysis reports to mitigate threats and attack vectors.

The two concerns around public information sharing centers are the privacy of shared information and the speed at which the information is shared.

NOTE 3

The CISCP program is free to join and use. Those interested must agree to a Cyber Information Sharing and Collaboration Agreement (CISCA), which enables DHS and its partners to exchange anonymized information. Once partners sign the agreement, DHS coordinates an on-boarding session to customize how DHS and the organization can exchange information.

Privacy A concern about using public information sharing centers is that of privacy. An organization that is the victim of an attack must be careful not to share proprietary or sensitive information when providing IOCs and attack details.

As a safeguard, most public information sharing centers have protections in place to prevent the disclosure of proprietary information. For example, Table 4-1 lists the privacy protections of the CISCP.

Speed Threat intelligence information must be distributed as quickly as possible to others. To rely on email alerts that require a human to read them and then react takes far too much time. As an alternative, **Automated Indicator Sharing (AIS)** can be used instead. AIS enables the exchange of cyberthreat indicators between parties through computer-to-computer communication, not email communication. Threat indicators such malicious IP addresses or the sender address of a phishing email can be quickly distributed to enable others to repel these attacks.

NOTE 4

Those participating in AIS generally are connected to a managed system controlled by the public information sharing center that allows bidirectional sharing of cyberthreat indicators. Not only do participants receive indicators, but they can also share indicators they have observed in their own network defenses to the public center, which then distributes them to all participants.

Table 4-1 CISCP privacy protections

Protection	Explanation	Example
Cybersecurity Information Sharing Act (CISA)	CISA is a federal law passed in 2015 that provides authority for cybersecurity information sharing between the private sector, state, and local governments and the federal government.	CISA requires a non-federal entity to remove any information from a cyberthreat indicator that it knows at the time of sharing to be personal information of a specific individual or information that identifies a specific individual that is not directly related to a cybersecurity threat.
Freedom of Information Act (FOIA)	FOIA was passed in 1967 and provides the public the right to request access to records from any federal agency.	Although federal agencies are required to disclose any information requested under the FOIA, they offer nine exemptions, one of which protects interests such as personal privacy.
Traffic-Light Protocol (TLP)	TLP is a set of designations used to ensure that sensitive information is shared only with the appropriate audience.	TLP uses four colors (red, amber, green, and white) to indicate the expected sharing limitations the recipients should apply.
Protected Critical Infrastructure Information (PCII)	The PCII Act of 2002 protects private sector infrastructure information that is voluntarily shared with the government for the purposes of homeland security.	To qualify for PCII protections, information must be related to the security of the critical infrastructure, voluntarily submitted, and not submitted in place of compliance with a regulatory requirement.

NOTE 5

APIs are covered in Module 3.

Two tools facilitate AIS. **Structured Threat Information Expression (STIX)** is a language and format used to exchange cyberthreat intelligence. All information about a threat can be represented with objects and descriptive relationships. STIX information can be visually represented for a security analyst to view or stored in a lightweight format to be used by a computer. **Trusted Automated Exchange of Intelligence Information (TAXII)** is an application protocol for exchanging cyberthreat intelligence over Hypertext Transfer Protocol Secure (HTTPS). TAXII defines an *application protocol interface (API)* and a set of requirements for TAXII clients and servers.

Closed Source Information

NOTE 6

AIS is used more extensively with public information sharing centers than private centers.

Closed source is the opposite of open source. It is *proprietary*, meaning it is owned by an entity that has an exclusive right to it. Organizations that are participants in closed source information are part of **private information sharing centers** that restrict both access to data and participation. Whereas private sharing centers are similar to public sharing centers in that members share threat intelligence information, insights, and best practices, private sharing centers are restrictive regarding who may participate. All candidates must go through a vetting process and meet certain criteria.

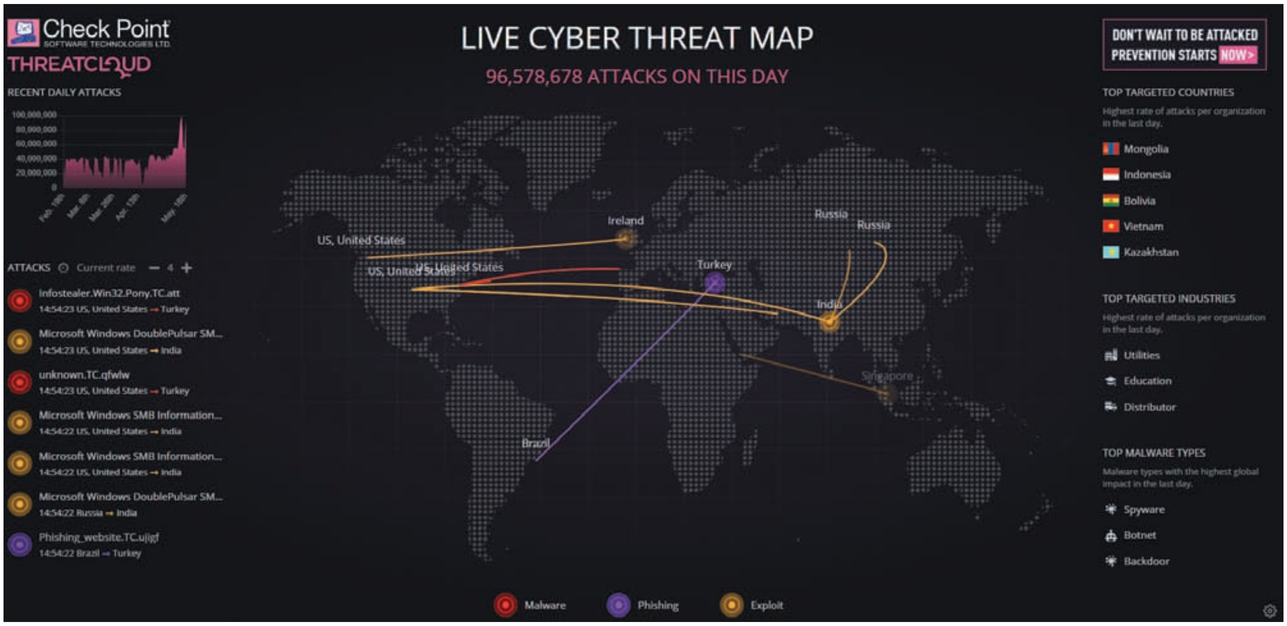
Sources of Threat Intelligence

Several sources of threat intelligence are useful. These include the following:

- *Vulnerability databases*. A **vulnerability database** is a repository of known vulnerabilities and information as to how they have been exploited. These databases create "feeds" of the latest cybersecurity incidences. Common cybersecurity data feeds include *vulnerability feeds* that provide information on the latest vulnerabilities and *threat feeds* that outline current threats and attacks. The *adversary tactics, techniques, and procedures (TTP)* is a database of the behavior of threat actors and how they orchestrate and manage attacks.
- *Threat maps*. A cybersecurity **threat map** illustrates cyberthreats overlaid on a diagrammatic representation of a geographical area. Figure 4-1 illustrates a threat map. Threat maps help in visualizing attacks and provide a limited amount of context of the source and the target countries, the attack types, and historical and near real-time data about threats.

NOTE 7

Data feeds, vulnerability feeds, threat feeds, and TTP are covered in Module 2.



Infostealer.Win32.Pony.TC.att
14:54:23 US, United States → Turkey

Microsoft Windows DoublePulsar SM...
14:54:23 US, United States → India

unknown.TC.qfwlw
14:54:23 US, United States → Turkey

Microsoft Windows SMB Information...
14:54:22 US, United States → India

Microsoft Windows SMB Information...
14:54:22 US, United States → India

Microsoft Windows DoublePulsar SM...
14:54:22 Russia → India

Phishing_website.TC.ujigf
14:54:22 Brazil → Turkey

TOP TARGETED COUNTRIES

Highest rate of attacks per organization in the last day.

- Mongolia
- Indonesia
- Bolivia
- Vietnam
- Kazakhstan

TOP TARGETED INDUSTRIES

Highest rate of attacks per organization in the last day.

- Utilities
- Education
- Distributor

TOP MALWARE TYPES

Malware types with the highest global impact in the last day.

- Spyware
- Botnet
- Backdoor

Source: Check Point Software Technologies Ltd.

Figure 4-1 Threat map

CAUTION

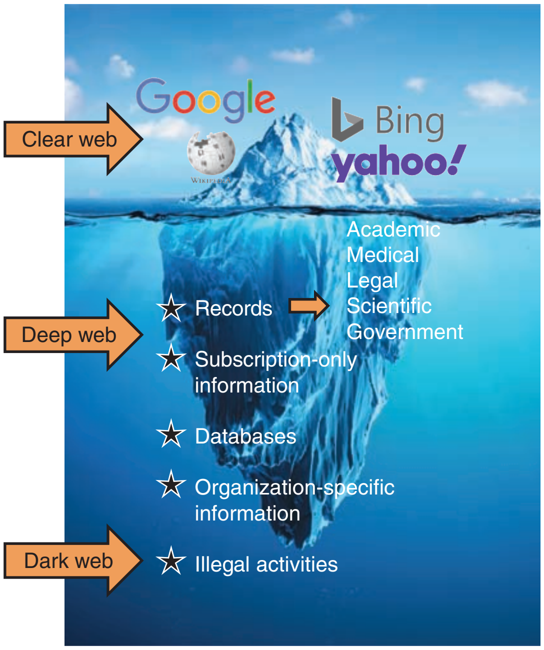
Threat maps may look impressive, but in reality, they provide limited valuable information. Many maps claim that they show data in real time, but most are simply a playback of previous attacks. Because threat maps show anonymized data, it is impossible to know the identity of the attackers or the victims. Also, threat actors usually mask their real locations, so what is displayed on a threat map is incorrect. As a result, many cybersecurity professionals question the value of threat maps.

- *File and code repositories*. **File and code repositories** are another source of threat intelligence. Victims of an attack can upload malicious files and software code that can then be examined by others to learn more about the attacks and craft their defenses. Several entities of the U.S. government—including the Federal Bureau of Investigation (FBI), the Cybersecurity and Infrastructure Security Agency (CISA), and the Department of Defense (DoD) U.S. Cyber Command—are particularly active in posting to file and code repositories. Often samples of recently discovered malware variants are uploaded to the VirusTotal malware aggregation repository along with published detailed malware analysis reports (MARs) containing IOCs for each malware variant.

NOTE 8

The Hands-On Projects 3-1 and 3-2 in Module 3 provide you with experience using VirusTotal.

- *Dark web*. The web has three levels, as illustrated in Figure 4-2: the *clear web*, which includes ordinary websites (social media, ecommerce, news, etc.) that most users access regularly and can be located by a search engine; the *deep web*, which includes exclusive and protected websites (corporate email, material behind a digital paywall, cloud hosting services, etc.) that are hidden from a search engine and cannot be accessed without valid credentials; and the **dark web**. The dark web is like the deep web in that it is beyond the reach of a normal search engine, but it is the domain of threat actors. Using special software such as *Tor* or *I2P (Invisible Internet Project*) this software will mask the user's identity to allow for malicious activity such as selling drugs and stolen personal information and buying and selling malicious software used for attacks. Some security professionals and organizations use the dark web on a limited basis to look for signs that information critical to that enterprise is being sought out or sold on the dark web.



Source: Romolo Tavani/Shutterstock.com

Figure 4-2 Dark web

CAUTION

Finding information on the dark web is difficult. First, it requires using Tor or IP2, which prevents a device's IP address being traced. Second, although some dark web search engines are available, they are unlike regular search engines such as Google. The dark web search engines are difficult to use and notoriously inaccurate. One reason is because merchants who buy and sell stolen data or illicit drugs are constantly on the run, and their dark websites appear and then suddenly disappear with no warning. Finally, dark websites use a naming structure that results in URLs such as *p6f47s5p3dq3qkd.onion*. All of these are hurdles that keep out anyone who does not understand these inner workings.

TWO RIGHTS & A WRONG

1. Two concerns about public information sharing centers are the privacy of shared information and the speed at which the information is shared.
2. Two tools that facilitate AIS are STIX and TAXII.
3. Security professionals consider threat maps a vital source of information.

See Appendix B for the answer.

SECURING ENDPOINT COMPUTERS

CERTIFICATION

3.2 Given a scenario, implement host or application security solutions.

4.4 Given an incident, apply mitigation techniques or controls to secure an environment.

Despite the fact that endpoint devices like smartphones, tablets, and wearable fitness trackers receive the bulk of attention today, the "workhorse" of technology remains the personal computer. More than 260 million computers are sold annually in the United States[1] compared to 160 million smartphones sold each year.[2] While all endpoints must be protected from attacks, endpoint desktop and laptop computers must be secured because they are connected to corporate networks and its data, they contain data stored locally, and they can be used as a springboard to attack other endpoints.

Securing endpoint computers primarily involves three major tasks: *confirming* that the computer has started securely, *protecting* the computer from attacks, and then *hardening* it for even greater protection.

Confirm Boot Integrity

One of the steps that is often overlooked in securing endpoint computers is to confirm that the computer has started without any malicious activity taking place. Ensuring secure startup involves the Unified Extensible Firmware Interface (UEFI) and its boot security features.

Unified Extensible Firmware Interface (UEFI)

Early cowboys and workhands were known for wearing tall, tight-fitting boots. These boots had a tab or loop at the top through which a tool called a boot hook could be inserted to assist in pulling on the boot. In the mid-1800s, the expression *pull yourself up by your own bootstraps* was used to describe an impossible task of lifting oneself off the ground by pulling on the bootstrap. The phrase later came to mean to improve your situation by your own efforts without any external help.

Computers adopted this language to describe the process of starting a computer when it has been powered off. Because a computer is unable to rely on external assistance when powered on, starting a computer is called *booting up* or just *booting*.

The booting process on early personal computers, both Apple Mac and Windows PC, used firmware called the *BIOS (Basic Input/Output System)*. The BIOS was a chip integrated into the computer's motherboard. When the computer was powered on, the BIOS software would "awaken" and perform the following steps in a *legacy BIOS boot*:

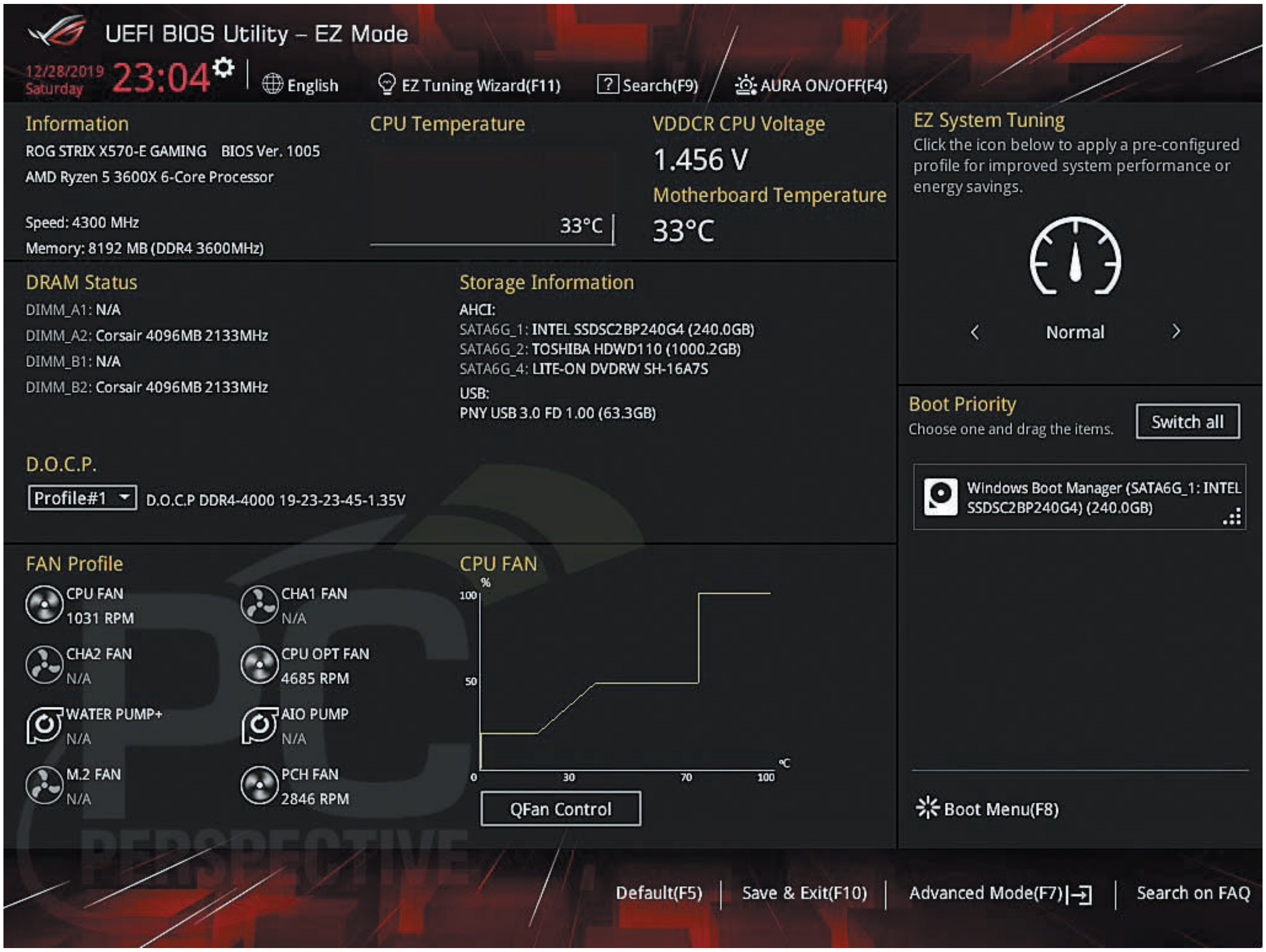
1. The BIOS would first test the various components of the computer to ensure that they were functioning properly (called the *POST* or *Power-On Self-Test*).
2. Next, the BIOS would reference the *Master Boot Record* (*MBR*) that specified the computer's *partition table*, which instructed the BIOS where the computer's operating system (OS) could be located.
3. Finally, the BIOS passed control to the installed boot loader, which launched the OS.

Originally, BIOS firmware was stored in a *ROM* (*read-only memory*) chip on the motherboard, supplemented by a *CMOS* (*complementary metal-oxide-semiconductor*) chip that stored any changes to the BIOS. Later computer systems stored the BIOS contents in *flash memory* so it could be easily updated. This provided the ability to update to the BIOS firmware so new features could be added.

NOTE 9

Although BIOS chips were nonvolatile (they retained the information even when the computer was turned off), CMOS needed its own dedicated power source, which was a lithium-ion battery about the size of a coin that could hold a charge for up to 10 years before needing to be replaced. If the CMOS battery died, the BIOS settings were not lost but instead were reset to their default settings.

To add functionality, an improved firmware interface was developed to replace the BIOS. Known as **UEFI (Unified Extensible Firmware Interface)**, it provides several enhancements over BIOS. This includes the ability to access hard drives that are larger than two terabytes (TB), support for an unlimited number of primary hard drive partitions, faster booting, and support for networking functionality in the UEFI firmware itself to aid in remote troubleshooting. UEFI also has a more advanced user interface for configurations and information, as seen in Figure 4-3.


UEFI BIOS Utility – EZ Mode

12/28/2019 Saturday 23:04

Information
ROG STRIX X570-E GAMING BIOS Ver. 1005
AMD Ryzen 5 3600X 6-Core Processor
Speed: 4300 MHz
Memory: 8192 MB (DDR4 3600MHz)

CPU Temperature: 33°C

VDDCR CPU Voltage: 1.456 V

Motherboard Temperature: 33°C

EZ System Tuning
Click the icon below to apply a pre-configured profile for improved system performance or energy savings.
Normal

DRAM Status
DIMM_A1: N/A
DIMM_A2: Corsair 4096MB 2133MHz
DIMM_B1: N/A
DIMM_B2: Corsair 4096MB 2133MHz

Storage Information
AHCI:
SATA6G_1: INTEL SSDSC2BP240G4 (240.0GB)
SATA6G_2: TOSHIBA HDWD110 (1000.2GB)
SATA6G_4: LITE-ON DVDRW SH-16A7S
USB:
PNY USB 3.0 FD 1.00 (63.3GB)

Boot Priority
Choose one and drag the items.
Windows Boot Manager (SATA6G_1: INTEL SSDSC2BP240G4) (240.0GB)

D.O.C.P.
Profile#1 D.O.C.P DDR4-4000 19-23-23-45-1.35V

FAN Profile
CPU FAN 1031 RPM
CHA1 FAN N/A
CHA2 FAN N/A
CPU OPT FAN 4685 RPM
WATER PUMP+ N/A
AIO PUMP N/A
M.2 FAN N/A
PCH FAN 2846 RPM

Source: ASUS

Figure 4-3 UEFI user interface

NOTE 10

Legacy BIOS boot support from motherboard manufacturers ended in 2020 and UEFI is now the standard.

Boot Security

Another significant improvement of UEFI over BIOS relates to boot security. The ability to update the BIOS in firmware also opened the door for a threat actor to create malware to infect the BIOS. Called a *BIOS attack*, it would exploit the update feature of the BIOS. Because the BIOS resides in firmware and an infected BIOS would then persistently re-infect the computer whenever it was powered on, BIOS attacks were difficult to uncover and hard to disinfect. UEFI, used along with other components, is designed to combat these BIOS vulnerabilities and provide improved boot security.

CAUTION UEFI by itself does not provide enhanced boot security. It must be paired with other boot security functions.

Boot security involves validating that each element used in each step of the boot process has not been modified. This process begins with the validation of the first element (boot software). Once the first element has been validated, it can then validate the next item (such as software drivers) and so on until control has been handed over to the OS.

This is called a *chain of trust*: each element relies on the confirmation of the previous element to know that the entire process is secure.

But how does the chain begin? What if a threat actor were to inject malware prior to start of the chain of trust? If the starting point is software, it can be replaced or modified. That would then compromise each element of the chain. To prevent this, a chain of trust requires a strong starting point.

The strongest starting point is hardware, which cannot be modified like software. This is known as the **hardware root of trust**. Security checks are "rooted" in (begin with) hardware checks. Because this chain of trust begins with a hardware verification, each subsequent check can rely upon it (called **boot attestation**).

Several techniques can be used to assure boot security, all of which rely on UEFI; some also rely on the hardware root of trust. Boot security modes are listed in Table 4-2.

Table 4-2 Boot security modes

Name	Description	Advantages	Disadvantages
Legacy BIOS Boot	Uses BIOS for boot functions	Compatible with older systems	No security features
UEFI Native Mode	Uses UEFI standards for boot functions	Security boot modules can be patched or updated as needed.	No validation or protection of the boot process
Secure Boot	Each firmware and software executable at boot time must be verified as having prior approval.	All system firmware, bootloaders, kernels, and other boot-time executables are validated.	Custom hardware, firmware, and software may not pass without first being submitted to system vendors like Microsoft.
Trusted Boot	Windows OS checks the integrity of every component of boot process before loading it.	Takes over where Secure Boot leaves off by validating the Windows 10 software before loading it	Requires using Microsoft OS
Measured Boot	Computer's firmware logs the boot process so the OS can send it to a trusted server to assess the security.	Provides highest degree of security	Could slow down the boot process

CAUTION

The Secure Boot security standard is designed to ensure that a computer boots using only software that is trusted by the computer manufacturer. Manufacturers can update the list of trusted hardware, drivers, and OS for a computer, which are stored in the Secure Boot database on the computer. Although it is possible for the user to disable Secure Boot to install hardware or run software or OS that have not been trusted by the manufacturer, this makes it difficult or impossible to reactivate Secure Boot without restoring the computer back to its original factory state.

