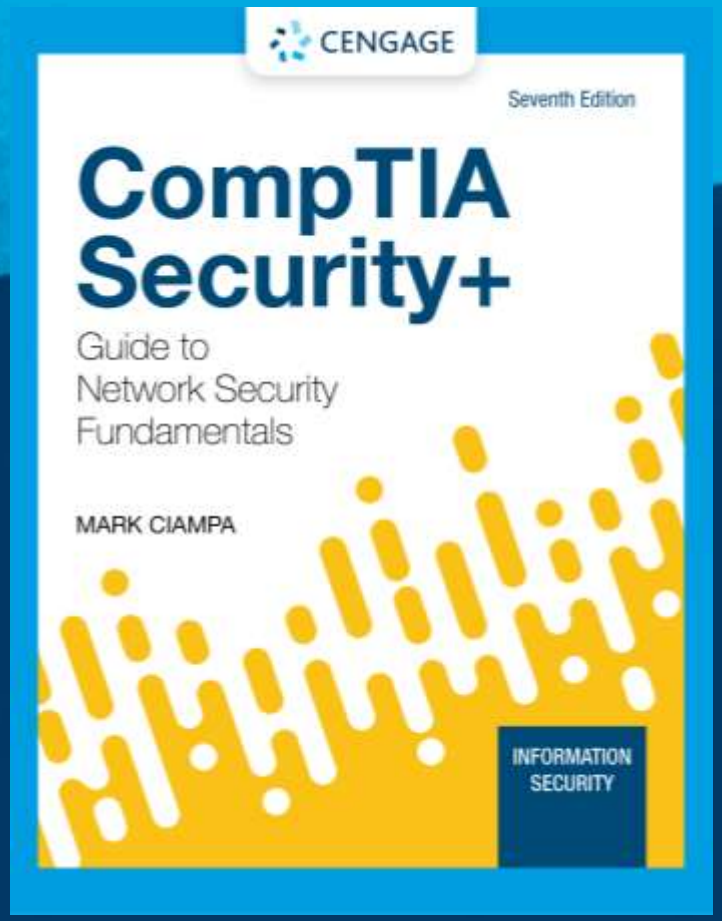


CompTIA Security+ Guide to Network Security Fundamentals, 7th Edition

Module 4: Endpoint and Application Development Security

CENGAGE

Module Objectives

By the end of this module, you should be able to:

1. Describe different threat intelligence sources

2. List the steps for securing an endpoint

3. Explain how to create and deploy SecDevOps

Threat Intelligence Sources

- Organizations are now pooling resources and knowledge about the latest attacks with the broader security community
- One type of shared information is the evidence of an attack
- *Key risk indicators* (*KRIs*) are metrics 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 an **indicator of compromise** (**IOC**)
  - An IOC shows that a malicious activity is occurring but is still in the early stages of an attack
  - IOC information aids others in their predictive analysis or discovering an attack before it occurs

Categories of Sources (1 of 3)

- Two categories of threat intelligence sources are open source and closed source
- Open Source Information
  - "open source" refers to anything that could be freely used without restrictions
  - Open source threat intelligence information is often called open source intelligence (OSINT)
  - Cyber Information Sharing and Collaboration Program (CISCP) enables actionable, relevant, and timely unclassified information exchange through partnerships
  - CISP services include:
    - *Analyst-to-analyst technical exchanges*
    - *CISCP analytical products*
    - *Cross industry orchestration*
    - *Digital malware analysis*

Categories of Sources (2 of 3)

- Two concerns around public information sharing centers are:
  - 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
  - Speed – **Automated Indicator Sharing** (**AIS**) enables the exchange of cyberthreat indicators between parties through computer-to-computer communication
    - Two tools facilitate AIS:
      - **Structured Threat Information Expression** (**STIX**) is a language and format used to exchange cyberthreat intelligence
      - **Trusted Automated Exchange of Intelligence Information** (**TAXII**) is an application protocol for exchanging cyberthreat intelligence over HTTPS

Categories of Sources (3 of 3)

- Closed Source Information
  - **Closed source** is proprietary
  - Organizations that are participants in closed source information are part of private information sharing centers that restrict both access to data and participation
  - All candidates must go through a vetting process and meet certain criteria

Sources of Threat Intelligence (1 of 3)

- Sources of threat intelligence that are useful:
  - *Vulnerability database* is a repository of known vulnerabilities and information as to how they have been exploited
  - *Threat maps* illustrate cyberthreats overlaid on a diagrammatic representation of a geographical area
  - *File and code repositories* are where victims of an attack can upload malicious files and software code that can be examined by others to learn more about the attacks and craft their defenses
  - *Dark web* – 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

Sources of Threat Intelligence (2 of 3)

Figure 4-1 Threat map

Figure 4-1 Threat map

Sources of Threat Intelligence (3 of 3)

Figure 4-2 Dark web

Figure 4-2 Dark web

Knowledge Check Activity 1

What is the significance of a KRI exceeding its normal bounds?

a. It must be referred to the DHS.
b. It could be an IOC.
c. It probably contains a TTP.
d. An AIS should be generated.