Protect Endpoints

Once boot security has been established; the computer endpoints must be actively protected. The protection can be done through software installed on the endpoint, such as antivirus software, antimalware, web browser protections, and monitoring and response systems.

Antivirus

One of the first software protections was **antivirus (AV)** software. This software can examine a computer for file-based virus infections as well as monitor computer activity and scan new documents that might contain a virus. (Scanning is typically performed when files are opened, created, or closed.) If a virus is detected, options generally include cleaning the file of the virus, quarantining the infected file, or deleting the file. Log files created by AV products can also provide beneficial information regarding attacks.

NOTE 11

Viruses are covered in Module 3.

Many AV products use signature-based monitoring, also called *static analysis*. The AV software scans files by attempting to match known virus patterns against potentially infected files (called *string scanning*). Other variations include *wildcard scanning* (a wildcard is allowed to skip bytes or ranges of bytes instead of looking for an exact match) and *mismatch scanning* (mismatches allow a set number of bytes in the string to be any value regardless of their position in the string).

CAUTION The weakness of signature-based monitoring is that the AV vendor must constantly be searching for new viruses, extracting virus signatures, and distributing those updated databases to all users. Any out-of-date signature database could result in an infection.

A newer approach to AV is heuristic monitoring (called *dynamic analysis*), which uses a variety of techniques to spot the characteristics of a virus instead of attempting to make matches. The difference between static analysis and dynamic analysis detection is similar to how airport security personnel in some nations screen for terrorists. A known terrorist attempting to go through security can be identified by comparing his face against photographs of known terrorists (static analysis). What about a new terrorist with no photograph? Security personnel can look at the person's characteristics—holding a one-way ticket, not checking any luggage, showing extreme nervousness—as possible indicators that the individual may need to be questioned (dynamic analysis).

NOTE 12

One AV heuristic monitoring technique used is *code emulation* in which a virtual environment is created that simulates the CPU and memory of the computer. Any questionable program code is executed in the virtual environment (no actual virus code is executed by the real CPU) to determine if it is a virus.

Antimalware

Instead of only protecting against file-based viruses as with AV, **antimalware** is a suite of software intended to provide protections against multiple types of malware, such as ransomware, cryptomalware, and Trojans.

Some antimalware software protects against spam that has evaded the corporate email gateway and monitors emails for spam and other unwanted content. Antimalware spam protection is often performed using a technique called *Bayesian filtering*. The software divides email messages that have been received into two piles, spam and nonspam. The filter then analyzes every word in each email and determines how frequently a word occurs in the spam pile compared to the nonspam pile. A word such as "the" would occur equally in both piles and be given a neutral 50 percent ranking. A word such as "report" may occur frequently in nonspam messages and would receive a 99 percent probability of being a nonspam word, while a word like "sex" may receive a 99 percent probability of being a spam word. Whenever email arrives, the filter looks for the 15 words with the highest probabilities to calculate the message's overall spam probability rating.

NOTE 13

Bayesian filters generally trap a much higher percentage of spam than other techniques.

Another component of an antimalware suite is *antispyware*, which helps prevent computers from becoming infected by spyware. One common technique is to use a *pop-up blocker*. A *pop-up* is a small web browser window that appears over a webpage. Most pop-up windows are created by advertisers and launch as soon as a new website is visited. Using a pop-up blocker, users can often select the level of blocking, ranging from blocking all pop-ups to allowing specific pop-ups.

NOTE 14

A pop-up blocker can be part of an antimalware package, a separate program, or a feature incorporated within a browser that stops pop-up advertisements from appearing.

Web Browsers

Web browsers have a degree of security that can protect endpoint computers. This security includes secure cookies and HTTP headers.

Secure Cookies The Hypertext Transfer Protocol (HTTP) is the Internet-based protocol that is the foundation of all data exchanges on the web. It is a client-server protocol so that requests are initiated by the recipient or client, usually a web browser, to a web server.

One of the limitations of HTTP is that it is a *stateless protocol*. Unlike a *stateful protocol*, which "remembers" everything that occurs between the browser client and the server, a stateless protocol "forgets" what occurs when the session is interrupted or ends. Three ways the stateless protocol HTTP can mimic a stateful protocol include the following:

- Using a URL extension so the state is sent as part of the URL as a response
- Using "hidden form fields" in which the state is sent to the client as part of the response and returned to the server as part of a form's hidden data
- Storing user-specific information in a file on the user's local computer and then retrieve it later in a file called a *cookie*.

Websites use several types of cookies. A *first-party cookie* is created from the website that a user is currently viewing; whenever the user returns to this site, that cookie is used by the site to view the user's preferences and better customize the browsing experience. Some websites attempt to place additional cookies on the local hard drive. These cookies often come from third parties that advertise on the site and want to record the user's preferences. These cookies are called *third-party cookies*. A *session cookie* is stored in random-access memory (RAM), instead of on the hard drive, and only lasts for the duration of visiting the website.

Cookies can pose security risks as well as privacy risks. First-party cookies can be stolen and used to impersonate the user, while third-party cookies can be used to track the browsing or buying habits of a user. When multiple websites are serviced by a single marketing organization, cookies can be used to track browsing habits on all the client's sites.

As a means of protection for cookies, a web browser can send a **secure cookie**. This cookie is only sent to the server with an encrypted request over the secure HTTPS protocol. This prevents an unauthorized person from intercepting a cookie that is being transmitted between the browser and the web server.

NOTE 15

A cookie can contain a variety of information based on the user's preferences when visiting a website. For example, if a user inquires about a rental car at the car agency's website, that site might create a cookie that contains the user's travel itinerary. In addition, it may record the pages visited on a site to help the site customize the view for any future visits. Cookies can also store any personally identifiable information (name, email address, work address, telephone number, and so on) that was provided when visiting the site; however, a website cannot gain access to private information stored on the local computer.

HTTP Response Headers When users visit a website through their web browser, the web server answers back with **HTTP Response Headers**. These headers tell the browser how to behave while communicating with the website. Several HTTP Response Headers can improve security; these are listed in Table 4-3.

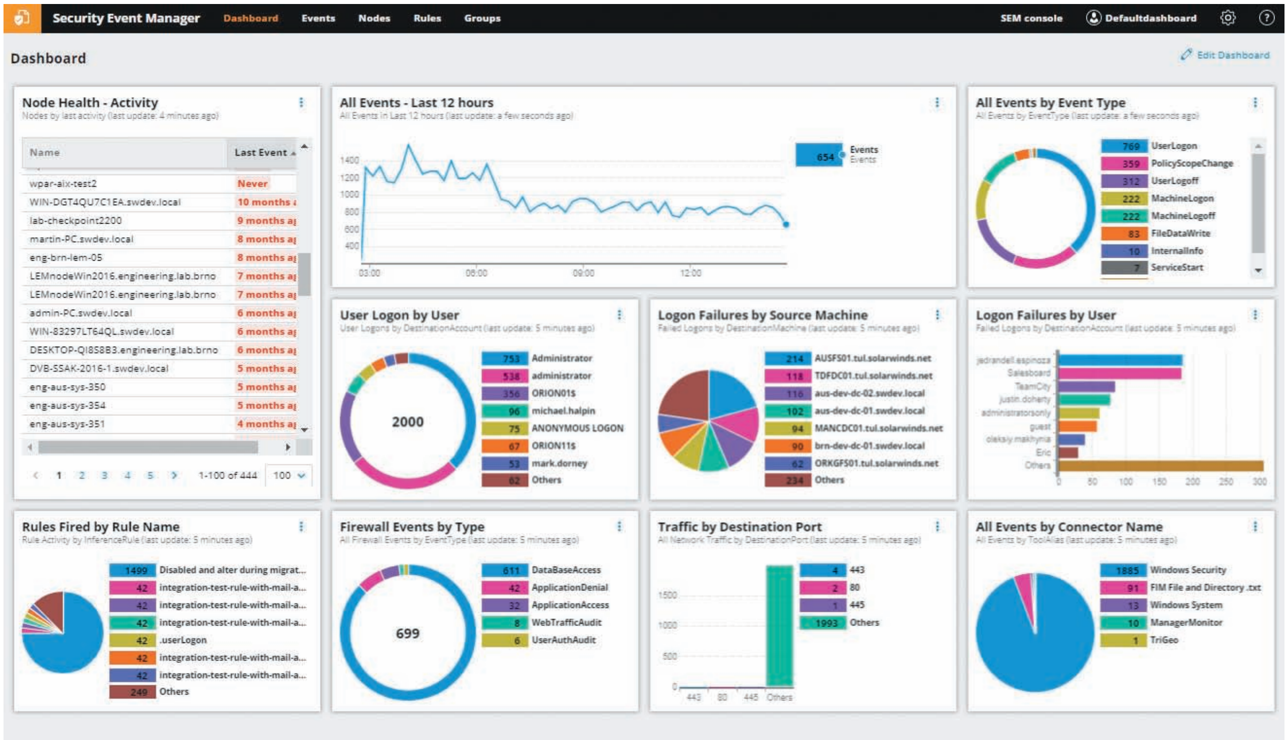
Table 4-3 HTTP response headers

HTTP response header	Description	Protection
HTTP Strict Transport Security (HSTS)	Forces browser to communicate over more secure HTTPS instead of HTTP	Encrypts transmissions to prevent unauthorized user from intercepting
Content Security Policy (CSP)	Restricts the resources a user is allowed to load within the website	Protects against injection attacks
Cross Site Scripting Protection (X-XSS)	Prohibits a page from loading if it detects a cross-site scripting attack	Prevents XSS attacks
X-Frame-Options	Prevents attackers from "overlaying" their content on the webpage	Foils a threat actor's attempt to trick a user into providing personal information

Monitoring and Response Systems

The three types of monitoring and response systems for endpoint computers are host intrusion detection systems (HIDS), host intrusion prevention systems (HIPS), and endpoint detection and response (EDR).

Host Intrusion Detection Systems (HIDS) A **host intrusion detection system (HIDS)** is a software-based application that runs on an endpoint computer and can detect that an attack has occurred. The primary function of a HIDS is automated detection, which saves someone from sorting through log files to find an indication of unusual behavior. HIDS can quickly detect evidence that an intrusion has occurred. Figure 4-4 shows a HIDS dashboard.


Security Event Manager
Dashboard
Events
Nodes
Rules
Groups
SEM console
Defaultdashboard
Dashboard
Edit Dashboard
Node Health - Activity
Nodes by last activity (last update: 4 minutes ago)
1-100 of 444
100
All Events - Last 12 hours
All Events in Last 12 hours (last update: a few seconds ago)
654 Events
1400
1200
1000
800
600
400
03:00
06:00
09:00
12:00
All Events by Event Type
All Events by EventType (last update: a few seconds ago)
769 UserLogon
359 PolicyScopeChange
312 UserLogoff
222 MachineLogon
222 MachineLogoff
83 FileDataWrite
10 InternalInfo
7 ServiceStart
User Logon by User
User Logons by DestinationAccount (last update: 5 minutes ago)
2000
753 Administrator
538 administrator
356 ORION01$
96 michael.halpin
75 ANONYMOUS LOGON
67 ORION11$
53 mark.dorney
62 Others
Logon Failures by Source Machine
Failed Logons by DestinationMachine (last update: 5 minutes ago)
214 AUSFS01.tul.solarwinds.net
118 TDFDC01.tul.solarwinds.net
116 aus-dev-dc-02.swdev.local
102 aus-dev-dc-01.swdev.local
94 MANCDC01.tul.solarwinds.net
90 brn-dev-dc-01.swdev.local
62 ORKGFS01.tul.solarwinds.net
234 Others
Logon Failures by User
Failed Logons by DestinationAccount (last update: 5 minutes ago)
jedrandell.espinoza
Salesboard
TeamCity
justin.doherty
administratorsonly
guest
oleksiy.makhynia
Eric
Others
0
50
100
150
200
250
300
Rules Fired by Rule Name
Rule Activity by InferenceRule (last update: 5 minutes ago)
1499 Disabled and alter during migrat...
42 integration-test-rule-with-mail-a...
42 integration-test-rule-with-mail-a...
42 integration-test-rule-with-mail-a...
42 .userLogon
42 integration-test-rule-with-mail-a...
42 integration-test-rule-with-mail-a...
249 Others
Firewall Events by Type
All Firewall Events by EventType (last update: 5 minutes ago)
699
611 DataBaseAccess
42 ApplicationDenial
32 ApplicationAccess
8 WebTrafficAudit
6 UserAuthAudit
Traffic by Destination Port
All Network Traffic by DestinationPort (last update: 5 minutes ago)
4 443
2 80
1 445
1993 Others
1500
1000
500
0
443
80
445
Others
All Events by Connector Name
All Events by ToolAlias (last update: 5 minutes ago)
1885 Windows Security
91 FIM File and Directory .txt
13 Windows System
10 ManagerMonitor
1 TriGeo

Name	Last Event
wpar-aix-test2	Never
WIN-DGT4QU7C1EA.swdev.local	10 months
lab-checkpoint2200	9 months a
martin-PC.swdev.local	8 months a
eng-brn-lem-05	8 months a
LEMnodeWin2016.engineering.lab.brno	7 months a
LEMnodeWin2016.engineering.lab.brno	7 months a
admin-PC.swdev.local	6 months a
WIN-83297LT64QL.swdev.local	6 months a
DESKTOP-QI8S8B3.engineering.lab.brno	6 months a
DVB-SSAK-2016-1.swdev.local	5 months a
eng-aus-sys-350	5 months a
eng-aus-sys-354	5 months a
eng-aus-sys-351	4 months a

Source: SolarWinds

Figure 4-4 HIDS dashboard

A HIDS relies on agents installed directly on the endpoint, and these agents work closely with the OS to observe activity. HIDSs typically monitor these types of endpoint computer functions:

- *System calls*. Each operation in a computing environment starts with a *system call*. A system call is an instruction that interrupts the program being executed and requests a service from the operating system. HIDS can monitor system calls based on the process, mode, and action being requested.
- *File system access*. System calls usually require specific files to be opened to access data. A HIDS works to ensure that all file openings are based on legitimate needs and are not the result of malicious activity.
- *Host input/output*. HIDS monitors all input and output communications to watch for malicious activity. For example, if the system never uses instant messaging (IM) and suddenly a threat attempts to open an IM connection from the system, the HIDS would detect this as anomalous activity.

Host Intrusion Prevention Systems (HIPS) As its name implies, an intrusion *prevention* system not only monitors to detect malicious activities but also attempts to stop them. A **host intrusion prevention system (HIPS)** monitors endpoint activity to immediately block a malicious attack by following specific rules. Activity that a HIPS watches for includes an event that attempts to control other programs, terminate programs, and install devices and drivers. When a HIPS blocks action, it then alerts the user so an appropriate decision about what to do can be made.

CAUTION One of the drawbacks to a HIPS is a high number of false positives can be generated. Both legitimate and malicious programs often access the same resource, and each can cause a HIPS to then block the action.

Endpoint Detection and Response (EDR) **Endpoint detection and response (EDR)** tools have a similar functionality to HIDS of monitoring endpoint events and of HIPS of taking immediate action. However, EDR tools are considered more robust than HIDS and HIPS. First, an EDR can aggregate data from multiple endpoint computers to a centralized database so that security professionals can further investigate and gain a better picture of events occurring across multiple endpoints instead of just on a single endpoint. This can help determine if an attack is more widespread across the enterprise and if more comprehensive and higher-level action needs to be taken. Second, EDR tools can perform more sophisticated analytics that identify patterns and detect anomalies. This can help detect unusual or unrecognized activities by performing baseline comparisons of normal behavior.

NOTE 16

Many EDRs also allow for a manual or user analysis of the data.

Harden Endpoints

After boot security has been established and the endpoints have been protected, the next step is to harden the endpoints for further protection. Hardening endpoints involves patch management and OS protections.

Patch Management

One of the most important steps in securing an endpoint computer is to promptly install patches. Threat actors often watch for the release of a patch and then immediately craft an attack around the vulnerability the patch addresses, knowing that many users and organizations are lax in applying patches.

Effective patch management involves two types of patch management tools to administer patches. The first type includes tools for patch distribution, while the second type involves patch reception.

Patch Distribution Modern operating systems—such as Red Hat Linux, Apple macOS, Ubuntu Linux, and Microsoft Windows—frequently distribute patches. A growing number of application and utility software developers are also distributing patches for their products (**third-party updates**).

These patches, however, can sometimes create new problems, such as preventing a custom application from running correctly. Organizations that have these types of applications usually test patches when they are released to ensure that they do not adversely affect any customized applications. In these instances, the organization delays the installation of a patch from the vendor's online update service until the patch is thoroughly tested. But how can an organization prevent its employees from installing the latest patch until it has passed testing and still ensure that all users download and install necessary patches?

The answer is an *automated patch update service*. This service is used to manage patches within the enterprise instead of relying upon the vendor's online update service. An automated patch update service typically consists of a component installed on one or more servers inside the corporate network. Because these servers can replicate information among themselves, usually only one of the servers must be connected to the vendor's online update service, as seen in Figure 4-5.

Advantages of using an automated patch update service include the following:

- Downloading patches from a local server instead of using the vendor's online update service can save bandwidth and time because each computer does not have to connect to an external server.
- Administrators can approve or decline updates for client systems, force updates to install by a specific date, and obtain reports on what updates each computer needs.
- Administrators can approve updates for "detection" only; this allows them to see which computers require the update without installing it.

Patch Reception Early versions of OSs allowed the user to configure how they receive patches. For example, prior to Windows 10, Microsoft users had several options regarding accepting or even rejecting patches. These options included *Install updates automatically*, *Download updates but let me choose whether to install them*, *Check for updates but let me choose whether to download and install them,* and *Never check for updates.* However, this approach frequently resulted in important security patches being ignored by users and putting their computers at risk.

Today users have fewer—if *any*—options regarding patches: usually patches are automatically downloaded and installed whenever they become available. This is called **auto-update**, and it ensures that the software is always up to date.

Microsoft Windows 10 is typical of the enhancements of patch reception. Figure 4-6 shows the Windows 10 Advanced options. These options include the following:

- *Forced updates*. Users can no longer refuse or indefinitely delay security updates. By default, all updates will be downloaded and installed automatically. However, users can defer the "quality updates" (those with security patches) but only for seven days (Windows 10 Home edition) or 35 days (all other versions). New feature updates (those without security patches) can be delayed for 35 days (Windows 10 Home edition) or 365 days (all other versions).

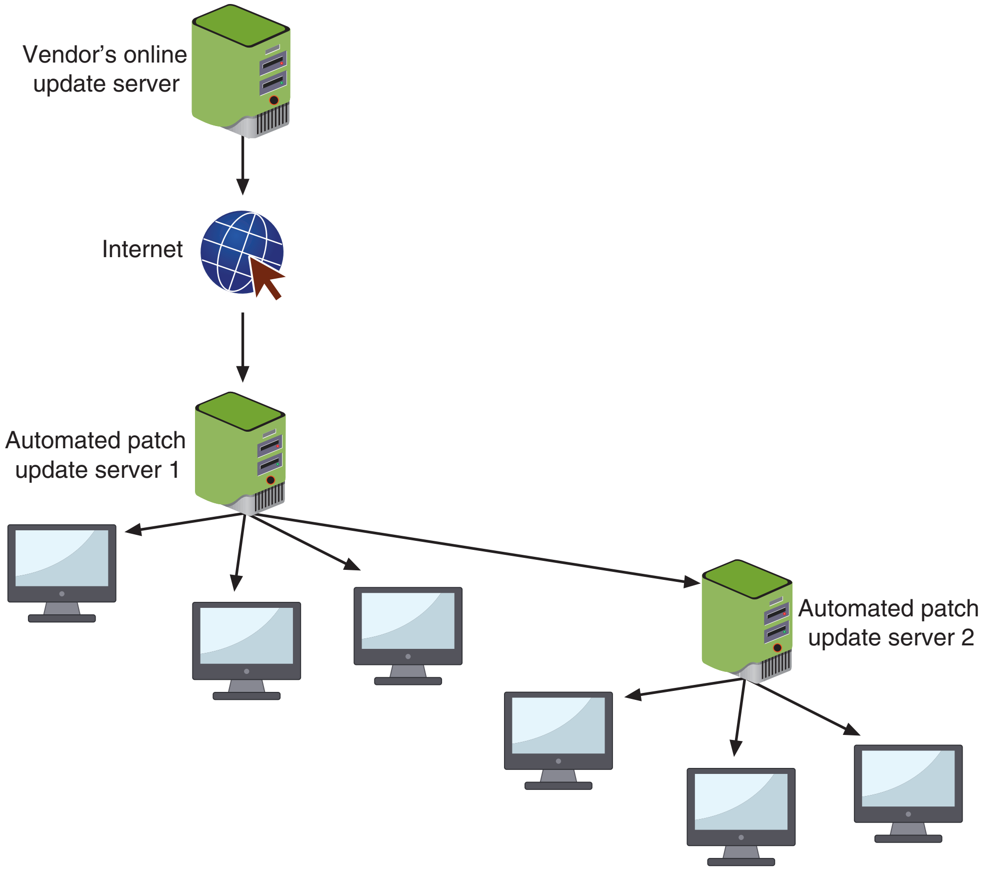

Vendor's online update server
Internet
Automated patch update server 1
Automated patch update server 2

Figure 4-5 Automated patch update service

- *No selective updates*. Unlike in previous versions of Windows, users cannot select individual Windows updates to download and install. However, users can select if they want to receive updates for other installed Microsoft products (such as Office).
- *More efficient distribution*. If many Windows 10 devices are connected to a network, each device does not have to download the updates over the Internet individually. Instead, once one device has downloaded the updates, they can then be distributed to the other devices across the local network. In addition, Windows will not download updates on mobile devices unless that device is connected to an unrestricted Wi-Fi network (so that it does not use the cellular data connections that users pay for).

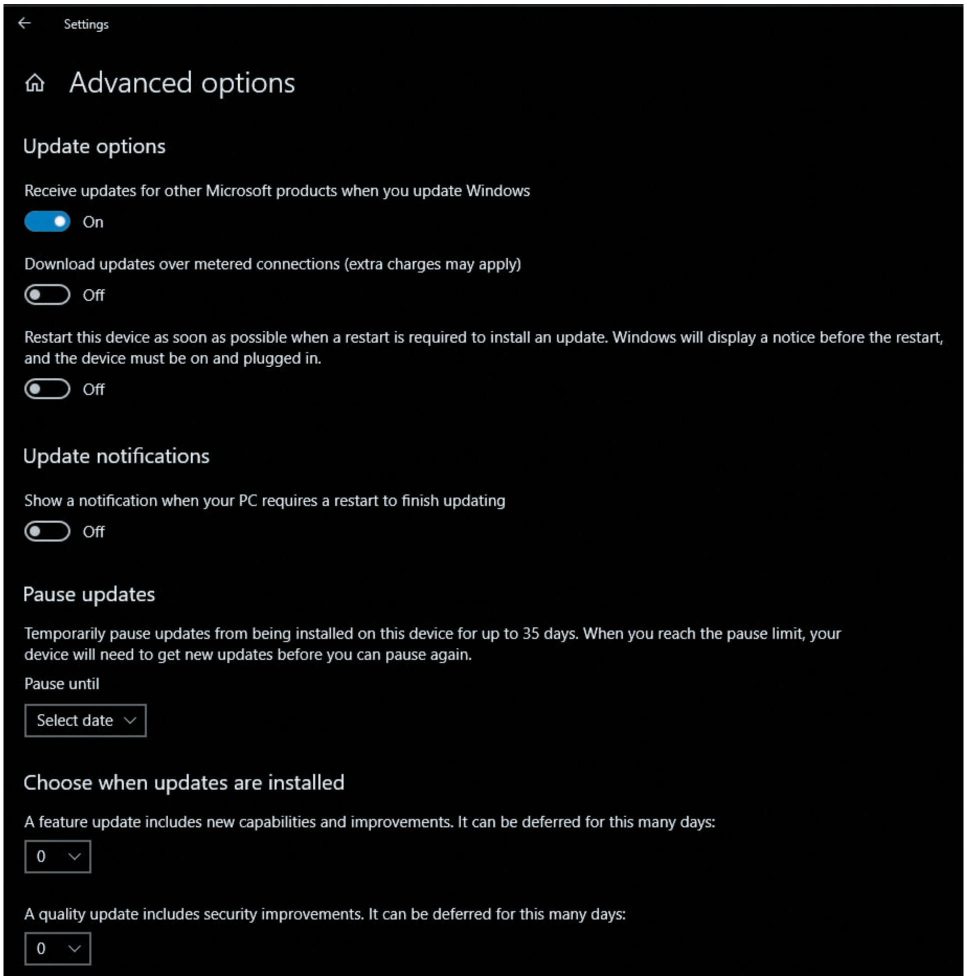
Operating Systems

There are different types and uses of operating systems. Several of the major types are listed in Table 4-4.

Although protections within the OS are designed to provide security for the endpoint device, the OS itself must be protected. Securing an OS involves proper security configurations and using confinement tools.

Security Configuration The security of an OS depends upon the proper configuration of its built-in security features. Modern operating systems have hundreds of security settings. A typical OS security configuration should include the following:

- *Disabling unnecessary ports and services*. One of the primary OS security configurations involves **disabling unnecessary open ports and services**, or "turning off" any service that is not being used, such as Microsoft Windows ASP.NET State Service, Portable Device Enumerator Service, and Apple macOS Spotlight Indexing. In addition, closing any unnecessary TCP ports can also enhance security.
- *Disabling default accounts/passwords.* Another important disabling function is disabling default accounts and passwords. Some OSs include unnecessary accounts. For example, Microsoft Windows 10 includes a *built-in Administrator account* that can be used for those building new computers to run programs and applications



Advanced options

Update options

Receive updates for other Microsoft products when you update Windows

On

Download updates over metered connections (extra charges may apply)

Off

Restart this device as soon as possible when a restart is required to install an update. Windows will display a notice before the restart, and the device must be on and plugged in.

Off

Update notifications

Show a notification when your PC requires a restart to finish updating

Off

Pause updates

Temporarily pause updates from being installed on this device for up to 35 days. When you reach the pause limit, your device will need to get new updates before you can pause again.

Pause until

Select date

Choose when updates are installed

A feature update includes new capabilities and improvements. It can be deferred for this many days:

0

A quality update includes security improvements. It can be deferred for this many days:

0

Source: Used with permissions from Microsoft

Figure 4-6 Microsoft Windows 10 Advanced options

Table 4-4 Types of OSs

OS type	Uses	Examples
Network OS	Software that runs on a network device like a firewall, router, or switch	Cisco Internetwork Operating System (IOS), Juniper JUNOS, MikroTik RouterOS
Server OS	Operating system software that runs on a network server to provide resources to network users	Microsoft Windows Server, Apple macOS Server, Red Hat Linux
Workstation OS	Software that manages hardware and software on a client computer	Microsoft Windows, Apple macOS, Ubuntu Linux
Appliance OS	OS in firmware that is designed to manage a specific device like a digital video recorder or video game console.	Linpus Linux
Kiosk OS	System and user interface software for an interactive kiosk	Microsoft Windows, Google Chrome OS, Apple iOS, Instant WebKiosk, KioWare (Android)
Mobile OS	Operating system for mobile phones, smartphones, tablets, and other handheld devices	Google Android, Apple iOS, Apple iPadOS

before a user account is created. In addition, some accounts may come with default passwords that should be changed.

- *Employing least functionality.* The concept of "least functionality" states a user should only be given the minimum set of permissions required to perform necessary tasks; all other permissions should be configured as not available to the user. For example, a user should not have the ability to modify system security features.

Instead of recreating the same security configuration on each endpoint computer, tools can be used to automate the process. In Microsoft Windows, a *security template* is a collection of security configuration settings. These settings typically include account policies, user rights, event log settings, restricted groups, system services, file permissions, and registry permissions. Once a single endpoint computer has been configured properly, a security template from that device can be developed and used for deploying to other systems. Predefined security templates are also available to be imported, and these settings then can be modified to create a unique security configuration for all endpoints.

NOTE 17

Although a Microsoft Windows security template can be deployed manually, this requires an administrator to access each computer and apply the security template either through using the command line or a *snap-in*, which is a software module that provides administrative capabilities for a device. A preferred method is to use *Group Policy*, which is a feature that provides centralized management and configuration of computers and remote users who are using specific Microsoft directory services known as *Active Directory (AD)*. Group Policy allows a single configuration to be set and then deployed to many or all users.

For a Microsoft Windows endpoint computer, it is also important to secure the **registry**, which is a database that contains low-level settings used by the Windows OS and for those applications that use the registry. Threat actors who can modify the registry could be able to disable antivirus and antimalware protections, disable any cloud-delivered protection, and remove security updates.

To mitigate this risk, the Windows 10 Tamper Protection security feature prevents Windows security settings from being changed or disabled by a threat actor who modifies the registry. Instead, the security settings can only be accessed directly through the Windows 10 user interface or through enterprise management software.

In addition to Tamper Protection, a Group Policy setting can prevent access to the tool that can alter the registry. This setting is *Prevent access to registry editing tools* and is shown in Figure 4-7.

NOTE 18

Tamper Protection also prevents changes to security settings by programs, Windows command line commands, or through Group Policy.

NOTE 19

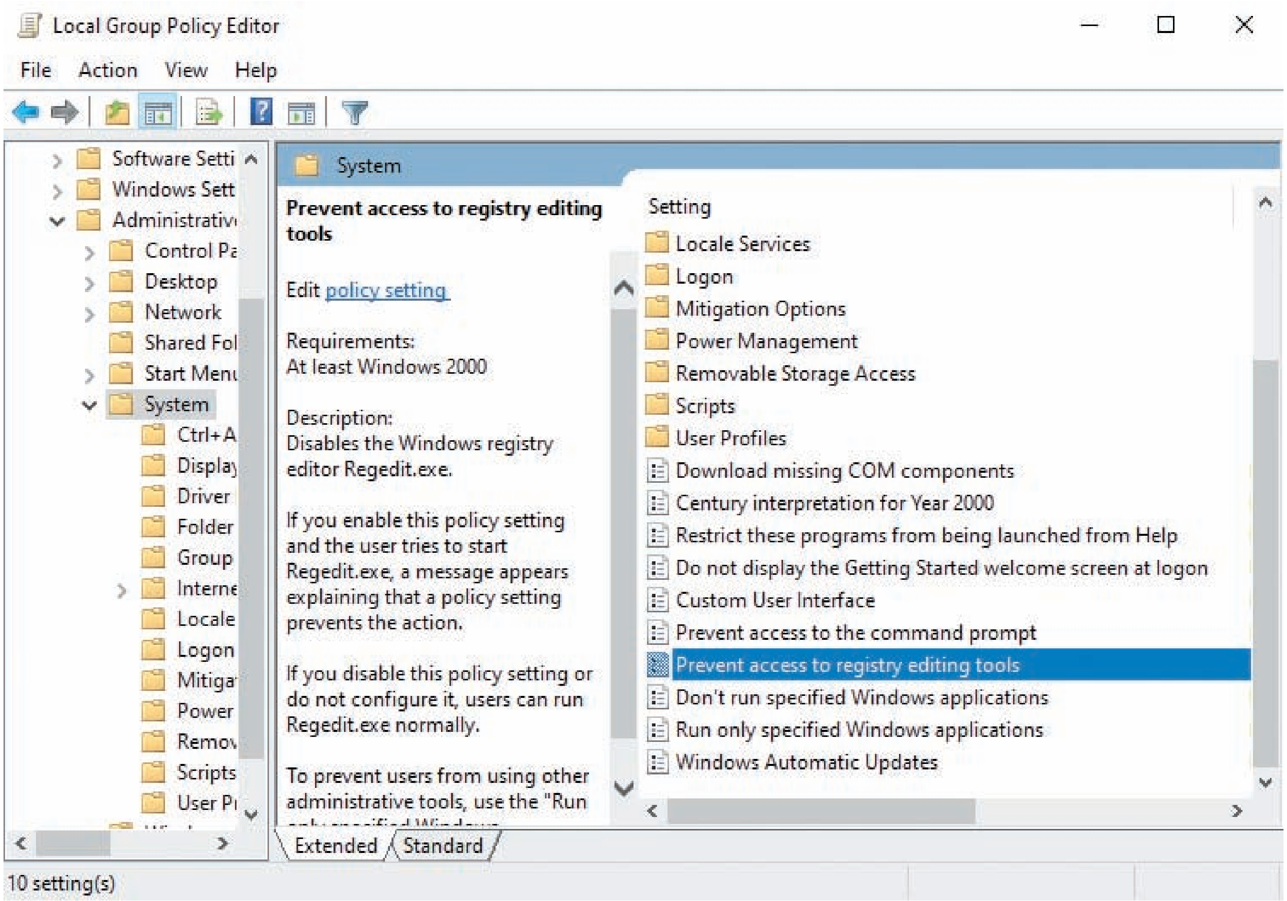
Instead of managing security options on an OS that has been deployed, in some cases, it is necessary to tighten security during the design and coding of the OS. This is called OS hardening. An OS that has been designed in this way to be secure is a *trusted OS*.

Confinement Tools Several tools can be used to "confine" or restrict malware. These tools include the following:

- *Application whitelisting/blacklisting.* An increasingly popular approach to client OS security is to employ **application whitelisting/blacklisting**. **Whitelisting** is approving in advance only specific applications to run on the OS so that any item not approved is either restricted or denied ("default-deny"). The inverse of whitelisting is **blacklisting**, creating a list of unapproved software so that any item not on the list of blacklisted applications can run ("default-allow"). Application whitelisting/blacklisting requires preapproval for an application to run or not run.

NOTE 20

The elite Tailored Access Operations (TAO) section of the National Security Agency (NSA) is responsible for compromising networks owned by hostile nations to spy on them. The head of the TAO spoke at a security conference about the best practices of security from the NSA's perspective (in his own words, "what can you do to defend yourself to make my life hard?"). One of the most important steps was to employ whitelisting for the software that runs on servers. A similar step is to whitelist a predefined set of websites to which users can connect to prevent malware from accessing a C&C or to exfiltrate stolen information.[3]


Local Group Policy Editor
File
Action
View
Help
Software Setti
Windows Sett
Administrativ
Control Pa
Desktop
Network
Shared Fol
Start Menu
System
Ctrl+A
Display
Driver
Folder
Group
Interne
Locale
Logon
Mitiga
Power
Remov
Scripts
User Pr
System
Setting
Locale Services
Logon
Mitigation Options
Power Management
Removable Storage Access
Scripts
User Profiles
Download missing COM components
Century interpretation for Year 2000
Restrict these programs from being launched from Help
Do not display the Getting Started welcome screen at logon
Custom User Interface
Prevent access to the command prompt
Prevent access to registry editing tools
Don't run specified Windows applications
Run only specified Windows applications
Windows Automatic Updates
Extended
Standard
10 setting(s)

Prevent access to registry editing tools

Edit policy setting

Requirements:
At least Windows 2000

Description:
Disables the Windows registry editor Regedit.exe.

If you enable this policy setting and the user tries to start Regedit.exe, a message appears explaining that a policy setting prevents the action.

If you disable this policy setting or do not configure it, users can run Regedit.exe normally.

To prevent users from using other administrative tools, use the "Run

Figure 4-7 Prevent access to registry editing tools

Source: Used with permissions from Microsoft

- *Sandbox*. Figure 4-8 illustrates a conceptual view of applications that interact with an OS. A **sandbox** is a "container" in which an application can be run so that it does not impact the underlying OS, as illustrated in Figure 4-9. Anything that occurs within the sandbox is not visible to other applications or the OS outside the sandbox. Also, the contents of the sandbox are not saved when the sandbox is closed. Sandboxes are often used when downloading or running suspicious programs to ensure that the endpoint will not become infected.
- *Quarantine*. Whereas a sandbox is used to contain an *application*, **quarantine** is a process that holds a suspicious *document*. Quarantine is most commonly used with email attachments. When an attachment is received, the quarantine process removes the attachment and, depending upon the policy set by the organization, either sends to the user sent a sanitized version of the attachment (such as a Word DOCX document that has been converted to a PDF document) or a URL to the document on a restricted computer so that the user can view, print, or delete the attachment.

NOTE 21

A sandbox is not the same as a virtual machine. A virtual machine is a "computer within a computer" in which an entire OS runs as an application on top of the regular OS. However, its contents can be saved for future use.

NOTE 22

Microsoft Office documents that are received as attachments, opened from an Internet location, or opened from an unsafe location are by default quarantined. The documents are displayed in Protected View, which is a read-only mode with most editing functions disabled. If the file needs to be saved or printed, the user can click the "Enable Editing" button to open the document as normal.

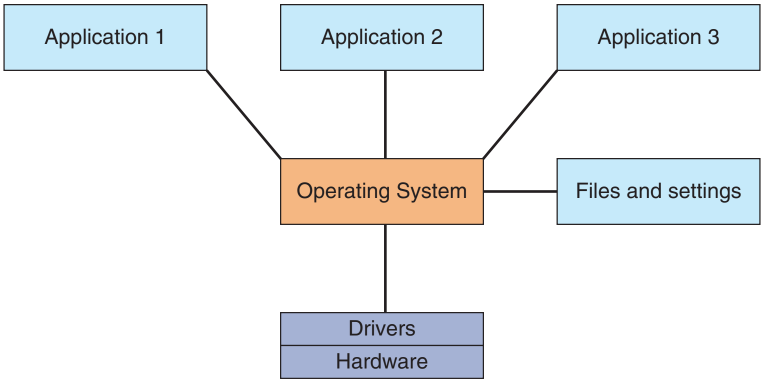


Figure 4-8 Applications interacting with an OS

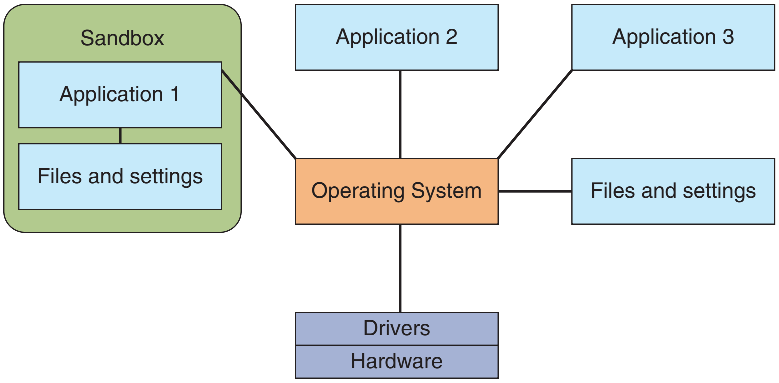


Figure 4-9 Using a sandbox

TWO RIGHTS & A WRONG

1. In a Trusted Boot, the endpoint's firmware logs the boot process to the OS can send it to a trusted server to assess the security.
2. Dynamic analysis uses heuristic monitoring.
3. Cookies are a workaround of the stateless protocol HTTP.

See Appendix B for the answer.

CREATING AND DEPLOYING SECDEVOPS

CERTIFICATION

2.3 Summarize secure application development, deployment, and automation concepts.

3.2 Given a scenario, implement host or application security solutions.

Confirming boot integrity, protecting endpoints, and hardening endpoints are all essential steps in securing an endpoint computer. But an additional element that is also critical is creating and deploying secure applications. Because endpoint computers run applications, the best endpoint boot security, antivirus and antimalware, patch management, and OS security configurations can all be negatively impacted—and sometimes negated—by an application that contains vulnerabilities. An unsecure application can open the door for attackers to exploit the application, the data that it uses, and even the underlying OS. Table 4-5 lists attacks that can be launched using vulnerabilities in applications.

Table 4-5 Attacks based on application vulnerabilities

Attack	Description	Defense
Executable files attack	Trick the vulnerable application into modifying or creating executable files on the system	Prevent the application from creating or modifying executable files for its proper function
System tampering	Use the vulnerable application to modify special sensitive areas of the operating system (Microsoft Windows registry keys, system startup files, etc.) and take advantage of those modifications	Do not allow applications to modify special areas of the OS
Process spawning control	Trick the vulnerable application into spawning executable files on the system	Take away the process spawning ability from the application

One particularly dangerous attack can be the result of a vulnerability in an application. The *root directory* is a specific directory on a web server's file system, and users who access the server are usually restricted to the root directory and directories and files beneath the root directory, but they cannot access other directories. For example, the default root directory of Microsoft's Internet Information Services (IIS) web server is *C:\Inetpub\wwwroot*. Users have access to this directory and subdirectories beneath this root (*C:\Inetpub\wwwroot\news*) if given permission, but they do not have access to other directories in the file system, such as *C:\Windows\System32*. A **directory traversal** attack takes advantage of vulnerability in the web application program or the web server software so that a user can move from the root directory to other restricted directories. The ability to move to another directory could allow an unauthorized user to view confidential files or even enter commands to execute on a server known as *command injection*. A directory traversal attack is illustrated in Figure 4-10. Other dangerous weaknesses in an application can create vulnerabilities in computer memory or buffer areas that can be easily exploited. These poor **memory management** vulnerabilities result in attacks such as buffer overflow, integer overflow, pointer/object deference, and DLL injection attacks.

NOTE 23

Buffer overflow, integer overflow, pointer/object dereference, and DLL injection attacks are all covered in Module 3.

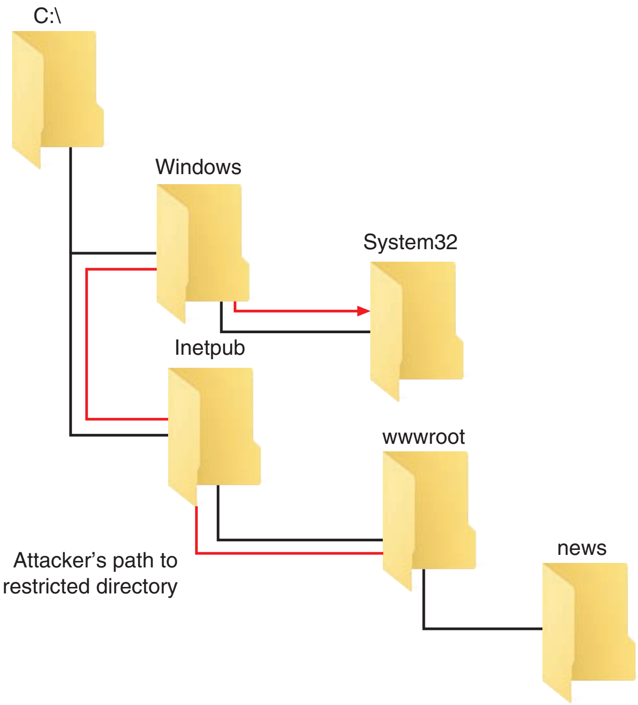


Figure 4-10 Directory traversal attack

The cause of most unsecure applications is usually the result of how the application was designed and written. Creating and developing secure software involves understanding application development concepts, secure coding techniques, and code testing.

Application Development Concepts

The two levels of application development concepts include general concepts that apply to all application development and those that apply to a more rigorous security-based approach.

General Concepts

Developing an application requires completing several stages. These stages include the following:

- *Development*. At the **development stage**, the requirements for the application are established, and it is confirmed that the application meets the intended business needs before the actual coding begins.
- *Testing*. The **testing stage** thoroughly tests the application for any errors that could result in a security vulnerability.
- *Staging*. The **staging stage** tests to verify that the code functions as intended.
- *Production*. In the **production stage** the application is released to be used in its actual setting.

Often application development will involve **software diversity**. Software diversity is a software development technique in which two or more functionally identical variants of a program are developed from the same specification but by different programmers or programming teams. The intent is to provide error detection, increased reliability, and additional documentation. It also can reduce the probability that errors created by different **compilers**, which are programs that create **binary** machine code from human source code, will influence the end results.

Another concept regarding application development involves how the completed application will be used in the context of the larger IT footprint of the enterprise. **Provisioning** is the enterprise-wide configuration, deployment, and management of multiple types of IT system resources, of which the new application would be viewed as a new resource. **Deprovisioning** in application development is removing a resource that is no longer needed.

Integrity measurement is an "attestation mechanism" designed to be able to convince a remote party (external to the coding team) that an application is running only a set of known and approved executables. Whenever a file is called in an executable mode, such as when a program is invoked or a sharable library is mapped, the integrity measurement tool generates a unique digital value of that file. On request, the tool can produce a list of all programs run and their corresponding digital values. This list can then be examined to ensure that no unknown or known vulnerable applications have been run.

SecDevOps

An *application development lifecycle model* is a conceptual model that describes the stages involved in creating an application. Most projects use one of two major application development lifecycle models.

The *waterfall model* uses a sequential design process: as each stage is fully completed, the developers move on to the next stage. This means that once a stage is finished, developers cannot go back to a previous stage without starting all over again. For example, in the waterfall model, **quality assurance (QA)**—verification of quality—occurs only after the application has been tested and before it is finally placed in production. However, this makes any issues uncovered by QA difficult to address since it is at the end of the process. The waterfall model demands extensive planning in the very beginning and requires that it be followed carefully.

The *agile model* was designed to overcome the disadvantages of the waterfall model. Instead of following a rigid sequential design process, the agile model takes an incremental approach. Developers might start with a simplistic project design and begin to work on small modules. The work on these modules is done in short (weekly or monthly) "sprints," and at the end of each sprint, the project's priorities are again evaluated as tests are being run. This approach allows for software issues to be incrementally discovered so that feedback and changes can be incorporated into the design before the next sprint is started.

One specific type of software methodology that follows the agile model and heavily incorporates **secure coding practices and techniques** to create secure software applications is called *SecDevOps*. SecDevOps (also known as DevSecOps and DevOpsSec) is the process of integrating secure development best practices and methodologies into application software development and deployment processes using the agile model. It is a set of best practices designed to help organizations implant secure coding deep in the heart of their applications.

SecDevOps is often promoted in terms of its **elasticity** (flexibility or resilience in code development) and its **scalability** (expandability from small projects to very large projects). However, the cornerstone of SecDevOps is automation. With standard application development, security teams often find themselves stuck with time-consuming manual tasks. SecDevOps, on the other hand, applies what is called **automated courses of action** to develop the code as quickly and securely as possible. This automation enables **continuous monitoring** (examining the processes in real time instead of at the end of a stage), **continuous validation** (ongoing approvals of the code), **continuous integration** (ensuring that security features are incorporated at each stage), **continuous delivery** (moving the code to each stage as it is completed), and **continuous deployment** (continual code implementation).