Knowledge Check Activity 1: Answer

What is the significance of a KRI exceeding its normal bounds?

Answer: b. It could be an IOC.

A key risk indicator (KRI) exceeding its normal bounds could be an indicator of compromise (IOC).

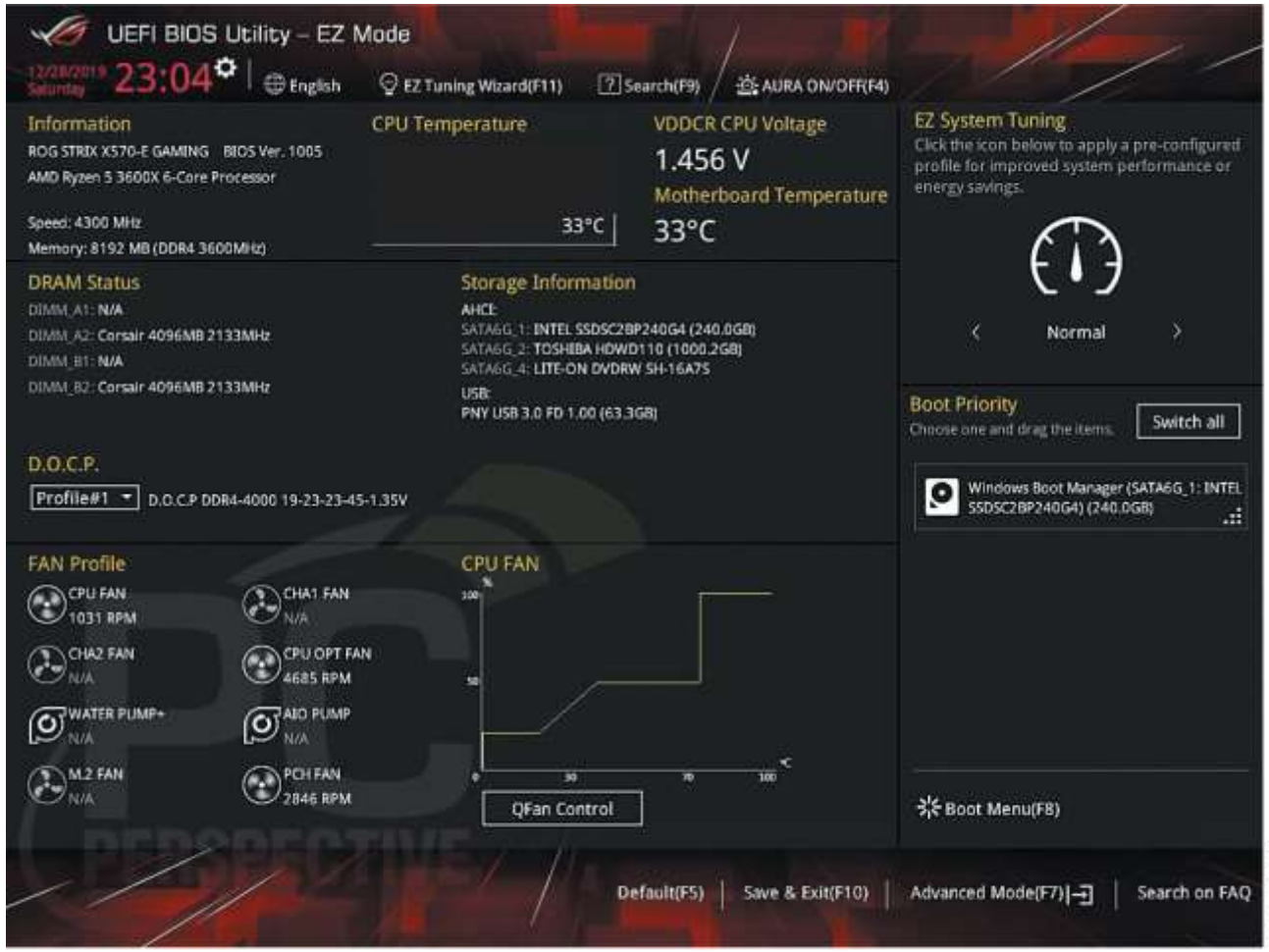
Securing Endpoint Computers

- Securing endpoint computers primarily involves three major tasks:
  - *Confirming* that the computer has started securely
  - *Protecting* the computer from attacks
  - *Hardening* it for even greater protection