NOTE 24

The SecDevOps methodology also includes concepts such as *immutable systems* (once a value or configuration is employed as part of an application, it is not modified; if changes are necessary, a new system must be created), *infrastructure as code* (managing a hardware and software infrastructure using the same principles as developing computer code), and *baselining* (creating a starting point for comparison purposes in order to apply targets and goals to measure success).

Table 4-6 lists sources of recommendations for SecDevOps.

Table 4-6 Secure SDLC sources

Source	Description	Materials available
OWASP (Open Web Application Security Project)	A group that monitors web attacks	Maturity models, development guides, testing guides, code review guides, and application security verification standards
SANS (SysAdmin, Audit, Network and Security Institute)	A company that specializes in cybersecurity and secure web application development	White papers, research reports, and best practices guidelines
CIS (Center for Internet Security)	Not-for-profit organization that compiles CIS security controls	Training, assessment tools, and consulting services

Because SecDevOps is based on the agile method, it involves continuous modifications throughout the process. With these continual changes, it is important to use tools that support change management or creating a plan for documenting changes to the application. One tool for change management is **version control** software that allows changes to be automatically recorded and, if necessary, "rolled back" to a previous version of the software.

Secure Coding Techniques

Several coding techniques should be used to create secure applications and limit **data exposure** or disclosing sensitive data to attackers. These techniques include determining how encryption will be implemented and ensuring that memory management is handled correctly so as not to introduce memory vulnerabilities. Other techniques are summarized in Table 4-7.

Code Testing

Testing is one of the most important steps in SecDevOps. Instead of testing only after the application is completed, testing should be performed much earlier during the implementation and verification phases of a software development process. Testing involves static code analysis and dynamic code analysis.

NOTE 25

Edsger W. Dijkstra, a famous software engineer, once said, "Program testing can be used to show the presence of bugs, but never to show their absence!"

Table 4-7 Secure coding techniques

Coding technique	Description	Security advantage
Proper input validation	Accounting for errors such as incorrect user input (entering a file name for a file that does not exist).	Can prevent Cross-site scripting (XSS) and Cross-site request forgery (CSRF) attacks
Normalization	Organizing data within a database to minimize redundancy.	Reduces footprint of data exposed to attackers
Stored procedure	A subroutine available to applications that access a relational database.	Eliminates the need to write a subroutine that could have vulnerabilities
Code signing	Digitally signing applications.	Confirms the software author and guarantees the code has not been altered or corrupted
Obfuscation/ camouflaged code	Writing an application in such a way that its inner functionality is difficult for an outsider to understand.	Helps prevent an attacker from understanding a program's function
Dead code	A section of an application that executes but performs no meaningful function.	Provides an unnecessary attack vector for attackers
Server-side execution and validation or **Client-side execution and validation**	Input validation generally uses the server to perform validation but can also have the client perform validation by the user's web browser.	Adds another validation to the process
Code reuse of third-party libraries and SDKs	Code reuse is using existing software in a new application; a software development kit (SDK) is a set of tools used to write applications.	Existing libraries that have already been vetted as secure eliminate the need to write new code

Static Code Analysis

Analysis and testing of the software should occur from a security perspective before the source code is even compiled. These tests are called **static code analysis**. Figure 4-11 illustrates an automated static code analysis tool.

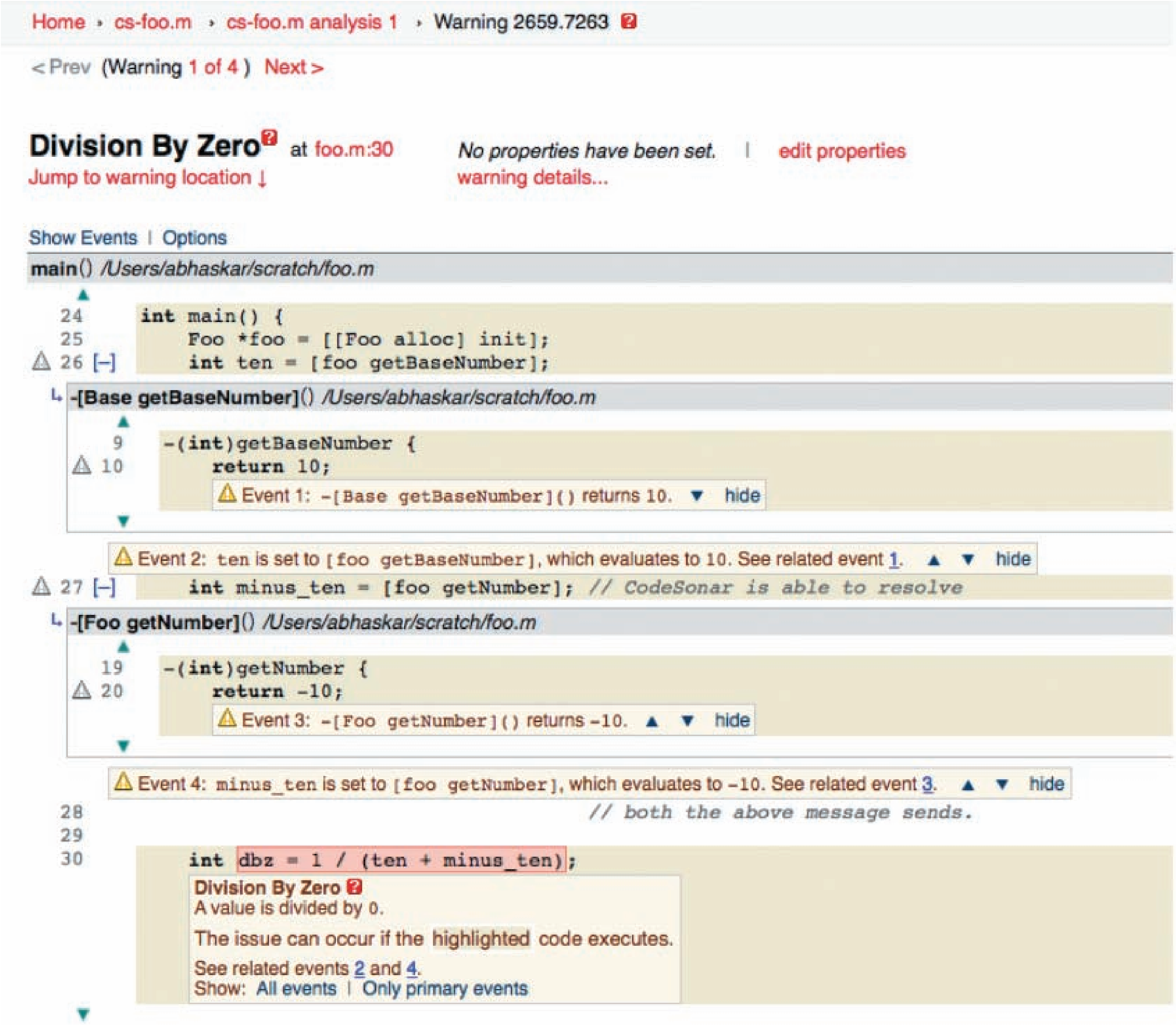
Automated static code analysis may also be accompanied by **manual peer reviews**. In these reviews, software engineers and developers are paired together or grouped in larger teams to laboriously examine each line of source code, looking for vulnerabilities.

Dynamic Code Analysis

Security testing should also be performed after the source code is compiled (a process called **dynamic code analysis** or *run-time verification*) and when all components are integrated and running. This testing typically uses a tool or suite of pre-built attacks or testing tools that specifically monitor the application's behavior for memory corruption, user privilege issues, and other critical security problems.

Some of the most common dynamic code analysis tools use a process called **fuzzing**. Fuzzing provides random input to a program in an attempt to trigger exceptions, such as memory corruption, program crashes, or security breaches. An advantage of fuzzing is that it produces a record of what input triggered the exception so it can be reproduced to track down the problem within the code. Fuzzing test software consists of an *execution engine* and an *input generator*, which usually allows the tester to configure the types of inputs (see Figure 4-12).

CAUTION A single pass of a fuzzer is unlikely to find all exceptions in software due to the randomness in the fuzzing process. The mutation of the inputs relies on randomness to determine where to mutate input and what to mutate. Fuzzers require multiple trials and statistical tests.



Home › cs-foo.m › cs-foo.m analysis 1 › Warning 2659.7263

< Prev (Warning 1 of 4) Next >

Division By Zero at foo.m:30 | No properties have been set. | edit properties

Jump to warning location ↓ | warning details...

Show Events | Options

main() /Users/abhaskar/scratch/foo.m

```
int main() {
    Foo *foo = [[Foo alloc] init];
    int ten = [foo getBaseNumber];
```

-[Base getBaseNumber]() /Users/abhaskar/scratch/foo.m

```
-(int)getBaseNumber {
    return 10;
```

⚠ Event 1: -[Base getBaseNumber]() returns 10. ▼ hide

⚠ Event 2: ten is set to [foo getBaseNumber], which evaluates to 10. See related event 1. ▲ ▼ hide

```
    int minus_ten = [foo getNumber]; // CodeSonar is able to resolve
```

-[Foo getNumber]() /Users/abhaskar/scratch/foo.m

```
-(int)getNumber {
    return -10;
```

⚠ Event 3: -[Foo getNumber]() returns -10. ▲ ▼ hide

⚠ Event 4: minus_ten is set to [foo getNumber], which evaluates to -10. See related event 3. ▲ ▼ hide

```
                                     // both the above message sends.


    int dbz = 1 / (ten + minus_ten);
```

Division By Zero
A value is divided by 0.

The issue can occur if the highlighted code executes.

See related events 2 and 4.
Show: All events | Only primary events

Source: GrammaTech

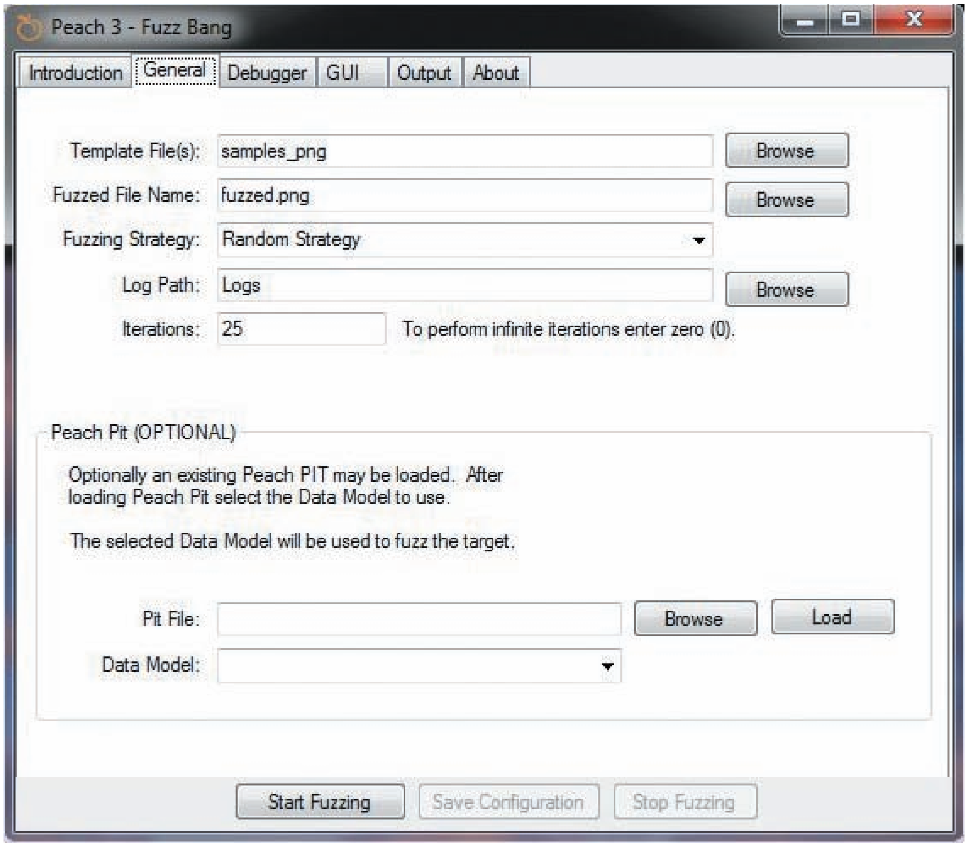
Figure 4-11 Automated static code analysis tool

TWO RIGHTS & A WRONG

1. A goal of software diversity is to reduce the probability that errors created by different compilers will influence the end results.
2. Provisioning is removing a resource that is no longer needed.
3. SecDevOps has elasticity and scalability.

See Appendix B for the answer.

VM LAB You're now ready to complete the live virtual machine labs for this module. The labs can be found in each module in the MindTap.



Template File(s): samples_png

Fuzzed File Name: fuzzed.png

Fuzzing Strategy: Random Strategy

Log Path: Logs

Iterations: 25 To perform infinite iterations enter zero (0).

Peach Pit (OPTIONAL)

Optionally an existing Peach PIT may be loaded. After loading Peach Pit select the Data Model to use.

The selected Data Model will be used to fuzz the target.

Pit File:

Data Model:

Source: Déjà vu Software

Figure 4-12 Fuzzer input generator

SUMMARY

- Organizations are pooling their experiences and knowledge gained about the latest attacks with the broader security community because sharing this type of information has become an important aid to help other organizations shore up their defenses. Open source threat intelligence information that is freely available, often called open source intelligence (OSINT), has become a vital resource. This information is often collected and then disseminated through public information sharing centers. The two concerns around public information sharing centers are the privacy of shared information and the speed at which the information is shared. Closed source is the opposite of open source. Organizations that are participants in closed source information are part of private information sharing centers that restrict both access to data and participation.
- Several sources of threat intelligence are useful. A vulnerability database is a repository of known vulnerabilities and information as to how they have been exploited. These databases create “feeds” of the latest cybersecurity incidences. A cybersecurity threat map illustrates cyberthreats overlaid on a diagrammatic representation of a geographical area. Threat maps help in visualizing attacks and provide a limited amount of context of the source and the target countries, the attack types, and historical and near real-time data about threats, although they provide limited valuable information. File and code repositories are used by victims of an attack who can upload malicious files and software code that can then be examined by others to learn more about these attacks and craft their defenses. The dark web is the domain of threat actors and beyond the reach of a normal search engine. Malicious activity such as selling drugs and stolen personal information and buying and selling malicious software used for attacks occurs on the dark web. Some security professionals and organizations use the dark web on a limited basis to look for signs that information critical to that enterprise is being sought out or sold on the dark web.

- One of the steps that is often overlooked in securing endpoint computers is to confirm that the computer has started without any malicious activity taking place. The booting process on early personal computers used firmware called the BIOS (Basic Input/Output System). Although the ability to update the BIOS firmware enabled new features to be added, it also opened the door for threat actors to create malware to infect the BIOS. To combat this vulnerability and add functionality, UEFI (Unified Extensible Firmware Interface) was developed, in conjunction with the Secure Boot security standard. Several techniques can be used to assure boot security by taking advantage of the features in UEFI.
- Antivirus (AV) software can examine a computer for any file-based virus infections and monitor computer activity and scan new documents that might contain a virus. A new approach to AV is dynamic analysis, which uses a variety of techniques to spot the characteristics of a virus instead of attempting to make matches. Antimalware is a suite of software intended to provide protections against multiple types of malware, such as ransomware, cryptomalware, Trojans, spam, and spyware.
- Web browsers have a degree of security that can protect endpoint computers. User-specific information stored in a file on the user's local computer is called a cookie. There are several types of cookies. As a means of protection for cookies, a web browser can send a secure cookie. This cookie is only sent to the server with an encrypted request over the secure HTTPS protocol. When a user visits a website through their web browser, the web server answers back with HTTP Response Headers. These headers tell the browser how to behave while communicating with the website. Several HTTP Response Headers can improve security.
- A host intrusion detection system (HIDS) is a software-based application that runs on an endpoint computer and can detect that an attack has occurred. The primary function of a HIDS is automated detection, which saves someone from sorting through log files to find an indication of unusual behavior. A HIDS relies on agents installed directly on the endpoint, and these agents work closely with the OS to observe activity. A host intrusion prevention system (HIPS) monitors endpoint activity to immediately react to block a malicious attack by following specific rules. Activity that a HIPS watches for includes an event that attempts to control other programs, terminate programs, and install devices and drivers. Endpoint detection and response (EDR) tools have a similar functionality to HIDS of monitoring endpoint events and of HIPS of taking immediate action. However, EDR tools are considered more robust than HIDS and HIPS.
- One of the most important steps in securing an endpoint computer is to promptly install patches. Modern operating systems—such as Red Hat Linux, Apple macOS, Ubuntu Linux, and Microsoft Windows—frequently distribute patches. A growing number of application and utility software developers are also distributing patches called third-party updates. An automated patch update service is used to manage patches within the enterprise instead of relying upon the vendor's online update service. Early versions of OSs allowed users to configure how they receive patches; however, today patches are usually automatically downloaded and installed whenever they become available.
- An unsecure application can open the door for attackers to exploit the application, the data that it uses, and even the underlying OS. The cause of most unsecure applications is usually the result of how the application was designed and written. Developing an application requires several stages. An application development lifecycle model is a conceptual model that describes the stages involved in creating an application. The waterfall model uses a sequential design process: as each stage is fully completed, the developers move on to the next stage. The agile model was designed to overcome the disadvantages of the waterfall model. Instead of following a rigid sequential design process, the agile model takes an incremental approach. One specific type of software methodology that follows the agile model and heavily incorporates secure coding practices and techniques to create secure software applications is called SecDevOps. Several coding techniques should be used to create secure applications and limit data exposure or disclosing sensitive data to attackers. These techniques include determining how encryption will be implemented and ensuring that memory management is handled correctly so as to not introduce memory vulnerabilities.
- Testing is one of the most important steps in SecDevOps. Yet, instead of testing only after the application is completed, testing should be performed much earlier during the implementation and verification phases of a software development process. Analysis and testing of the software should occur from a security perspective before the source code is even compiled. These tests are called static code analysis. Security testing should also be performed after the source code is compiled (a process called dynamic code analysis).

Key Terms

antimalware
antivirus (AV)
application whitelisting/blacklisting
automated courses of action
Automated Indicator Sharing (AIS)
auto-update
binary
blacklisting
boot attestation
client-side execution and validation
closed source
code reuse of third-party libraries and SDKs
code signing
compilers
continuous delivery
continuous deployment
continuous integration
continuous monitoring
continuous validation
dark web
data exposure
dead code
deprovisioning
development stage
directory traversal
disabling unnecessary open ports and services
dynamic code analysis
elasticity
endpoint detection and response (EDR)
file and code repositories
fuzzing
hardware root of trust
host intrusion detection system (HIDS)
host intrusion prevention system (HIPS)
HTTP Response Headers
indicator of compromise (IOC)
integrity measurement
manual peer reviews
Measured Boot
memory management
normalization
obfuscation/camouflaged code
open source
OWASP (Open Web Application Security Project)
predictive analysis
private information sharing centers
production stage
proper input validation
provisioning
public information sharing centers
quality assurance (QA)
quarantine
registry
sandbox
scalability
secure coding practices and techniques
secure cookie
server-side execution and validation
software diversity
staging stage
static code analysis
stored procedure
Structured Threat Information Expression (STIX)
testing stage
third-party updates
threat map
Trusted Automated Exchange of Intelligence Information (TAXII)
UEFI (Unified Extensible Firmware Interface)
version control
vulnerability database
whitelisting

Review Questions

1. An IOC occurs when what metric exceeds its normal bounds?
   a. IRR
   b. LRG
   c. EXR
   d. KRI
2. What are the two concerns about using public information sharing centers?
   a. Cost and availability
   b. Privacy and speed
   c. Security and privacy
   d. Regulatory approval and sharing
3. Which privacy protection uses four colors to indicate the expected sharing limitations that are to be applied by recipients of the information?
   a. CISA
   b. FOIA
   c. TLP
   d. PCII
4. Oskar has been receiving emails about critical threat intelligence information from a public information sharing center. His team leader has asked him to look into how the process can be automated so that the information can feed directly into the team's technology security. What technology will Oskar recommend?
   a. Automated Indicator Sharing (AIS)
   b. Bidirectional Security Protocol (BSP)
   c. Linefeed Access
   d. Lightwire JSON Control
5. Which of the following is an application protocol for exchanging cyberthreat intelligence over HTTPS?
   a. STIX
   b. AIP-TAR

- **c.** TAXII
- **d.** TCP-Over-Secure (ToP)

6. What are the two limitations of private information sharing centers?
- **a.** Access to data and participation
- **b.** Government approval and cost
- **c.** Timing of reports and remote access
- **d.** Bandwidth and CPU

7. Which of the following is NOT a limitation of a threat map?
- **a.** Many maps claim that they show data in real time, but most are simply a playback of previous attacks.
- **b.** Because threat maps show anonymized data, it is impossible to know the identity of the attackers or the victims.
- **c.** They can be difficult to visualize.
- **d.** Threat actors usually mask their real locations, so what is displayed on a threat map is incorrect.

8. Luka has been asked by his supervisor to monitor the dark web for any IOCs concerning their organization. The next week, Luca reports that he was unable to find anything because looking for information on the dark web is different from using the regular web. Which of the following is FALSE about looking for information on the dark web?
- **a.** It is necessary to use Tor or IP2.
- **b.** Dark web search engines are identical to regular search engines.
- **c.** Dark web merchants open and close their sites without warning.
- **d.** The naming structure is different on the dark web.

9. Which of the following is NOT an improvement of UEFI over BIOS?
- **a.** Stronger boot security
- **b.** Networking functionality in UEFI
- **c.** Access larger hard drives
- **d.** Support of USB 3.0

10. Which boot security mode sends information on the boot process to a remote server?
- **a.** UEFI Native Mode
- **b.** Secure Boot
- **c.** Trusted Boot
- **d.** Measured Boot

11. Which of the following is NOT an important OS security configuration?
- **a.** Employing least functionality
- **b.** Disabling default accounts
- **c.** Disabling unnecessary services
- **d.** Restricting patch management

12. Which stage conducts a test that will verify the code functions as intended?
- **a.** Production stage
- **b.** Testing stage
- **c.** Staging stage
- **d.** Development stage

13. Which model uses a sequential design process?
- **a.** Secure model
- **b.** Agile model
- **c.** Rigid model
- **d.** Waterfall model

14. Which of the following is NOT an advantage of an automated patch update service?
- **a.** Downloading patches from a local server instead of using the vendor's online update service can save bandwidth and time because each computer does not have to connect to an external server.
- **b.** Administrators can approve updates for "detection" only; this allows them to see which computers require the update without installing it.
- **c.** Users can disable or circumvent updates just as they can if their computer is configured to use the vendor's online update service.
- **d.** Administrators can approve or decline updates for client systems, force updates to install by a specific date, and obtain reports on what updates each computer needs.

15. What type of analysis is heuristic monitoring based on?
- **a.** Dynamic analysis
- **b.** Static analysis
- **c.** Code analysis
- **d.** Input analysis

16. Which of these is a list of preapproved applications?
- **a.** Greenlist
- **b.** Redlist
- **c.** Blacklist
- **d.** Whitelist

17. What is the advantage of a secure cookie?
- **a.** It cannot be stored on the local computer without the user's express permission.
- **b.** It is sent to the server over HTTPS.
- **c.** It is analyzed by AV before it is transmitted.
- **d.** It only exists in RAM and is deleted once the web browser is closed.

18. Which of the following tries to detect and stop an attack?
- **a.** HIDS
- **b.** HIPS
- **c.** RDE
- **d.** SOMA

19. What does Windows 10 Tamper Protection do?

a. Limits access to the registry.

b. Prevents any updates to the registry until the user approves the update.

c. Compresses and locks the registry.

d. Creates a secure backup copy of the registry.

20. Which of the following is FALSE about a quarantine process?

a. It holds a suspicious application until the user gives approval.

b. It can send a sanitized version of the attachment.

c. It can send a URL to the document that is on a restricted computer.

d. It is most often used with email attachments.

Hands-On Projects

CAUTION If you are concerned about installing any of the software in these projects on your regular computer, you can instead install the software in the Windows in the Microsoft Sandbox or a virtual machine created in the Module 1 Hands-On Projects. Software installed within the virtual machine will not impact the host computer.

Project 4-1: Using the Microsoft Online Security Bulletins

Time Required: 20 minutes

Objective: Explain different threat actors, vectors, and intelligence sources.

Description: Microsoft has made its security bulletins available in a searchable online database. All security professionals need to be familiar with using this database. In this project, you will explore the online database.

1. Open your web browser and enter the URL **portal.msrc.microsoft.com/en-us/**. (The location of content on the Internet may change without warning. If you are no longer able to access the program through this URL, use a search engine to search for "Microsoft Security Response Center.")
2. Click **Read the Security Update Guide FAQ**.
3. Click **Expand all** to read through the information.
4. Click the link **www.icasi.org/cvrf/** (or enter it into another tab in your browser). What is the Common Vulnerability Reporting Framework (CVRF)? How is it used?
5. Return to the Microsoft Security Update Guide and then the MSRC main page.
6. Click the **Go to the Security Update Guide** button.
7. If no security updates appear, adjust the **From** date to the first day of the previous month.
8. Scroll through the list of security updates.
9. Click the first link under **Article**.
10. Read through this information.
11. Now return to the previous page and select another article to read.
12. How useful is this information? Is it presented in a format that is helpful?
13. Now click the CVE link under **Details** and read this information. Note the detail of this information.
14. Read the information under **Exploitability Assessment** (if the exploit you selected does not list an Exploitability Assessment, then select another that does include the assessment). What does this mean? Open another tab on your web browser, and search for **Microsoft Exploitability Index**. Read through the description that you find and keep this tab open.
15. Return to the Microsoft Security Update Guide and view the **Exploitability Assessment**. How serious is this security vulnerability?
16. How important is this information to a security professional? How easy is this online database to use?
17. Now compare the Microsoft database with Apple's. Enter the URL **support.apple.com/en-us/HT201222**. (The location of content on the Internet may change without warning. If you are no longer able to access the program through the above URL, use a search engine to search for "Apple Security Updates.")
18. Scroll down through the list of Apple security updates. How does this list compare with the updates from Microsoft?
19. Select a recent event under **Name and information link**.

20. Read the information about the update. How does this information compare with Microsoft's information? Why is there such a difference? Which provides better information for security professions?
21. Close all windows.

Project 4-2: Setting Windows Local Security Policy

Time Required: 20 minutes

Objective: Given a scenario, implement host or application security solutions.

Description: The Local Group Policy Editor is a Microsoft Management Console (MMC) snap-in that gives a single user interface through which all the Computer Configuration and User Configuration settings of Local Group Policy objects can be managed. The Local Security Policy settings are among the security settings contained in the Local Group Policy Editor. An administrator can use these to set policies that are applied to the computer. In this project, you will view and change local security policy settings.

CAUTION You will need to be an administrator to open the Local Group Policy Editor.

1. Click **Start**.
2. Type **secpol.msc** into the Search box, and then click **secpol**.

NOTE 26

If your computer is already joined to a domain then searching for secpol.msc might not launch the application. If this is the case, click **Start** and type **mmc.msc**. On the File menu, click **Add/Remove** snap-in, and then click **Add**. In **Add Standalone Snap-in**, double-click **Group Policy Object Editor**.

3. First create a policy regarding passwords. Expand **Account Policies** in the left pane, and then expand **Password Policy**.
4. Double-click **Enforce password history** in the right pane. This setting defines how many previously used passwords Windows will record. This prevents users from "recycling" old passwords.
5. Change **passwords remembered** to **4**.
6. Click **OK**.
7. Double-click **Maximum password age** in the right pane. The default value is 42, meaning that a user must change his password after 42 days.
8. Change **days** to **30**.
9. Click **OK**.
10. Double-click **Minimum password length** in the right pane. The default value is a length of eight characters.
11. Change **characters** to **10**.
12. Click **OK**.
13. Double-click **Password must meet complexity requirements** in the right pane. This setting forces a password to include at least two opposite case letters, a number, and a special character (such as a punctuation mark).
14. Click **Enabled**.
15. Click **OK**.
16. Double-click **Store passwords using reversible encryption** in the right pane. Because passwords should be stored in an encrypted format, this setting should not be enabled.
17. If necessary, click **Disabled**.
18. Click **OK**.
19. In the left pane, click **Account lockout policy**.
20. Double-click **Account lockout threshold** in the right pane. This is the number of times that a user can enter an incorrect password before Windows will lock the account from being accessed. (This prevents an attacker from attempting to guess the password with unlimited attempts.)
21. Change **invalid login attempts** to **5**.
22. Click **OK**.
23. Note that the Local Security Policy suggests changing the **Account lockout duration** and the **Reset account lockout counter after** values to 30 minutes.

24. Click **OK**.
25. Expand **Local Policies** in the left pane, and then click **Audit Policy**.
26. Double-click **Audit account logon events**.
27. Check both **Success** and **Failure**.
28. Click **OK**.
29. Right-click **Security Settings** in the left pane.
30. Click **Reload** to have these policies applied.
31. Close all windows.

Project 4-3: Configuring Microsoft Windows Security—Part 1

Time Required: 15 minutes

Objective: Given a scenario, implement host or application security solutions.

Description: It is important that security settings be properly configured on a computer in order to protect it. In this project, you examine several security settings on a Microsoft Windows 10 computer using the Windows interface.

CAUTION This project shows how to configure Windows security for a personal computer. If this computer is part of a computer lab or office, these settings should not be changed without the proper permissions.

1. Click **Start** and **Settings**.
2. Click **Update and security**.
3. If necessary, click **Windows Update** in the left pane.
4. Click **Pause updates for 7 days**. What warning are you given?
5. Click **Resume updates**.
6. Click **View update history**.
7. Expand each area and select one update to review. Read through the information on the update. How detailed is this information?
8. Return to the **View update history** page.
9. Return to the **Windows Update** page.
10. Click **Advanced options**.
11. Be sure that **Receive updates for other Microsoft products when you update Windows** is set to **On**. This will allow for updates for Microsoft software such as Office to also be updated.
12. Read the information under **Pause updates**. Why would you select this option?
13. Click the down arrow under **A quality update includes security improvements. It can be deferred for this many days:**. How many days can you defer security updates?
14. Return to the **Windows Update** page.
15. In the left pane, click **Windows Security**.
16. Click **Virus & threat protection**.
17. Click **Scan options** and be sure that **Quick scan** is selected.
18. Now perform a Quick scan of the most essential files. Click **Scan now**. Depending upon your system, it may take several minutes to complete. What was the result of the scan?
19. Return to the Virus & threat protection page. Under **Virus & threat protection settings**, click **Manage settings**.
20. Read through the details of the options. Are there any that you would change? Why?
21. Close all windows.

Project 4-4: Configuring Microsoft Windows Security—Part 2

Time Required: 15 minutes

Objective: Given a scenario, implement host or application security solutions.

Description: As seen from Project 4-3, Windows security settings are found across several different screens. This can make it easy to overlook important settings and time consuming to fine-tune the settings, especially when configuring the Windows Defender virus and threat protection product. A third-party tool called ConfigureDefender provides an easier interface. In this project, you will download and use the ConfigureDefender product.

NOTE 27

ConfigureDefender is not installed on the computer but runs as a stand-alone application.

1. Open your web browser and enter the URL **github.com/AndyFul/ConfigureDefender** (if you are no longer able to access the program through the URL, use a search engine to search for "ConfigureDefender").
2. Find the latest version of ConfigureDefender (the program is compressed in a ZIP file). Click the filename.
3. Click **Download**.
4. After the file has downloaded, unpack it, and then launch the program.
5. Click the **Info about Defender** button to see the computer's Defender settings. When finished, close the window.
6. Click the **Defender Security Log** button. Read the log file about recent actions. Does anything surprise you? When finished, close the window.
7. Scroll down and read the different settings. Were you aware that there were so many different options for Windows Defender?
8. Hover over the **DEFAULT** button and read the information.
9. Now hover over the **HIGH** button and read the information.
10. Click the **HIGH** button, and then close the pop-up box.
11. Scroll down through the settings. How much stronger are they than from the Default settings?
12. Now hover over the **MAX** button and read the information.
13. Click the **MAX** button, and then close the pop-up box.
14. Scroll down through the settings. How much stronger are they?
15. Finally click either the **DEFAULT** or **HIGH** button to set your computer at the security level that you choose.
16. How easy is ConfigureDefender to use? Would you recommend it to others?
17. Close all windows.

Case Projects

Case Project 4-1: AV Comparison

Select four antivirus products, one of which is a free product, and compare their features. Create a table that lists the features. How do they compare with the AV software you currently use? Which would you recommend to others? Why? Create a report on your research.

Case Project 4-2: Threat Maps

Locate four online cybersecurity threat maps. Compare their real-time results. Why are these results different? What type of information do they provide? How easy or hard are they to use? How could they be used? How should they not be used? What are the strengths? What are the weaknesses? What recommendations would you make for improving host security? Write a one-page paper on your analysis.

Case Project 4-3: Application Patch Management

Select four third-party applications (not OSs) that you frequently use. How does each of them address patch management? Visit their websites to determine how they alert users to new vulnerabilities. Are the patch management systems adequate? Should patch management be required of all third-party applications? What are the advantages? What are the disadvantages? Write a one-page paper on your findings.

Case Project 4-4: UEFI

Use the Internet to research UEFI. What are its advantages? What are its disadvantages? What criticisms have been leveled against it? Do you agree with the criticism? Write a one-page paper on your findings.

Case Project 4-5: STIX and TAXII

Research the Internet to find information on STIX and TAXII. How are they used? What formats do they provide? How widely are they used? What are their strengths and weaknesses? Write a one-page paper on your findings.

Case Project 4-6 Information Security Community Site Activity

The Information Security Community Site is an online companion to this textbook. It contains a wide variety of tools, information, discussion boards, and other features to assist learners. In order to gain the most benefit from the site, you will need to set up a free account.

Go to **community.cengage.com/infosec2**. Post your thoughts about the following: *Should the dark web be shut down?* What would be the advantages? What would it take for this to happen? What would threat actors do if it were suddenly unavailable? Is there any impact on free speech? Does free speech protect criminal enterprises? What do you think?

Case Project 4-7 North Ridge Security

North Ridge Security provides security consulting and assurance services. You have recently been hired as an intern to assist this organization.

You have been asked to prepare a presentation on SecDevOps for a group of students studying programming at a local college.

1. Create a PowerPoint presentation on SecDevOps, how it compares to standard application development, how it is different, and how it works. Your presentation should be at least seven slides in length.
2. As a follow-up to your presentation, you have been asked to write a one-page report on fuzzing. Use the Internet to research fuzzing, how it can be used, its strengths and weaknesses, and your recommendations.

References

1. Porter, Jon, “The PC market just had its first year of growth since 2011,” *The Verge*, Jan. 14, 2020, retrieved May 20, 2020, www.theverge.com/2020/1/14/21065100/pc-market-shipments-idc-gartner-growth-2019-laptops-desktops-windows-7-10.
2. O’Dea, S., “Number of smartphone unit shipments In the United States from 2013 to 2023 (In millions),” *Statista*, Feb. 27, 2020, retrieved May 20, 2020, www.statista.com/statistics/619811/smartphone-unit-shipments-in-the-us/.
3. Horowitz, Michael, “The head of NSA TAO advises on defensive computing for networks,” *Computerworld*, Feb. 1, 2016, accessed May 11, 2017, www.computerworld.com/article/3028025/security/defending-a-network-from-the-nsa.html.

MODULE 5

MOBILE, EMBEDDED, AND SPECIALIZED DEVICE SECURITY

After completing this module, you should be able to do the following:

1. List and compare the different types of mobile devices and how they are deployed
2. Explain the ways to secure a mobile device
3. Describe the vulnerabilities and protections of embedded and specialized devices
4. Explain the issues surrounding securing specialized devices

Front-Page Cybersecurity

Standardized testing dates back thousands of years. An ancient Chinese government conducted standardized testing to help select candidates for government jobs. In 1806, England also started using written testing for its civil service positions. In 1845, the educational pioneer Horace Mann suggested using a written test instead of an annual oral test given to Boston Public School children to measure their achievement. The written test was intended to eliminate bias because the oral test graders knew the child.

It was not long until standardized tests were being used in elementary, secondary, and college educational systems. However, unlike Mann's tests, which were designed to measure achievement *after* completing instruction, standardized school tests became a way to measure a student's ability *before* instruction. This form of standardized testing was promoted as a means to identify students who showed a high ability for success. Later the same tests were also used to evaluate the effectiveness of the teacher and institution.

One drawback of standardized tests was that manually grading was tedious and prone to error. In 1935, IBM introduced automated scoring to produce more reliable results. Automated scoring was expensive and even required the use of special pencils to record answers; however, as the technology evolved, computerized scoring became less expensive, more widely available, and more convenient, as students could use standard #2 pencils to fill in answer bubbles.

In the mid-1980s, computer-based testing was introduced as an electronic version of the traditional pencil-and-paper method. As computer-based standardized testing evolved, it provided the ability to determine what questions to ask each student. Today many standardized tests use a test taker's response on one question to determine the difficulty level of the next question.

However, as long as people have participated in standardized testing, there has been cheating on tests. At one time, cheating took the form of looking at another student's responses or writing answers on the back of the test taker's hand. Today, cheating is high-tech, with cheaters using mobile and specialized devices. Answers can be stored and retrieved from a programmable calculator or recorded on a smartphone and then displayed on a wearable smartwatch.

Like many countries, the Algerian nation has faced an epidemic of cheating among the more than 700,000 students who take Algeria's baccalaureate (four-year bachelor's degree) exit test. Test questions and answers started to appear on social media sites almost immediately after the start of the exam by students who used their smartphones to post the information while taking the test. Test latecomers could see the questions and answers before entering one of Algeria's 2,100 exam centers. The problem became so widespread in 2016 that the Algerian Education Ministry declared several exams void and required more than 500,000 students to retake the exam with new questions. Thirty-one people were arrested, including several Education Ministry employees.

In 2017, the Ministry installed mobile phone jammers in the exam centers and blocked access to Facebook, Twitter, and Instagram. Students who arrived late were banned from taking the exam but had to later attend an exam session at an alternative test center. This practice did not completely prevent test cheating.

In 2018, all exam centers installed metal detectors to prevent students from smuggling in smartphones. All teachers and test proctors also had to surrender their phones, tablets, and electronic devices. Devices for jamming wireless signals and video surveillance cameras were also installed.

Algeria took one more drastic step: it shut down all access to the Internet. Not only was Internet access unavailable to students in testing centers: the Internet was turned off all across the entire country.

By order of the Algerian government, private Internet service providers (ISPs) and the public telephone operator that provides much of the Internet access turned off the Internet for up to three hours per day during testing week. There were three one-hour blackouts on Wednesday and two each on Thursday through Monday. For everyone in Algeria, every kind of Internet or mobile connection, from wired to cellular to Wi-Fi, went dark while students were tested.

Algeria was not alone in this drastic action. The nations of Syria, Iraq, Mauritania, Uzbekistan, and several Indian states also blocked all access to the Internet during testing. Ethiopia shut down access to social media during the times of testing. China deployed drones carrying radio scanners to catch students who were using electronic devices during tests. Today most schools and testing centers ban all forms of electronic equipment, especially mobile devices such as smartphones and smartwatches, from all testing facilities.

If time travelers living 20 years ago could be transported to today's world, they likely would be shocked at how mobile devices have dramatically changed daily life in just a short period of time. Watching cars pass on the road, they would observe a high percentage of drivers talking or sending text messages on their mobile phones, often in violation of laws that prohibit it. Sitting in a classroom, the time travelers would see that almost all students use their mobile devices to read e-textbooks, access online files, and take notes. In the few remaining malls, shoppers scan bar codes on their smartphones to determine if the same item is offered at another mall store at a lower price or if it would be cheaper to immediately order it online. These dramatic changes might discourage the time travelers from jumping ahead another 20 years to see what a world filled with even more mobile devices would be like.

The statistics confirm that mobile devices have changed—and are continuing to change—our everyday lives. About 96 percent of 18- to 29-year-olds own a smartphone (the remaining 4 percent own a basic cell phone), and 81 percent of all Americans own a smartphone, compared with only 35 percent in 2011. Half of the public now owns a tablet computer (tablet ownership in 2010 was a mere 3 percent).[1] The average daily time spent consuming online media on a mobile device is 203 minutes compared to 128 minutes on desktop computers.[2]

However, just as users have flocked to mobile devices, so too have attackers. Because mobile devices have become the primary, if not exclusive, computing devices for a growing number of users, attacks directed at mobile device have increased dramatically.

In this final module on endpoint security, you will explore mobile, embedded, and specialized device security. You begin by looking at securing mobile devices and then survey embedded systems and the Internet of Things devices. Finally, you will examine how to keep specialized devices secure.

SECURING MOBILE DEVICES

CERTIFICATION

1.2 Given a scenario, analyze potential indicators to determine the type of attack.

3.5 Given a scenario, implement secure mobile solutions.

Each type of mobile device faces several cybersecurity risks. Security professionals can use a variety of techniques and technologies for securing mobile devices.

Introduction to Mobile Devices

Of the many types of mobile devices, each can connect to networks using different technologies. Enterprises also use different ways to deploy mobile devices to their employees.

Types of Mobile Devices

Most mobile devices have a common set of core features that differentiate them from other computing devices. Many, but not all, mobile devices extend their core features to include additional tools and technologies. Both types of features are listed in Table 5-1.

Table 5-1 Mobile device core and additional features

Core features	Additional features
Small form factor	Global Positioning System (GPS)
Mobile operating system	Microphone and/or digital camera
Wireless data network interface for accessing the Internet, such as Wi-Fi or cellular telephony	Wireless cellular connection for voice communications
Stores or other means of acquiring applications (apps)	Wireless personal area network interfaces such as Bluetooth or near field communications (NFC)
Local nonremovable data storage	Removable storage media
Data synchronization capabilities with a separate computer or remote servers	Support for using the device itself as removable storage for another computing device

Mobile devices include tablets, smartphones, wearables, and portable computers.

Tablets *Tablets* are portable computing devices first introduced in 2010. Designed for user convenience, tablets are thinner, lighter, easier to carry, and more intuitive to use than other types of computers. Tablets are often classified by their screen size. The two most common categories of tablet screen sizes are 5–8.5 inches (12.7–21.5 cm) and 8.5–10 inches (12.7–25.4 cm). The weight of tablets is generally less than 1.5 pounds (0.68 kg), and they are less than 1/2 inch (1.2 cm) thick. Figure 5-1 shows a typical tablet device.

Source: maximino/Shutterstock.com

Figure 5-1 Tablet device

NOTE 1

Tablets have a sensor called an accelerometer that detects vibrations and movements. It can determine the orientation of the device so that the screen image is always displayed upright.

Tablets generally lack a built-in keyboard or mouse. Instead, they rely on a touch screen that users manipulate with touch gestures to provide input. Table 5-2 lists the touch gestures for an Apple tablet.

Table 5-2 Apple touch gestures

Gesture name	Action	Usage
Tap	Lightly strike the screen	Make a selection
Double tap	Two quick taps in succession	Zoom in or out of content or an image
Flick	Place finger on the screen and quickly “swipe” in the desired direction	Scroll or pan quickly
Drag	Place finger on the screen and move it in the desired direction	Scroll or move the viewing area
Pinch open	Place thumb and finger close together on the screen and move them apart	Zoom in
Pinch close	Place thumb and finger a short distance apart on the screen and move them toward each other	Zoom out
Touch and hold	Touch the screen until the action occurs	Display an information bubble or magnify content
Two-finger scroll	Move two fingers together in the same direction	Scroll content in an element with overflow capability

Although tablets are primarily display devices with limited computing power, they have proven to be popular. Besides their portability, a primary reason for their popularity is that tablet computers have an operating system (OS) that allows them to run third-party apps. The most popular OSs for tablets are Apple iOS and iPadOS, Google Android, and Microsoft Windows.

Smartphones Earlier models of cellular telephones were called *feature phones* because they included a limited number of features, such as a camera, an MP3 music player, and ability to send and receive text messages. Many feature phones were designed to highlight a single feature, such as cameras for taking high-quality photos or a large amount of memory for music storage.

The feature phone has given way to today’s *smartphone*, which has all the tools of a feature phone plus an OS that allows it to run apps and access the Internet. Because it has an OS, a smartphone offers a broader range of functionality. Users can install apps to perform tasks for productivity, social networking, music, and so forth, much like a standard computer.

NOTE 2

Because of the ability to run apps, smartphones are essentially handheld personal computers.

Wearables Another class of mobile technology consists of devices that can be worn by the user instead of carried. Known as *wearables*, the devices can provide even greater flexibility and mobility.

The most popular wearable technology is a *smart watch*. Early smart watches were just a means to receive smartphone notifications on the user’s wrist. However, today wearables have evolved to much more sophisticated devices. A modern smart watch can still receive notifications of phone calls and text messages, but it can also be used as a fitness tracker, a contactless payment system, and safety monitor that calls emergency services if the watch detects the user has fallen. Figure 5-2 displays a smart watch.

Another popular type of wearable is a *fitness tracker*. Originally designed to monitor and record physical activity, such as counting steps, they likewise have evolved into sophisticated health-monitoring devices. Modern fitness trackers can provide continuous heart rate monitoring, GPS tracking, oxygen consumption, repetition counting (for weight training), and sleep monitoring.

Portable Computers As a class, *portable computers* are devices that closely resemble standard desktop computers. Portable computers have similar hardware (keyboard, hard disk drive, and RAM, for example) and run the same OS (Windows, Apple macOS, or Linux) and applications (such as Microsoft Office and web browsers) as general-purpose desktop computers. The primary difference is that portable computers are smaller, self-contained devices that can easily be transported from one location to another while running on battery power.

Source: Alexey Boldin/Shutterstock.com

Figure 5-2 Smart watch

NOTE 3

Many fitness trackers and smart watches use two colors of LED lights on the underside of the device to read vital signs on the human body and then measure the light absorption with photodiodes. They use green LED lights when the wearer is exercising (such as running or bicycle riding) by flashing green light onto the wrist hundreds of times per second. Human blood absorbs green light, so the heart rate can be determined by measuring the changes in green light absorption (a method called photoplethysmography, or PPG). Red LED lights are used when the wearer is not exercising. Human blood reflects red light, so about every 10 minutes, the red LEDs flash to measure the resting heart rate. The reason for having two colors of LED lights is due to accuracy and battery life. Green LEDs are more accurate, which is more important when assessing a rapid heart rate than a sedentary heart rate. But since green LEDs require more power, red LEDs are also used to save battery life.