Confirm Boot Integrity (1 of 3)

- Ensuring secure startup involves the Unified Extensible Firmware Interface (UEFI) and its boot security features
- Unified Extensible Firmware Interface (UEFI)
  - Early booting processes used firmware called the *BIOS* (*Basic Input/Output System*)
  - To add functionality, an improved firmware interface was developed to replace BIOS
  - **UEFI** includes:
    - The ability to access hard drives that are larger than 2TB
    - Support for an unlimited number of primary hard drive partitions
    - Faster booting
    - Support for networking functionality in the UEFI firmware itself to aid in troubleshooting

Confirm Boot Integrity (2 of 3)



Source: ASUS

Figure 4-3 UEFI user interface

Figure 4-3 UEFI user interface

Confirm Boot Integrity (3 of 3)

- Boot Security
  - The ability to update the BIOS in firmware opened the door for a threat actor to create malware to infect the BIOS (called a *BIOS attack*)
  - Boot security involves validating that each element used in each step of the boot process has not been modified
  - This process begins with validation of the boot software, then it can validate the software drivers, and so on until control has been handed over to the OS
    - Called *chain of trust* because each element relies on the confirmation of the previous element to know that the entire process is secure
    - The strongest starting point is hardware, which cannot be modified like software (known as **hardware root of trust**)

Protect Endpoints (1 of 4)

- Protection on computer endpoints can be accomplished through software installed on the endpoint, such as:
  - Antivirus software, antimalware, web browser protections, and monitoring and response systems
- Antivirus
  - **Antivirus** (**AV**) software can examine a computer for file-based virus infections and monitor computer activity (such as scanning new documents that might contain a virus)
  - Log files created by AV products can provide beneficial info regarding attacks
  - Many AV products use signature-based monitoring, called *static analysis*
  - A newer approach to AV is heuristic monitoring, called *dynamic analysis*

Protect Endpoints (2 of 4)

- Antimalware
  - **Antimalware** is a suite of software intended to provide protections against multiple types of malware
  - Antimalware spam protection is often performed using a technique called *Bayesian filtering*
    - Filters by analyzing every word in each email and determines how frequently a word occurs in a spam pile versus a nonspam pile
  - Another component of an antimalware suite is *antispyware*, which helps prevent computers from becoming infected by spyware
    - Uses pop-up blockers, which allow the user to select the level of blocking, ranging from blocking all pop-ups to allowing specific pop-ups

Protect Endpoints (3 of 4)

- Web Browsers
  - Web browsers offer the following security on endpoint computers:
    - Secure cookies are sent to a web 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
    - HTTP Response Header are headers that tell the browser how to behave while communicating with the website

Protect Endpoints (4 of 4)

- Monitoring and Response Systems
  - There are three types of monitoring and response systems for endpoint computers:
    - **Host Intrusion Detection Systems** (**HIDS**) is a software-based application that runs on an endpoint computer and can detect an attack has occurred
    - **Host Intrusion Prevention Systems** (**HIPS**) monitor endpoint activity to immediately block a malicious attack by following specific rules
    - **Endpoint Detection and Response** (**EDR**) tools are considered more robust than HIDS and HIPS
      - An EDR can aggregate data from multiple endpoint computers to a centralized database
      - EDR tools can perform more sophisticated analytics that identify patterns and detect anomalies

Harden Endpoints (1 of 6)

- Hardening endpoints involves patch management and OS protections
- Patch Management
  - Effective patch management involves two types of patch management tools to administer patches
    - Patch distribution using an *automated patch update service*
    - Patch reception in Microsoft Windows 10 includes the following options:
      - *Forced updates*
      - *No selective updates*
      - *More efficient distribution*

Harden Endpoints (2 of 6)

- Operating Systems
  - Securing an OS involves proper security configurations and using confinement tools
  - A typical OS security configuration should include the following:
    - *Disabling unnecessary ports and services*
    - *Disabling default accounts/passwords*
    - *Employing least functionality*
  - In Microsoft Windows, a *security template* is a collection of security configuration settings that can be used to deploy security settings to multiple computers
  - Windows 10 Tamper Protection security feature prevents Windows security settings from being changed or disabled by a threat actor who modifies the registry
    - A Group Policy setting can also *prevent access to registry editing tools* (see Figure 4-7)

Harden Endpoints (3 of 6)