A *laptop* computer is regarded as the earliest portable computer. A laptop is designed to replicate the abilities of a desktop computer with only slightly less processing power yet is small enough to be used on a lap or small table. A *notebook* computer is a smaller version of a laptop and is considered a lightweight personal computer. Notebook computers typically weigh less than laptops and are small enough to fit inside a briefcase. A *subnotebook* computer is even smaller than standard notebooks and use low-power processors and solid-state drives (SSDs). A *2-in-1* computer (also called a *hybrid* or *convertible*) can be used as either a subnotebook or a tablet. The devices have a touch screen and a physical keyboard; they can be transformed from a subnotebook to a tablet through a folding design or as a slate with a detachable keyboard, as shown in Figure 5-3.

A new type of computing device that resembles a laptop computer is a *web-based computer*. It contains a limited version of an OS and a web browser with an integrated

Source: Chesky/Shutterstock.com

Figure 5-3 2-in-1 computer with slate design media player. Web-based computers are designed to be used while connected to the Internet. No traditional software applications can be installed, and no user files are stored locally on the device. Instead, the device accesses online web apps and saves user files on the Internet. The most common OSs for web-based computers are the Google Chrome OS and Microsoft Windows 10 in S Mode.

NOTE 4

One of the first mobile devices was a *personal digital assistant (PDA)*, a handheld mobile device intended to replace paper systems. Most PDAs had a touch screen for entering data while others had a rudimentary keyboard that contained only a numeric keypad or thumb keyboard. Popular in the 1990s and early 2000s, PDAs fell out of favor as smartphones gained in popularity.

Mobile Device Connectivity Methods

Methods for connecting mobile devices to networks include the following:

- *Cellular.* Many mobile devices rely on **cellular telephony** for connectivity. The coverage area for a cellular telephony network is divided into cells; in a typical city, the hexagon-shaped cells measure 10 square miles (26 square kilometers). At the center of each cell is a transmitter that mobile devices in the cell use to send and receive signals. The transmitters are connected through a mobile telecommunications switching office (MTSO) that controls all of the transmitters in the cellular network and serves as the link between the cellular network and the wired telephone world. This configuration is illustrated in Figure 5-4.

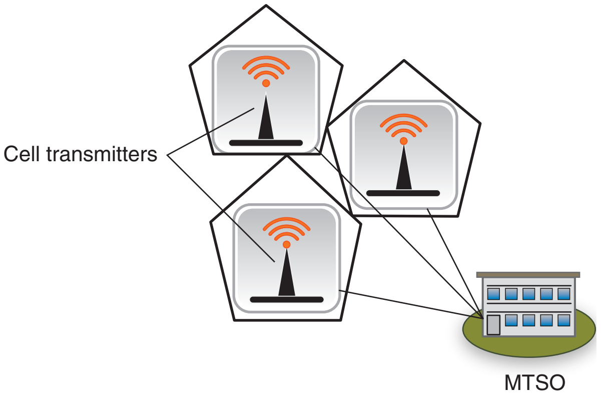


Figure 5-4 Cellular telephony network

- *Wi-Fi.* A *wireless local area network (WLAN)*, commonly called *Wi-Fi*, is designed to replace or supplement a wired local area network (LAN). Devices such as tablets, laptop computers, and smartphones within range of a centrally located connection device can send and receive data at varying transmission speeds.
- *Infrared*. Instead of using radio frequency (RF) as the communication media, some devices can use light. All the types of light that travel from the sun to the Earth make up the light spectrum, and visible light is a small part of that entire spectrum. (All other types of lights—such as X-rays, ultraviolet rays, and microwaves—are invisible to the human eye.) **Infrared** light is next to visible light on the light spectrum and, although invisible, has many of the same characteristics of visible light. At one time, infrared data ports were installed on laptop computers, printers, cameras, watches, and other devices so that data could be exchanged using infrared light. However, due to its slow speed and other limitations, infrared capabilities in mobile devices are rarely found today.
- *USB connections.* Different types and sizes of **Universal Serial Bus (USB) connectors** on mobile devices are used for data transfer. These include standard-size connectors, mini connectors, and micro connectors, all of which are available as either type A (flat) or type B (square).

NOTE 5

Other types of connectivity methods for mobile devices include Bluetooth and NFC. These are covered in detail in Module 11.

Enterprise Deployment Models

Due to the widespread use of mobile devices, it is not always feasible to require employees to carry company-owned smartphones along with their own personal cell phones. Many organizations have adopted an enterprise deployment model for mobile devices. These are listed in Table 5-3.

Table 5-3 Enterprise deployment models

Model name	Description	Employee actions	Business actions
Bring your own device (BYOD)	Employees use their own personal mobile devices for business purposes.	Employees have full responsibility for choosing and supporting the device.	This model is popular with smaller companies or those with a temporary staff.
Corporate owned, personally enabled (COPE)	Employees choose from a selection of company-approved devices.	Employees are supplied the device chosen and paid for by the company, but they can also use it for personal activities.	Company decides the level of choice and freedom for employees.
Choose your own device (CYOD)	Employees choose from a limited selection of approved devices but pay the upfront cost of the device while the business owns the contract.	Employees are offered a suite of choices that the company has approved for security, reliability, and durability.	Company often provides a stipend to pay monthly fees to wireless carrier.
Virtual desktop infrastructure (VDI)	Stores sensitive applications and data on a remote server accessed through a smartphone.	Users can customize the display of data as if the data were residing on their own mobile device.	Enterprise can centrally protect and manage apps and data on server instead of distributing to smartphones.
Corporate owned	The device is purchased and owned by the enterprise.	Employees use the phone only for company-related business.	Enterprise is responsible for all aspects of the device.

Several benefits of the BYOD, COPE, and CYOD models include the following for the enterprise:

- *Management flexibility*. BYOD and CYOD ease the management burden by eliminating the need to select a wireless data carrier and manage plans for employees.
- *Less oversight*. Businesses do not need to monitor employee telecommunications usage for overages or extra charges.
- *Cost savings*. Because employees are responsible for their own mobile device purchases and wireless data plans (BYOD) or receive a small monthly stipend (CYOD), the company can save money.
- *Increased employee performance*. Employees are more likely to be productive while traveling or working away from the office if they are comfortable with their device.
- *Simplified IT infrastructure*. By using the existing cellular telephony network, companies do not have to support a remote data network for employees.
- *Reduced internal service*. BYOD, COPE, and CYOD reduce the strain on IT help desks because users will be primarily contacting their wireless data carrier for support.

In addition, users are eager to accept the flexibility of these models. The user benefits include the following:

- *Choice of device.* Users like the freedom of choosing the type of mobile device with BYOD, COPE, and CYOD instead of being forced to accept a corporate device that may not meet their individual needs (corporate owned).
- *Choice of carrier.* Most users have identified a specific wireless data carrier they want to use and often resist being forced to use a carrier with whom they have experienced a poor past relationship.
- *Convenience.* Because almost all users already have their own device, the BYOD, COPE, and CYOD models provide the convenience of carrying only a single device.

Mobile Device Risks

Like all endpoints, mobile devices have security vulnerabilities and are at risk of being compromised. Many attacks directly target mobile devices, which has a profound impact on organizations. In a recent survey, almost 40 percent of organizations admitted to suffering a compromise due to a mobile device. Half of the incidents resulted in the loss of data, and more than one-third were described as major incidents with lasting repercussions. Threat actors often use compromised mobile devices to pivot to other targets: 58 percent of the attacks on mobile devices led to the compromise of other devices. Almost half of respondents reported that their organizations sacrificed mobile security to perform work more quickly.[3]

CAUTION The survey summarized by saying, "We found that many companies are failing to protect their mobile devices. And we're not talking about some almost-impossible-to-achieve gold standard. We're talking about companies failing to meet even a basic level of preparedness."[4]

Security risks associated with using mobile devices include mobile device vulnerabilities, connection vulnerabilities, and accessing untrusted content.

Mobile Device Vulnerabilities

Mobile device vulnerabilities include physical security, limited updates, location tracking, and unauthorized recording.

Physical Security The greatest asset of a mobile device—its portability—is also one of its greatest vulnerabilities. Mobile devices are frequently lost or stolen. Unlike desktop computers, mobile devices by their very nature are designed for use in a variety of locations, both public (coffee shops, hotels, and conference centers) and private (employee homes and cars). These locations are outside of the enterprise's normal protected physical perimeter of walls, security guards, and locked doors. One-quarter of all laptop thefts occurred from unattended cars or while traveling on airplanes and trains, 15 percent of thefts occurred in airports and hotels, and 12 percent occurred in restaurants.[5] However, due to the portable nature of a mobile device, even a strong physical perimeter does not always provide protection from theft. Almost half of all laptop thefts occur from school offices and classrooms.[6]

Unless properly protected, any data on a stolen or lost device could be retrieved by a thief. Of greater concern may be that the device itself can serve as an entry point into corporate data. On average, every employee at an organization has access to 17 million files and 1.21 million folders. The average organization has more than half a million sensitive files, and 17 percent of all sensitive files are accessible to each employee.[7]

Limited Updates Currently there are two dominant OSs for mobile devices. Apple iOS, developed by Apple for its mobile devices, is a closed and proprietary architecture. Google Android is not proprietary but is open for any original equipment manufacturer (OEM) to install or even modify. (However, modifications must adhere to Google's criteria to access all Google services.) Many OEMs worldwide make mobile devices that use Android because it is freely available.

Security patches and updates for these two mobile OSs are distributed through **firmware over-the-air (OTA) updates**. Though they are called "firmware" OTA updates, they include modifying the device's firmware and updating the OS software. Apple commits to providing OTA updates for at least four years after the OS is released. Users can set iOS updates to occur automatically or manually, either through the device itself or by connecting it to a computer through which the update is downloaded.

However, OTA updates for Android OSs vary considerably. Mobile hardware devices developed and sold by Google receive Android OTA updates for three years after the device is first released. Other OEMs are required to provide OTAs for at least two years. However, after two years, many OEMs are hesitant to distribute Google updates because it limits their ability to differentiate themselves from competitors if all versions of Android start to look the same through updates. Also, because OEMs want to sell as many devices as possible, they have no financial incentive to update mobile devices that users would then continue to use indefinitely.

Whereas users once regularly purchased new mobile devices about every two years, that is no longer the case. Due to the high cost of some mobile devices, users are keeping their devices for longer periods of time. This can result in people using mobile devices that no longer receive OTA security updates and thus have become vulnerable.

Location Tracking The **Global Positioning System (GPS)** is a satellite-based navigation system that provides information to a GPS receiver anywhere on (or near) the Earth with an unobstructed line of sight to four or more GPS satellites. Mobile devices with GPS capabilities typically support **geolocation**, or identifying the geographical location of the device. When finding a person carrying a mobile device, geolocation also identifies the location of a close friend or displays the address of the nearest coffee shop. Location services are used extensively by social media, navigation systems, weather systems, and other mobile-aware applications.

However, mobile devices using geolocation are at increased risk of targeted physical attacks. An attacker can determine where users with mobile devices are currently located and use that information to follow them and steal the mobile devices or inflict physical harm. In addition, attackers can craft attacking by compiling a list of people with whom the users associate and the types of activities they perform.

A related risk is **GPS tagging** (also called **geo-tagging**), which is adding geographical identification data to media such as digital photos taken on a mobile device. A user who, for example, posts a photo on a social networking site may inadvertently identify a private location to anyone who can access the photo.

Unauthorized Recording Video cameras ("webcams") and microphones on mobile devices have been a frequent target of attackers. By infecting a device with malware, a threat actor can secretly spy on an unsuspecting victim and record conversations or videos.

NOTE 6

Banks are expanding the use of geolocation to help reduce bank card fraud. When a user makes a purchase at a store, the bank can immediately check the location of the user's authorized cell phone. If the cell phone and the bank card are in the same place, then the purchase may be considered legitimate. But if the cell phone is in Nashville and someone is trying to make a purchase in a store in Tampa, then the payment may be rejected. Geolocation can also help prevent rejecting valid purchases. One credit card issuer says that it can reduce unnecessary declines by as much as 30 percent.[8]

Connection Vulnerabilities

Vulnerabilities in mobile device connections can also be exploited by threat actors. These vulnerabilities are summarized in Table 5-4.

Table 5-4 Connection vulnerabilities

Name	Description	Vulnerability
Tethering	A mobile device with an active Internet connection can be used to share that connection with other mobile devices through Bluetooth or Wi-Fi.	An unsecured mobile device may infect other tethered mobile devices or the corporate network.
USB On-the-Go (OTG)	An OTG mobile device with a USB connection can function as either a host (to which other devices may be connected such as a USB flash drive) for **external media access** or as a peripheral (such as a mass storage device) to another host.	Connecting a **malicious flash drive** infected with malware to a mobile device could result in an infection, just as using a device as a peripheral while connected to an infected computer could allow malware to be sent to the device.
Malicious USB cable	A USB cable could be embedded with a Wi-Fi controller that can receive commands from a nearby device to send malicious commands to the connected mobile device.	The device will recognize the cable as a Human Interface Device (similar to a mouse or keyboard), giving the attacker enough permissions to exploit the system.
Hotspots	A hotspot is a location where users can access the Internet with a wireless signal.	Because public hotspots are beyond the control of the organization, attackers can eavesdrop on the data transmissions and view sensitive information.

Accessing Untrusted Content

Normally users cannot download and install unapproved apps on their iOS or Android device. Instead, users must access the Apple App Store or Google Play Store (or other Android store) to download and install an app on a mobile device; in fact, Apple devices can only download from the App store. However, users can circumvent the built-in installation limitations on their smartphone (called **jailbreaking** on Apple iOS devices or **rooting** on Android devices) to download from an unofficial **third-party app store** (called **sideloading**) or even write their own **custom firmware** to run on their device. Because the apps have not been vetted, they may contain security vulnerabilities or malicious code.

CAUTION Jailbreaking and rooting give access to the underlying OS and file system of the mobile device with full permissions. For example, a jailbreak on an Apple iPhone gives users access to a UNIX shell with root privileges, essentially allowing them to do anything on the device.

Jailbreaking and rooting are not the same as **carrier unlocking**. Originally almost all cell phones were connected ("locked") to a specific wireless carrier so that neither the phone nor the phone number could be transferred to another carrier. The restriction was enforced by a 2012 decision from the Library of Congress that cell phone unlocking was a violation of the Digital Millennium Copyright Act. However, in 2015, the Unlocking Consumer Choice and Wireless Competition Act was passed to approve carrier unlocking.

Untrusted content can also invade mobile devices through **short message service (SMS)**, which are text messages of a maximum of 160 characters; **multimedia messaging service (MMS)**, which allows text message to include pictures, video, and audio; and **rich communication services (RCS)**, which can convert a texting app into a live chat platform and supports pictures, videos, location, stickers, and emojis. Threat actors can send SMS messages containing links to untrusted content or specially crafted MMS or RCS videos that can introduce malware into the device.

Mobile devices can also access untrusted content that other types of computing devices generally cannot access. One example is a *Quick Response (QR)* code. A QR code is a matrix or two-dimensional barcode consisting of black modules (square dots) arranged in a square grid on a white background. QR codes can store website URLs, plain text, phone numbers, email addresses, or virtually any alphanumeric data up to 4,296 characters, which can be read by an imaging device such as a mobile device's camera. A QR code for *www.cengage.com* is illustrated in Figure 5-5.

Source: qrstuff.com

Figure 5-5 QR code

An attacker can create an advertisement listing a reputable website, such as a bank, but include a QR code that contains a malicious URL. Once the user snaps a picture of the QR code using the camera on a mobile device, the code directs the web browser on the mobile device to the attacker's imposter website or to a site that immediately downloads malware.

Protecting Mobile Devices

Users can take steps to secure a mobile device. These include configuring the device and using mobile management tools.

Device Configuration

Several configurations should be considered when setting up a mobile device for use. These include using strong authentication, managing encryption, segmenting storage, and enabling loss or theft services.

Use Strong Authentication Verifying that the authentic user of a mobile device involves requiring a strong passcode and restricting unauthorized users with a screen lock.

Passcode. Almost all mobile devices have options for configuring a passcode that must be entered before access will be granted. Although passwords are the most secure option, most users unfortunately opt not to configure their device with a password. This is primarily due to the time needed to enter the password and the difficulty of entering a complex password on the device's small on-screen keyboard.

Another option is to use a **personal identification number (PIN)**. Unlike a password that can be comprised of letters, numbers, and characters, a PIN is made up of numbers only. Although the length of the PIN can usually range from four to 16 numbers, many users choose to set a short four-digit PIN, like those used with a bank's automated teller machine (ATM). However, short PIN codes provide only a limited amount of security. An analysis of 3.4 million users' four-digit (0000–9999) PINs that were compromised revealed that users create predictable PIN patterns. The PIN *1234* was used in more than one out of every 10 PINs. Table 5-5 lists the five most common PINs and their frequency of use. Of the 10,000 potential PIN combinations, 26.83 percent of all PINs could be guessed by attempting the top 20 most frequent PINs.[9]

Table 5-5 Most common PINs

PIN	Frequency of use
1234	10.71%
1111	6.01%
0000	1.88%
1212	1.19%
7777	0.74%

NOTE 7

The research also revealed that the least common PIN was *8068*, which appeared in only 25 of the 3.4 million PINs.

A third option is to use a fingerprint to unlock the mobile device. Several smartphone devices have the fingerprint sensor on the back of the phone. This allows the user to access the fingerprint reader without moving their index finger from the back of the phone (where the index finger is normally located while holding the phone).

A final option is to draw or swipe a specific pattern connecting dots to unlock the device, as illustrated in Figure 5-6. Swipe patterns can be detected by threat actors who watch a user draw the pattern or observe any lingering "smear" on the screen.

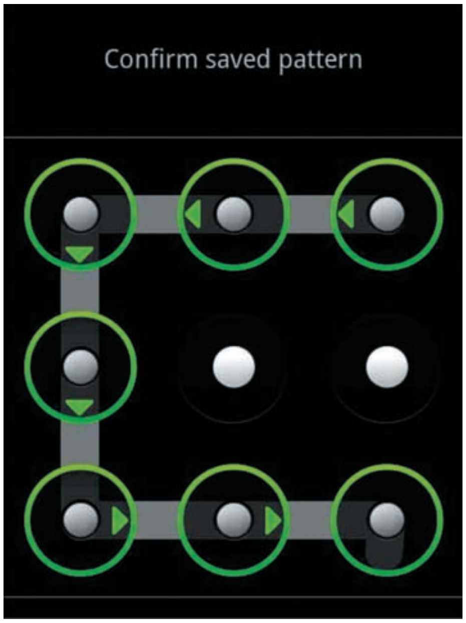
NOTE 8

Accessing a device through fingerprint, face, or voice is called *biometrics* and is covered in Module 12.

Screen Lock. A **screen lock** prevents the mobile device from being accessed until the user enters the correct passcode, permitting access. Lock screens should be configured so that whenever the device is turned on or is inactive for a period, the user must enter the passcode. Most mobile devices can be set to have the screen automatically lock after anywhere from 5 seconds to 50 minutes of inactivity.

Some mobile devices can be configured so that the device automatically unlocks and stays unlocked (ignoring the inactivity setting) until a specific action occurs. This is called **context-aware authentication**, which is using a contextual setting to validate the user. An example of context-aware authentication is in the Google Android OS, which has a feature called Smart Lock that can be configured depending on the context. These contexts are listed in Table 5-6.

Manage Encryption Early versions of mobile devices using Apple iOS or Google Android did not provide native encryption, so third-party apps had to be installed to encrypt data. However, later versions of both OSs encrypt all user data on their mobile devices (**full disk encryption**) by default when the device is locked.


Confirm saved pattern

Source: OnlineAndroidTips.com

Figure 5-6 Swipe pattern

NOTE 9

Android provides an encryption option called *file-based encryption*, which is considered more secure than full disk encryption. File-based encryption encrypts each file with a different key so that files can be unlocked independently without decrypting an entire partition at once. The device can decrypt and use files needed to boot the system and process critical notifications while not decrypting personal apps and data.

Table 5-6 Android Smart Lock configuration options

Configuration name	Explanation	Comments
On-body detection	Device turns on and remains on when it is on the user's body in a pocket or purse.	On-body detection learns the pattern of how the user walks, and if it detects a different walk style, it locks the device.
Trusted places	Can set a specific location where the phone will turn on and then off when the user leaves the location.	Device will remain unlocked in an 80-meter radius around a building; users can also designate a specific building at a single address.
Trusted devices	Device will unlock whenever it is connected to another specific device.	Common trusted devices are Bluetooth watches, fitness trackers, or car systems; users should avoid trusted devices that are always with the device such as a Bluetooth mouse.
Trusted face	Whenever the device is turned on, it will search for the designated face and unlock if it recognizes the user.	This is the least secure configuration option because the device could be tricked by someone who looks similar.
Trusted voice	If a user says, "OK Google," voice commands can be issued without unlocking the device.	Trusted voice does not completely unlock the device as with other options but only gives the ability to issue some voice commands.

Although user data on a mobile device—*local* data-at-rest—is encrypted so that unauthorized users cannot access it, mobile device data can still be accessed through *remote* data-at-rest.

Data from mobile devices is routinely backed up to Apple's iCloud or to a Google server. Although the data on the servers is encrypted, Apple and Google possess the decryption keys necessary to unlock the data on their servers. Because the data is encrypted on the user's device and is inaccessible to outside parties, courts routinely serve orders to Apple and Google to provide the same data stored on their servers using their decryption keys. Those users who are concerned about maintaining the highest level of security on their data often turn off backups to iCloud or Google servers.

Segment Storage With the exception of corporate-owned devices, each of the other enterprise deployment models (BYOD, COPE, and CYOD) permit the owner of a mobile device to use it for both business and personal needs. However, this usage may comingle critical business data with personal photos, downloads, and SMS text messages, which is not desirable to the enterprise or the user.

An option on mobile devices that contain personal and corporate data is **storage segmentation**, or separating business data from personal data. Users can apply **containerization**, or separating storage into business and personal "containers" and managing each appropriately. Segmenting storage on a mobile device used for both business and personal needs has advantages. It helps companies avoid data ownership privacy issues and legal concerns regarding a user's personal data stored on the device. In addition, it allows companies to delete only business data when necessary without touching personal data.

Enable Loss or Theft Services One of the greatest risks of a mobile device is the loss or theft of the device. Unprotected devices can be used to access corporate networks or view sensitive data stored on them. If a mobile device is lost or stolen, several security features can be used to locate the device or limit the damage. Many of these features can be configured through the OS or an installed third-party app. The features are listed in Table 5-7.

If a lost or stolen device cannot be located, it may be necessary to perform a **remote wipe**, which will erase sensitive data stored on the mobile device. A remote wipe ensures that even if a thief accesses the device, no sensitive data will be compromised.

To reduce the risk of theft or loss, users should consider the following best practices:

- Keep the mobile device out of sight when traveling in a high-risk area.
- Avoid becoming distracted by what is on the device. Always maintain an awareness of your surroundings.
- When holding a device, use both hands to make it more difficult for a thief to snatch.
- Do not use the device on escalators or near transit train doors.

- White or red headphone cords may indicate they are connected to an expensive device. Consider changing the cord to a less conspicuous color.
- If a theft does occur, do not resist or chase the thief. Instead, take note of the suspect's description, including any identifying characteristics and clothing, and then call the authorities. Also contact the organization or wireless carrier and change all passwords for accounts accessed on the device.

Table 5-7 Security features for locating lost or stolen mobile devices

Security feature	Explanation
Alarm	The device can generate an alarm even if it is on mute.
Last known location	If the battery is charged to less than a specific percentage, the device's last known location can be indicated on an online map.
Locate	The current location of the device can be pinpointed on a map through the device's GPS.
Remote lockout	The mobile device can be remotely locked and a custom message sent that is displayed on the login screen.
Thief picture	Thieves who enter an incorrect passcode three times will have their picture taken through the device's on-board camera and emailed to the owner.

Mobile Management Tools

When using mobile devices in the enterprise, several support tools can facilitate the management of the devices. These include mobile device management, mobile application management, mobile content management, and unified endpoint management.

Mobile Device Management (MDM) **Mobile device management (MDM)** tools allow a device to be managed remotely by an organization. MDM typically involves a server component, which sends out management commands to the mobile devices, and a client component, which runs on the mobile device to receive and implement the management commands. An administrator can then perform OTA updates or change the configuration on one device, groups of devices, or all devices.

Some features that MDM tools provide include the following:

- Apply or modify default device settings.
- Approve or quarantine new mobile devices.
- Configure email, calendar, contacts, and Wi-Fi profile settings.
- Detect and restrict jailbroken and rooted devices.
- Display an acceptable use policy that requires consent before allowing access.
- Distribute and manage public and corporate apps.
- Enforce encryption settings, antivirus updates, and patch management.
- Enforce **geofencing**, which is using the device's GPS to define geographical boundaries where an app can be used.
- Securely share and update documents and corporate policies.
- Selectively erase corporate data while leaving personal data intact.
- Send SMS text messages to selected users or groups of users (called **push notification services**).

Mobile Application Management (MAM) Whereas MDM focuses on the device, **mobile application management (MAM)** covers **application management**, which comprises the tools and services responsible for distributing and controlling access to apps. The apps can be internally developed or commercially available apps.

Mobile Content Management (MCM) **Content management** supports the creation and subsequent editing and modification of digital content by multiple employees. It can include tracking editing history, version control (recording changes and "rolling

NOTE 10

MDM provides a high degree of control over the device but a lower level of control on the apps, whereas MAM gives a higher level of control over apps but less control over the device.

back" to a previous version if necessary), indexing, and searching. A **mobile content management (MCM)** system is tuned to provide content management to hundreds or even thousands of mobile devices used by employees in an enterprise.

Unified Endpoint Management (UEM) All the capabilities in MDM, MAM, and MCM can be supported by **unified endpoint management (UEM)**. UEM is a group or class of software tools with a single management interface for mobile devices as well as computer devices. It provides capabilities for managing and securing mobile devices, applications, and content.

TWO RIGHTS & A WRONG

1. Due to its slow speed and other limitations, infrared capabilities in mobile devices are rarely found today.
2. COPE allows users to use their own personal mobile devices for business purposes.
3. Circumventing the installed built-in limitations on an Apple iPhone is called jailbreaking.

See Appendix B for the answer.

EMBEDDED SYSTEMS AND SPECIALIZED DEVICES

CERTIFICATION

2.6 Explain the security implications of embedded and specialized systems.

Not all computing systems are desktop or mobile devices designed for human input. Computing capabilities can be integrated into appliances and other devices. An **embedded system** is computer hardware and software contained within a larger system designed for a specific function. A growing trend is to add the capabilities to devices that have never had computing power before. These devices can pose security risks.

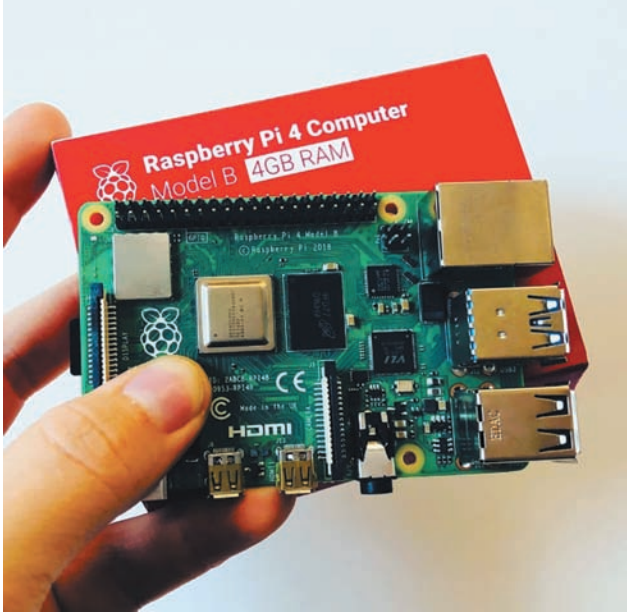
Types of Devices

Categories of embedded and specialized devices include the hardware and software that can be used to create these devices, specialized systems, industrial systems, other devices, and IoT devices.

Hardware and Software

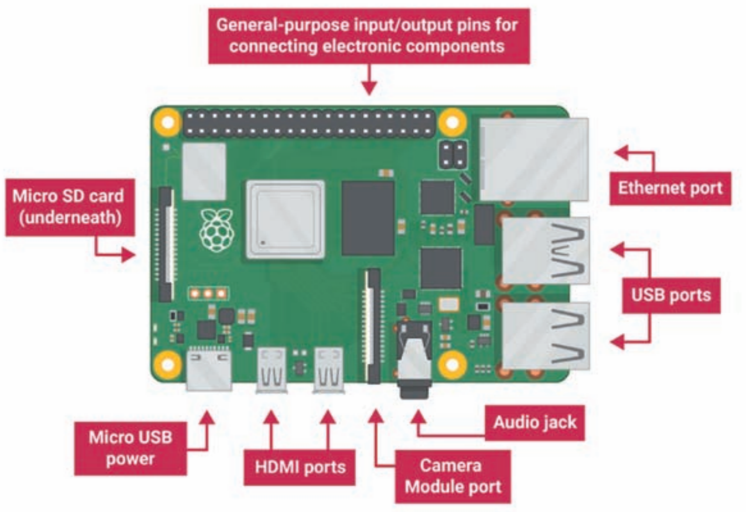
Hardware and software components are easily available for industrious users to create their own specialized device. One of the most common hardware components is the **Raspberry Pi**. This is a low-cost, credit-card-sized computer motherboard, as shown in Figure 5-7. The motherboard has hardware ports that can connect to a range of peripherals. Figure 5-8 shows the Raspberry Pi ports. The Raspberry Pi can perform almost any task that a standard computer device can, such as browsing the Internet, playing high-definition video, creating spreadsheets, and playing games. It can also be used to control a specialized device.

A device similar to the Raspberry Pi is the **Arduino**. Unlike the Raspberry Pi, which can function as a complete computer, the Arduino is designed as a controller for other devices: it has an eight-bit microcontroller instead of a 64-bit microprocessor on the Raspberry Pi, a limited amount of RAM, and no operating system. In addition, it can only run programs compiled for the Arduino platform, most of which



Source: Raspberry Pi Foundation

Figure 5-7 Raspberry Pi


General-purpose input/output pins for connecting electronic components
Micro SD card (underneath)
Ethernet port
USB ports
Audio jack
Micro USB power
HDMI ports
Camera Module port

Source: Raspberry Pi Foundation

Figure 5-8 Raspberry Pi ports

must be written in the C++ programming language. Although the Raspberry Pi and Arduino can be used to interact with other specialized devices—such as control a robot, build a weather station, broadcast an FM radio signal, or build an automatic plant-watering device—the Arduino is generally considered a better solution. It has only a single USB port, a power input, and a set of input/output pins for connections but consumes little power.

Although the Raspberry Pi and Arduino are small motherboards, a **field-programmable gate array (FPGA)** is a hardware "chip" or integrated circuit (IC) that can be programed by the user ("field programmable") to carry out one or more logical operations (ICs on standard computers as well as a Raspberry Pi and Arduino cannot be user programmed). Specifically, a FPGA is an IC that consists of internal hardware blocks with user-programmable interconnects to customize operations for a specific application. A user can write software that loads onto the FPGA chip and executes functions, and that software can later be replaced or deleted.

An even smaller component than the Raspberry Pi or Arduino is a **system on a chip (SoC)**. A SoC combines all the required electronic circuits of the various computer components on a single IC chip. (The Raspberry Pi and Arduino are tiny motherboards that contain ICs, one of which is an SoC.) SoCs often use a **real-time operating system (RTOS)**, an OS specifically designed for an SoC in an embedded or specialized system. Standard computer systems, such as a laptop with a mouse and a keyboard or a tablet with a touch screen, typically receive irregular "bursts" of input data from a user or a network connection. Embedded systems, on the other hand, receive large amounts of data very quickly, such as an aircraft preparing to land on a runway at night during a storm. An RTOS is tuned to accommodate high volumes of data that must be immediately processed for critical decision making.

NOTE 11

FPGAs are used in aerospace and defense, medical electronics, digital television, consumer electronics, industrial motor control, scientific instruments, cybersecurity systems, and wireless communications. Microsoft is now using FPGAs in its data centers to run Bing search algorithms.

Specialized Systems

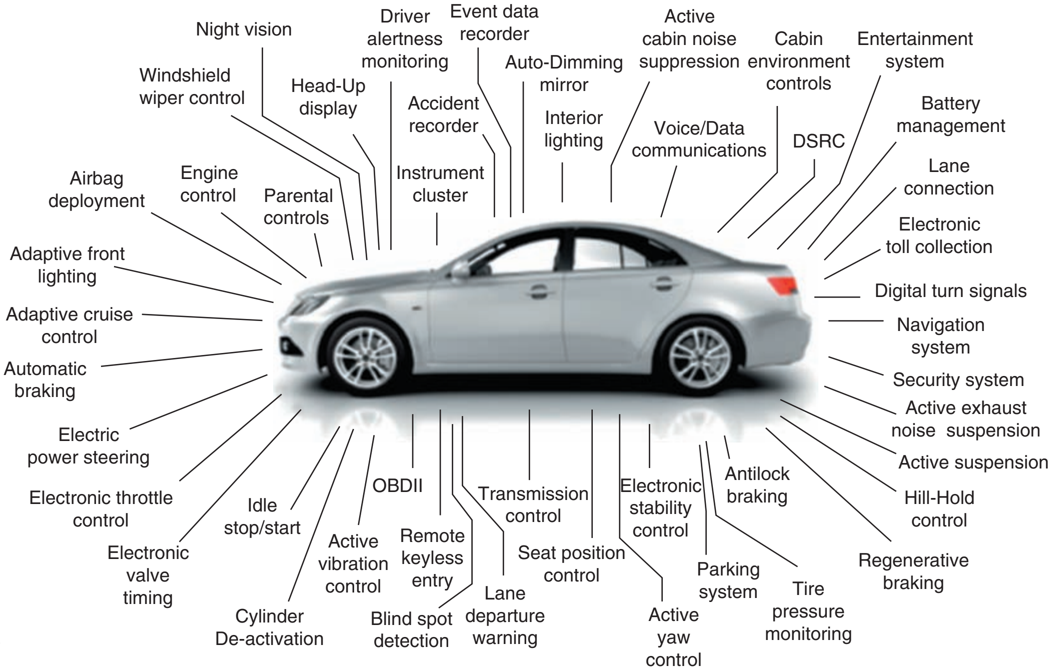
Several types of specialized systems are designed for specific applications. One example measures the amount of utilities consumed. Traditionally, households have had utilities such as electricity and water measured by an analog meter that records the amount of electricity or water being used. An employee from the utility must visit each home and read from the meter the amount that was consumed for the month so that a bill can be send to the occupant. Analog meters are being replaced by digital **smart meters**. Smart meters have several advantages over analog meters, and these are listed in Table 5-8.

Other specialized systems include medical systems, aircraft, and vehicles. The progression of specialized systems in automobiles is an example of how the systems have dramatically changed human-to-machine interaction. The first automobile embedded systems appeared in mass-production vehicles in the mid-1970s in response to regulations calling for higher fuel economy and emission standards. They handled basic functions such as engine ignition timing

Table 5-8 Analog meters vs. smart meters

Action	Analog meter	Smart meter
Meter readings	Employee must visit the dwelling each month to read the meter.	Meter readings are transmitted daily, hourly, or even by the minute to the utility company.
Servicing	Annual servicing is required in order to maintain accuracy.	Battery replacement every 20 years.
Tamper protection	Data must be analyzed over long periods to identify anomalies.	Can alert utility in the event of tampering or theft.
Emergency communication	None available	Transmits “last gasp” notification of a problem to utility company.

and transmission shifting. By the 1980s, more sophisticated computerized engine-management systems enabled the use of reliable electronic fuel-injection systems, and later active safety systems such as antilock braking and traction and stability control features were added, all controlled by embedded systems. Today, embedded systems in cars use sonar, radar, and laser emitters to control brakes, steering, and the throttle to perform functions such as blind spot and pedestrian collision warnings, automated braking, safe distance keeping, and fully automated parking. Some of the embedded systems in cars are shown in Figure 5-9.



ChipsEtc.com

Figure 5-9 Embedded systems in cars

Industrial Systems

Industrial control systems (ICSs) in local or at remote locations collect, monitor, and process real-time data so that machines can directly control devices such as valves, pumps, and motors without human intervention. ICSs are managed by a larger **supervisory control and data acquisition (SCADA)** system. SCADA systems are crucial today for industrial organizations. They help to maintain efficiency and provide information on issues to help reduce downtime.

Other Specialized Systems

Other examples of specialized systems include **heating, ventilation, and air conditioning (HVAC)** environmental systems, which provide and regulate heating and cooling.

A **multifunctional printer (MFP)** combines the functions of a printer, copier, scanner, and fax machine. These peripheral devices are essentially special-purpose computers with a CPU; a hard drive that stores all received print jobs, faxes, and scanned images; a LAN or wireless LAN connection; a telephone connection for faxes; and a USB port to allow users to print documents stored on that device. *Smart MFDs* even have an OS that allows additional applications to be installed that extend the abilities of the MFD.

An **unmanned aerial vehicle (UAV)**, commonly known as a **drone**, is an aircraft without a human pilot on board to control its flight. Drones can be controlled by a remote human operator, usually on the ground, or autonomously by preprogramming the onboard computers. While drones were originally used in military applications, today they have expanded into commercial, scientific, agricultural, and recreational uses. They are commonly used for policing and surveillance, product deliveries, aerial photography, infrastructure inspections, and even drone racing. A drone is shown in Figure 5-10.

For several years, different forms of digital communications have been unified into a single mode of transmission by shifting to an all-digital technology infrastructure. One of the most visible unification efforts is the convergence of voice and data traffic over a single Internet Protocol (IP) network. Using IP, various services such as voice, video, and data can be combined (*multiplexed*) and transported under a universal format. Known as **voice over IP (VoIP)**, it uses a data-based IP network to add digital voice clients and new voice applications onto the IP network.

Source: Den Rozhnovsky/Shutterstock.com

Figure 5-10 Drone

NOTE 12

The term drone was first used to refer to unmanned aircraft that were used for target practice by battleships in the 1920s.

Internet of Things

The Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T) defines the **Internet of Things (IoT)** as *a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies*.[10] More simply put, the IoT is connecting any device to the Internet for the purpose of sending and receiving data to be acted upon. Although this definition could encompass laptop computers and tablets, more often IoT refers to devices that heretofore were not considered as computing devices connected to a data network. IoT devices include wearable technology and multifunctional devices as well as many everyday home automation items such as thermostats, coffee makers, tire sensors, slow cookers, keyless entry systems, washing machines, electric toothbrushes, headphones, and light bulbs, to name just a few. It is estimated that, by 2025, there will be more than 25 billion IoT devices, of which more than half will be consumer devices.[11]

An example of IoT and the great promise it holds can be seen in devices that can be used for *body area networks (BAN)*, which is a network system of IoT devices in close proximity to a person's body that cooperate for the benefit of the user. Sensors are placed on the human body to monitor electrocardiogram (EKG) impulses, blood pressure, glucose, and other human biological functions. The readings are transmitted via computer or smartphone to a third-party physician who can make decisions regarding any medications to prescribe or lifestyle changes to recommend. This is called a *managed body sensor network (MBSN)*.

A more robust approach is the *autonomous body sensor network (ABSN)*. Instead of only reading and transmitting information, an ABSN introduces actuators in addition to the sensors so that immediate effects can be made on the human body. Dozens of IoT micro-stimulator implant devices can be used to treat paralysis and other conditions. These devices take in signals from the human nervous system and then stimulate nerves through electrical charges that cause muscles to contract and limbs to move, bypassing areas of the nervous system that have been impaired by strokes or spinal cord or brain injuries. The ABSN can expand the use of functional electric stimulation to restore sensation, mobility, and function to those persons with paralyzed limbs and organs. Another ABSN being tested installs "stretchy"

NOTE 13

IoT BAN sensors that continuously monitor body temperature are being proposed to help stem the spread of viruses among humans. The sensors could help to shut down emerging viruses before a pandemic can take hold.

microprocessors on the tips of cardiac catheters, which are threaded through arteries into the heart. The catheters will be used to monitor the heart's electrical activity to pinpoint the location of irregular heartbeats and, if necessary, can treat the heart by "zapping" the tissue that is malfunctioning.

Security Issues

Although embedded systems and specialized devices are widely used and will continue to grow exponentially, they introduce significant security issues.

Consider the following actual incidents involving embedded systems and specialized devices:

- A worm was discovered that was actively targeting Windows computers that managed large-scale SCADA systems—which are often found in military installations, oil pipeline control systems, manufacturing environments, and even nuclear power plants. Named Stuxnet, the malware attempted to gain administrative access to other computers through the network to control the SCADA system. It appears that Stuxnet's primary target was nuclear reactors at the Bushehr nuclear power plant. Located in southwestern Iran near the Persian Gulf, Bushehr was a source of tension between Iran and the West (including the United States) because of fear that spent fuel from the reactor could be reprocessed elsewhere in the country to produce weapons-grade plutonium for use in nuclear warheads. Stuxnet was ultimately not successful in its attack.[12]
- Marc G. was in the kitchen when he began to hear strange sounds coming from the nursery of his two-year-old daughter Allyson. Marc and his wife entered the nursery and heard a stranger's voice calling out Allyson's name, cursing at her, and calling her vile names. The parents discovered that the voice was coming from the electronic baby monitor in Allyson's room that contained a camera, microphone, and speaker connected to their home Wi-Fi network. Because they did not have any security set on their wireless network, the attacker had been able to take control of the baby monitor from an unknown remote location. When Marc and his wife stepped in front of the camera, the attacker turned his verbal attack toward them. They quickly unplugged the device. The parents surmised that the attacker knew their daughter's name because he saw "Allyson" spelled out on the wall in her room. This situation is not unique: it is estimated that hundreds of thousands of wireless IoT cameras can easily be exploited because they have virtually no security.

These incidents illustrate the lack of security in embedded systems and specialized devices and can result in a wide range of attacks. Several **constraints** (limitations) make security a challenge for these systems and specialized devices. These security constraints are listed in Table 5-9.

Table 5-9 Security constraints for embedded systems and specialized devices

Constraint	Explanation
Power	To prolong battery life, devices and systems are optimized to draw very low levels of power and thus lack the ability to perform strong security measures.
Compute	Due to their size, small devices typically possess low processing capabilities, which restricts complex and comprehensive security measures.
Network	To simplify connecting a device to a network, many device designers support network protocols that lack advanced security features.
Cryptography	Encryption and decryption are resource-intensive tasks that require significant processing and storage capacities that these devices lack.
Inability to patch	Few, if any, devices have been designed with the capacity for being updated to address exposed security vulnerabilities.
Authentication	To keep costs at a minimum, most devices lack authentication features.
Range	Not all devices have long-range capabilities to access remote security updates.
Cost	Most developers are concerned primarily with making products as inexpensive as possible, which means leaving out all security protections.
Implied trust	Many devices are designed without any security features but operate on an "implied trust" basis that assumes all other devices or users can be trusted.
Weak defaults	User names (such as "root," "admin," and "support") and passwords ("admin," "888888," "default," "123456," "54321," and even "password") for accessing devices are often simple and well known.

Over several years, many industry-led initiatives have attempted to address security vulnerabilities in IoT and embedded devices. However, the initiatives were scattered and did not represent a comprehensive solution to the problem. To address security in these devices, governments have begun to propose or enact legislation to require stronger security on embedded systems and specialized devices. The *Internet of Things (IoT) Cybersecurity Improvement Act of 2019* was legislation introduced in the U.S. Senate in May 2019 with the following requirements:

- Require the National Institute of Standards and Technology (NIST) to issue recommendations addressing, at a minimum, secure development, identity management, patching, and configuration management for IoT devices.
- Direct the Office of Management and Budget (OMB) to issue guidelines for each agency that are consistent with the NIST recommendations, and charge OMB with reviewing the policies at least every five years.
- Require any Internet-connected devices purchased by the federal government to comply with those recommendations.
- Direct NIST to work with cybersecurity researchers and industry experts to publish guidance on coordinated vulnerability disclosure to ensure that vulnerabilities related to agency devices are addressed.
- Require contractors and vendors providing IoT devices to the U.S. government to adopt coordinated vulnerability disclosure policies so that if a vulnerability is uncovered, that information is disseminated.

California and Oregon passed state laws addressing IoT security that went into effect in January 2020. Both state laws require that connected devices be equipped with "reasonable security features" appropriate for the nature and function of the device and the information the device collects, contains, or transmits. Devices must be designed to protect both the device itself and any information contained within the device from unauthorized access, destruction, use, modification, or disclosure.

CAUTION

When defining a "reasonable security feature," both California and Oregon laws say that such a feature may consist of a preprogrammed password unique to each device manufactured. An alternative is a security feature that requires a user to create a new means of authentication before accessing a device for the first time. Beyond this example, however, neither law provides clear guidance as to what else could constitute a "reasonable security feature."

TWO RIGHTS & A WRONG

1. Multiple SCADAs are controlled by an ICS.
2. Power, compute, and network are all security constraints for embedded systems and specialized devices.
3. An RTOS is tuned to accommodate very high volumes of data that must be immediately processed for critical decision making.

See Appendix B for the answer.

VM LAB

You're now ready to complete the live virtual machine labs for this module. The labs can be found in each module in the MindTap.

SUMMARY

- There are several types of mobile devices. Tablet computers are portable computing devices smaller than portable computers, larger than smartphones, and focused on ease of use. Tablets generally lack a built-in keyboard and rely on a touch screen. A smartphone includes an operating system that allows it to run apps and access the Internet, and it offers a broad range of functionality. A new class of mobile technology is wearable technology, devices that can be worn by the user instead of being carried.

- Portable computers are devices that closely resemble standard desktop computers. A laptop is designed to replicate the abilities of a desktop computer with only slightly less processing power yet is small enough to be used on a lap or small table. A notebook computer is a smaller version of a laptop computer that is designed to include only the most basic frequently used features of a standard computer in a smaller size that is easy to carry. A subnotebook computer is even smaller than the standard notebook. A 2-in-1 computer can be used as either a subnotebook or a tablet. Web-based computers are designed to be used primarily while connected to the Internet.
- Connectivity methods used to connect mobile devices to networks include cellular telephony, which divides the coverage area into cells. Wi-Fi is a wireless local area network standard. USB connectors on mobile devices are used for data transfer.
- It is not always feasible to require an employee to carry a company-owned smartphone along with a personal cell phone. Many organizations have adopted an enterprise deployment model as it relates to mobile devices. Bring your own device (BYOD) allows users to use their own personal mobile devices for business purposes. Corporate owned, personally enabled (COPE) gives employees a choice from a selection of company approved devices. Choose your own device (CYOD) gives employees a limited selection of approved devices, though the employee pays the upfront cost of the device while the business owns the contract. A virtual desktop infrastructure (VDI) allows users to customize the display of data as if it were residing on their mobile device. Corporate-owned devices are purchased and owned by the enterprise.
- Several risks are associated with using mobile devices. Mobile devices are used in a wide variety of locations outside of the organization's normal physical perimeter. Devices can easily be lost or stolen, and any unprotected data on the device can be retrieved by a thief. As mobile devices age, they may no longer receive security updates. Geolocation, or the process of identifying the geographical location of a device, can be helpful but also is a security risk because it can identify the location of a person carrying a mobile device. Video cameras and microphones on mobile devices have been used by attackers to secretly "spy" on an unsuspecting victim. Vulnerabilities in mobile device connections can also be exploited by threat actors.
- Mobile devices have the ability to access untrusted content that other types of computing devices generally do not have. Users can circumvent the installed built-in limitations on their smartphone (jailbreaking on Apple iOS devices or rooting on Android devices) to download from an unofficial third-party app store (sideloading) or even write their own custom firmware to run on their device. Because these apps have not been approved, they may contain security vulnerabilities or even malicious code. Other means by which untrusted content can invade mobile devices include short message service (SMS), multimedia messaging service (MMS), and rich communication services (RCS).
- Users should consider security when initially setting up a mobile device. Mobile devices have options for configuring different types of passcodes that must be entered as authentication credentials. Although passwords are the most secure option, many users instead opt for a weaker personal identification number (PIN) or fingerprint reader, or they draw or swipe a specific pattern connecting dots to unlock the device. Although later versions of both iOS and Android encrypt all data on mobile devices, threat actors can access mobile device data that has been backed up to Apple or Google servers. Personal and corporate data can be separated into different containers and each can be managed appropriately. If a mobile device is lost or stolen, several security features can be used to locate the device or limit the damage.
- Several support tools can facilitate the management of mobile devices in the enterprise. Mobile device management (MDM) tools allow a device to be managed remotely by an organization. Mobile application management (MAM) covers application management, which comprises the tools and services responsible for distributing and controlling access to apps. A mobile content management (MCM) system provides content management to mobile devices used by employees in an enterprise. All of the capabilities in MDM, MAM, and MCM can be supported by unified endpoint management (UEM).
- Embedded and specialized devices can be classified into several categories. Hardware and software components are easily available for an industrious user to create their own specialized device. The Raspberry Pi

is a low-cost credit-card-sized computer motherboard with hardware ports that can connect to a range of peripherals. Unlike the Raspberry Pi, which can function as a complete computer, the Arduino is designed as a controller for other devices. A field-programmable gate array (FPGA) is a hardware integrated circuit (IC) that can be programed by the user. A system on a chip (SoC) combines all the required electronic circuits of the various computer components on a single IC chip. SoCs often use a real-time operating system (RTOS) that is specifically designed for an SoC in an embedded or specialized system.

- Specialized systems designed for specific applications include a smart meter, which is a digital meter for measuring the consumption of utilities. Industrial control systems (ICSs) control locally or at remote locations by collecting, monitoring, and processing real-time data so that machines can directly control devices such as valves, pumps, and motors without the need for human intervention. Multiple ICSs are managed by a larger supervisory control and data acquisition (SCADA) system. Heating, ventilation, and air conditioning (HVAC) environmental systems provide and regulate heating and cooling. A multifunctional device (MFD) combines the functions of a printer, copier, scanner, and fax machine. An unmanned aerial vehicle (UAV), commonly known as a drone, is an aircraft without a human pilot on board to control its flight. Voice over IP (VoIP) uses a data-based IP network to add digital voice clients and new voice applications onto the IP network. The Internet of Things (IoT) is connecting any device to the Internet for the purpose of sending and receiving data to be acted upon.
- Security in embedded systems and specialized devices is lacking and can result in a wide range of attacks. Several constraints make security a challenge for these systems. To address security in these devices, governments have begun to propose or enact legislation to require stronger security on embedded systems and specialized devices.

Key Terms

application management
Arduino
bring your own device (BYOD)
carrier unlocking
cellular telephony
choose your own device (CYOD)
constraints
containerization
content management
context-aware authentication
corporate owned, personally enabled (COPE)
corporate owned
custom firmware
drone
embedded system
external media access
field-programmable gate array (FPGA)
firmware OTA updates
full disk encryption
geofencing
geolocation
Global Positioning System (GPS)
GPS tagging (geo-tagging)
heating, ventilation, and air conditioning (HVAC)
hotspot
industrial control systems (ICS)
infrared
Internet of Things (IoT)
jailbreaking
malicious flash drive
malicious USB cable
mobile application management (MAM)
mobile content management (MCM)
mobile device management (MDM)
multifunctional printer (MFP)
multimedia messaging service (MMS)
personal identification number (PIN)
push notification services
Raspberry Pi
real-time operating system (RTOS)
remote wipe
rich communication services (RCS)
rooting
screen lock
short message service (SMS)
sideloading
smart meters
storage segmentation
supervisory control and data acquisition (SCADA)
system on a chip (SoC)
tethering
third-party app store
unified endpoint management (UEM)
Universal Serial Bus (USB) connectors
unmanned aerial vehicle (UAV)
USB On-the-Go (OTG)
virtual desktop infrastructure (VDI)
voice over IP (VoIP)

Review Questions

1. Akira is explaining to his team members the security constraints that have made it a challenge for protecting a new embedded system. Which of the following would Akira NOT include as a constraint?
   a. Authentication
   b. Cost
   c. Power
   d. Availability

2. Agape has been asked to experiment with different hardware to create a controller for a new device on the factory floor. She needs a credit-card-sized motherboard that has a microcontroller instead of a microprocessor. Which would be the best solution?
   a. Arduino
   b. Raspberry Pi
   c. SoC
   d. FPGA

3. Hakaku needs a tool with a single management interface that provides capabilities for managing and securing mobile devices, applications, and content. Which tool would be the best solution?
   a. MCCM
   b. MDM
   c. UEM
   d. MMAM

4. In her job interview, Xiu asks about the company policy regarding smartphones. She is told that employees may choose from a limited list of approved devices but that she must pay for the device herself; however, the company will provide her with a monthly stipend. Which type of enterprise deployment model does this company support?
   a. CYOD
   b. COPE
   c. BYOD
   d. Corporate owned

5. Aoi has been asked to provide research regarding adding a new class of Android smartphones to a list of approved devices. One of the considerations is how frequently the smartphones receive firmware OTA updates. Which of the following reasons would Aoi NOT list in her report as a factor in the frequency of Android firmware OTA updates?
   a. OEMs are hesitant to distribute Google updates because it limits their ability to differentiate themselves from competitors if all versions of Android start to look the same through updates.
   b. Because many of the OEMs have modified Android, they are reluctant to distribute updates that could potentially conflict with their changes.
   c. Wireless carriers are reluctant to provide firmware OTA updates because of the bandwidth the updates consume on their wireless networks.
   d. Because OEMs want to sell as many devices as possible, they have no financial incentive to update mobile devices that users would then continue to use indefinitely.

6. What is the process of identifying the geographical location of a mobile device?
   a. Geotracking
   b. Geolocation
   c. GeoID
   d. Geomonitoring

7. Which of these is used to send SMS text messages to selected users or groups of users?
   a. Pull notification services
   b. Replay notification distribution (RND)
   c. Push notification services
   d. MAM mass SMS

8. Enki received a request by a technician for a new subnotebook computer. The technician noted that he wanted USB OTG support and asked Enki's advice regarding it. Which of the following would Enki NOT tell him?
   a. A device connected via USB OTG can function as a peripheral for external media access.
   b. A device connected via USB OTG can function as a host.
   c. USB OTG is only available for connecting Android devices to a subnotebook.
   d. Connecting a mobile device to an infected computer using USB OTG could allow malware to be sent to that device.

9. Banko's sister has just downloaded and installed an app that allows her to circumvent the built-in limitations on her Android smartphone. What is this called?
   a. Rooting
   b. Sideloading
   c. Jailbreaking
   d. Ducking

10. Which of the following technologies can convert a texting app into a live chat platform?
- **a.** MMS
- **b.** QR
- **c.** SMS
- **d.** RCS

11. What prevents a mobile device from being used until the user enters the correct passcode?
- **a.** Swipe identifier (SW-ID)
- **b.** Screen lock
- **c.** Screen timeout
- **d.** Touch swipe

12. Hisoka is creating a summary document for new employees about their options for different mobile devices. One part of his report covers encryption. What would Hisoka NOT include in his document?
- **a.** All modern versions of mobile device OS encrypt all user data by default.
- **b.** Encryption occurs when the mobile device is locked.
- **c.** Apple uses file-based encryption to offer a higher level of security.
- **d.** Data backed up to an Apple or Google server could be unlocked by a court order.

13. What does containerization do?
- **a.** It splits operating system functions only on specific brands of mobile devices.
- **b.** It places all keys in a special vault.
- **c.** It slows down a mobile device to half speed.
- **d.** It separates personal data from corporate data.

14. What allows a device to be managed remotely?
- **a.** Mobile device management (MDM)
- **b.** Mobile application management (MAM)
- **c.** Mobile resource management (MRM)
- **d.** Mobile wrapper management (MWM)

15. Which of these is NOT a security feature for locating a lost or stolen mobile device?
- **a.** Remote lockout
- **b.** Last known good configuration
- **c.** Alarm
- **d.** Thief picture

16. What enforces the location in which an app can function by tracking the location of the mobile device?
- **a.** Location resource management
- **b.** Geofencing
- **c.** GPS tagging
- **d.** Graphical Management Tracking (GMT)

17. Which of these is considered the strongest type of passcode to use on a mobile device?
- **a.** Password
- **b.** PIN
- **c.** Fingerprint swipe
- **d.** Draw connecting dots pattern

18. Which of the following is NOT a context-aware authentication?
- **a.** On-body detection
- **b.** Trusted places
- **c.** Trusted devices
- **d.** Trusted contacts

19. Which tool manages the distribution and control of apps?
- **a.** MAM
- **b.** MDM
- **c.** MCM
- **d.** MFM

20. Which type of OS is typically found on an embedded system?
- **a.** SoC
- **b.** RTOS
- **c.** OTG
- **d.** COPE

Hands-On Projects

CAUTION If you are concerned about installing any of the software in these projects on your regular computer, you can instead use the Windows Sandbox or install the software in the Windows virtual machine created in the Module 1 Hands-On Projects. Software installed within the virtual machine will not impact the host computer.

Project 5-1: Creating and Using QR Codes

Time Required: 15 minutes

Objective: Given a scenario, implement secure mobile solutions.

Description: Quick Response (QR) codes can be read by an imaging device such as a mobile device's camera or online. However, they pose a security risk. In this project, you create and use QR codes.

1. Use your web browser to go to **www.qrstuff.com**. (If you are no longer able to access the program through this URL, use a search engine to search for "Qrstuff.")
2. First create a QR code. Under **DATA TYPE**, be sure that **WEBSITE URL** is selected.
3. Under **CONTENT**, enter the URL **www.cengage.com**. Watch how the **QR CODE PREVIEW** changes as you type.
4. Under **Encoding Options**, select **Static-Embed URL into code as-is**.
5. Under **QR CODE PREVIEW**, click **DOWNLOAD QR CODE** to download an image of the QR code.
6. Navigate to the location of the download and open the image. Is there anything you can tell by looking at this code? How could threat actors use this to their advantage? Where could malicious QR codes be used? Is there any protection for the user when using QR codes?
7. Now use an online reader to interpret the QR code. Use your web browser to go to **blog.qr4.nl/Online-QR-Code_Decoder.aspx**. (The location of content on the Internet may change without warning. If you are no longer able to access the program through this URL, use a search engine and search for "Free Online QR Code Reader.")
8. Click **Choose File**.
9. Navigate to the location of the QR code that you downloaded on your computer and click **Open**.
10. Click **Upload**.
11. What does the text box display? How could an attacker use a QR code to direct a victim to a malicious website?
12. Use your web browser to go to **qrcode-monkey.com**. (If you are no longer able to access the program through this URL, use a search engine to search for "qrcodemonkey.")
13. Click **LOCATION**.
14. On the map, drag the pointer to an address with which you are familiar. Note how the **Latitude** and **Longitude** change.
15. Click **Create QR Code**.
16. Click **Download PNG** to download this QR code to your computer.
17. Navigate to the location of the download and open the image. How does it look different from the previous QR code? Is there anything you can tell by looking at this code?
18. Use your web browser to return to **blog.qr4.nl/Online-QR-Code_Decoder.aspx**.
19. Click **Choose File**.
20. Navigate to the location of the map QR code that you downloaded on your computer and click **Open**.
21. Click **Upload**.
22. In the text box, a URL will be displayed. Paste this URL into a web browser.
23. What does the browser display? How could an attacker use this for a malicious attack?
24. Return to **www.qrstuff.com**.
25. Click each option under **DATA TYPE** to view the different items that can be created by a QR code. Select three and indicate how they could be used by an attacker.
26. Close all windows.

Project 5-2: Using Software to Locate a Missing Laptop

Time Required: 20 minutes

Objective: Given a scenario, implement secure mobile solutions.

Description: If a mobile device is lost or stolen, there are several security features that can be used to locate the device or limit the damage. Many of these can be used through an installed third-party app. In this project, you download and install software that can locate a missing laptop computer. Note that for this project, a portable computer or desktop computer can be used.

1. Open your web browser and enter the URL **preyproject.com**. (The location of content on the Internet may change without warning. If you are no longer able to access the program through this URL, use a search engine and search for "Prey Project.")

2. Click **Features** and then **Tracking and Location**.
3. Read through the information so you will understand what Prey does.
4. Click **Download**.
5. Select the latest version for your computer.
6. When the file finishes downloading, run the program and follow the default installation procedures.
7. Click **Finish** to configure the Prey settings.
8. Be sure that **New user** is selected. Click **Next**.
9. Enter your information to create an account and click **Create**.
10. Go to **panel.preyproject.com**.
11. Enter your login information, and on the **All your devices** page, click the name of your recently added device.
12. You will then receive in your email information about this added device.
13. Notice that your device is shown on a map regarding its current location. How accurate is this?
14. Click **Hardware information**. How accurate is this information?
15. Click **Map and actions**.
16. Under **Actions** in the right pane, click **Sound alarm**. Read the pop-up window about this function. How would it be useful? Depending upon your setting, either click **Close** (to cancel this) or **Confirm** (to sound the alarm).
17. Under **Actions** in the right pane, click **Send message**. Click **Confirm**. Notice that the message appears on the screen. Is this wording strong enough to compel the person to return a missing laptop? Should a reward be offered? How would you frame this message?
18. Close the message.
19. Under **Actions** in the right pane, click **Lock device**. Read about the function this will perform. How useful would this be? Depending upon your setting, either click **Close** (to cancel this) or **Confirm** (to lock the device).
20. Click **Activity log**. Read through the list of events that have occurred.
21. Click **Set device to missing** and read through what occurs when this is selected. Click **Advanced Options**. How helpful would it be to have a photo of the person who is using the device and a screen capture of what they are doing on the device?
22. Click **Yes, my device is missing**. It may take up to 10 minutes for the alarm to sound depending on how frequently the device checks into Prey.
23. When a report is generated, click **Reports**, and read the information about the location of the device. Would this be sufficient information to find the missing device?
24. Change the settings so that your device is no longer registered as missing.
25. Close all windows.

Project 5-3: Installing BlueStacks Android Emulator

Time Required: 20 minutes

Objective: Given a scenario, implement secure mobile solutions.

Description: In this project, you install an Android emulator on a personal computer to test different tools. Note that you will need a Google account to access these tools.

1. Use your web browser to go to **www.bluestacks.com**. (The location of content on the Internet may change without warning. If you are no longer able to access the program through this URL, use a search engine and search for “BlueStacks.”)
2. Click **Download BlueStacks**.
3. When the download is complete, launch the installation file and accept the defaults to install BlueStacks.
4. Click **Sign in with Google**.
5. Enter your Google account information.
6. Click **Done**.
7. Click the right arrow in the window.
8. Click **Continue**.
9. If necessary, sign in with your Google account.
10. Click **OK**.
11. Click the right arrow in the window.

12. If necessary, personalize the information and click the right arrow in the window.
13. Click **Finish**.
14. Click **Got it**.
15. Remain in BlueStacks for the next project.

Project 5-4: Installing Security Apps Using BlueStacks

Time Required: 20 minutes
Objective: Given a scenario, implement secure mobile solutions.
Description: In this project, you download and install Android apps to test different antimalware tools.

1. Click **Search apps**.
2. In the search bar enter **Antivirus**.
3. Select **Norton Security**.
4. Click **Install**.
5. Click **Accept**.
6. After the app has installed, click **OPEN**.
7. If the button **Agree and launch** appears, click it to start the app.
8. After the app has loaded and scanned, click through the various options for this app. Would you consider this antivirus app one that you would use on your Android device?
9. Click **Play Store**.
10. Click the magnifying glass to start another search.
11. In the search bar, enter **Antivirus**.
12. This time select a different antivirus product. Install it and then view the different options.
13. Click **Play Store**.
14. Click the magnifying glass to start another search.
15. Enter **Security** and press **Enter**.
16. Scroll down through the different apps available.
17. Select a different security app and install it.
18. How easy were these apps to install and configure? How do they compare with comparable desktop antimalware apps?
19. Close all windows.

Case Projects

Case Project 5-1: Unified Endpoint Management Tools

Use the Internet to identify and compare three different unified endpoint management tools. Create a table that lists their various features. Which of the tools would you recommend for a small business with 10 employees who use smartphones but has a single person managing IT services? Why?

Case Project 5-2: Enterprise Deployment Model Comparison

Research the different enterprise deployment models listed in Table 5-3. Create a detailed table listing their typical features, how they are used, and their advantages and disadvantages to both the enterprise as well as to the employee. Which of them is the most secure option? Which is the least secure option? Which of them is most advantageous for the enterprise? Which would you prefer to use? Which would you recommend for your school or place of employment? Why? Create a one-paragraph summary along with your table.

Case Project 5-3: Raspberry Pi and Arduino

Research information on the Raspberry Pi and the Arduino. What are the strengths of each platform? What are the weaknesses? How much do they cost? What additional peripherals are needed for each platform? How difficult are they to program and use? What are some interesting uses for each? Create a one-page document of your comparisons.

Case Project 5-4: Rooting and Jailbreaking

Research Android rooting and Apple jailbreaking. What privileges can be obtained by rooting and jailbreaking? What are the advantages? What are the disadvantages? Can a device that has been broken return to its default state? If so, how? Finally, create a list of at least five reasons why rooting and jailbreaking are considered harmful in a corporate environment.

Case Project 5-5: MFPs

MFPs are widely used in corporate environments. Use the Internet to research MFPs. Identify three MFPs, and list their features. What are the security risks of an MFP? How should they be protected? Write a one-page paper on your research.

Case Project 5-6: Internet of Things

Use the Internet to research the Internet of Things (IoT). In your own words, what is IoT? How is it being used today? How will it be used in the near future? What impact will IoT have on technology, society, and the economy over the next five years? What are its advantages and disadvantages? Finally, identify five of the most unusual IoT devices that you can find described online. Write a one-page paper on the information that you find.

Case Project 5-7: North Ridge Security Consulting

North Ridge Security provides security consulting and assurance services to more than 500 clients in more than 20 states for a wide range of enterprises. A new initiative at North Ridge is for each of its seven regional offices to provide internships to students who are in their final year of the security degree program at the local college.

The Carlyle-Stedman Museum provides patrons with mobile devices that contain prerecorded information that can be listened to while viewing the museum's artifacts. Recently, an incident occurred in which a patron circumvented the security on the device and, because it was not examined after it was turned in, the next patron who tried to use it was exposed to inappropriate content. The executive board of Carlyle-Stedman decided that something must be done to prevent this from recurring and wants to ensure that all employee mobile devices are also secure. They have asked North Ridge to make a presentation about mobile device security, and you have been given this assignment.

1. Create a PowerPoint presentation for the staff about the security risks of mobile technology and steps to be taken to secure mobile devices. Be sure to cover these from the perspective of the organization, the IT department, and the user. Your presentation should contain at least eight slides.
2. After the presentation, the IT director at the museum has asked North Ridge for recommendations on using MDM, MAM, MCM, and/or UEM. Write a one-page memo listing the features of these tools and how they could be used to help the museum.

Case Project 5-8: Information Security Community Site Activity

The Information Security Community Site is an online companion to this textbook. It contains a wide variety of tools, information, discussion boards, and other features to assist learners. Go to *community.cengage.com/infosec2* and click the *Join or Sign in* icon to log in, using your login name and password that you created in Module 1. Click **Forums (Discussion)** and click **Security+ Case Projects (7th edition)**. Read the following case study.

Read again the information in the module regarding the security risks of data-in-transit and remote data-at-rest. What are your feelings regarding the ability of the government to access data on iCloud and Google servers? Do you believe this is a safeguard for the nation, or is it a serious violation of personal privacy? What are the advantages and the risks of a government that has these powers? How could they be abused? Post your thoughts about app data sharing on the discussion board.

References

1. “Mobile fact sheet,” *Pew Research Center*, Jun. 12, 2019, accessed May 26, 2020, www.pewresearch.org/internet/fact-sheet/mobile/.
2. “Mobile vs. desktop usage (latest 2020 data)," *BroadbandSearch*, accessed May 26, 2020, www.broadbandsearch.net/blog/mobile-desktop-internet-usage-statistics.
3. “Mobile security Index - 2020 report,” *Verizon Wireless*, accessed May 26, 2020, https://enterprise.verizon.com/resources/reports/2020-msi-report.pdf.
4. Constantin, Lucian, “One In three organizations suffered data breaches due to mobile devices,” *CSO,* Mar. 5, 2019, accessed May 26, 2020, www.csoonline.com/article/3353560/one-in-three-organizations-suffered-data-breaches-due-to-mobile-devices.html.
5. “Survey: IT Security and Laptop Theft,” *Kensington,* Aug. 2016, accessed May 15, 2017, www.kensington.com/a/283005.
6. “Laptop and mobile device theft awareness,” *University of Pittsburgh*, accessed May 27, 2020, www.technology.pitt.edu/security/laptop-theft.
7. “Data gets personal: 2019 global data risk report from the Varonis data lab,” *Veronis*, accessed May 27, 2020, www.varonis.com/2019-data-risk-report/.
8. “Visa Tech Matters,” *Visa*, Feb. 12, 2015, accessed May 26, 2017, http://visacorporate.tumblr.com/post/110835709353/visatechmatters-visa-launches-mobile-location.
9. “Pin analysis,” *DataGenetics*, accessed Mar. 10, 2014, http://datagenetics.com/blog/september32012/index.html.
10. “Overview of the Internet of things, Series y: Global information infrastructure, internet protocol aspects and next-generation networks: Next Generation Networks; Frameworks and functional architecture models,” Jun. 2012, Retrieved May 18, 2017, www.itu.int/rec/T-REC-Y.2060-201206-I.
11. “Forecast number of IoT connected objects worldwide from 2018 to 2025, by type,” *Statista*, retrieved May 28, 2020, www.statista.com/statistics/976079/number-of-iot-connected-objects-worldwide-by-type/.
12. Kushner, David, “The real story of Stuxnet,” *IEEE Spectrum*, Feb. 26, 2013, retrieved May 26, 2017, http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet.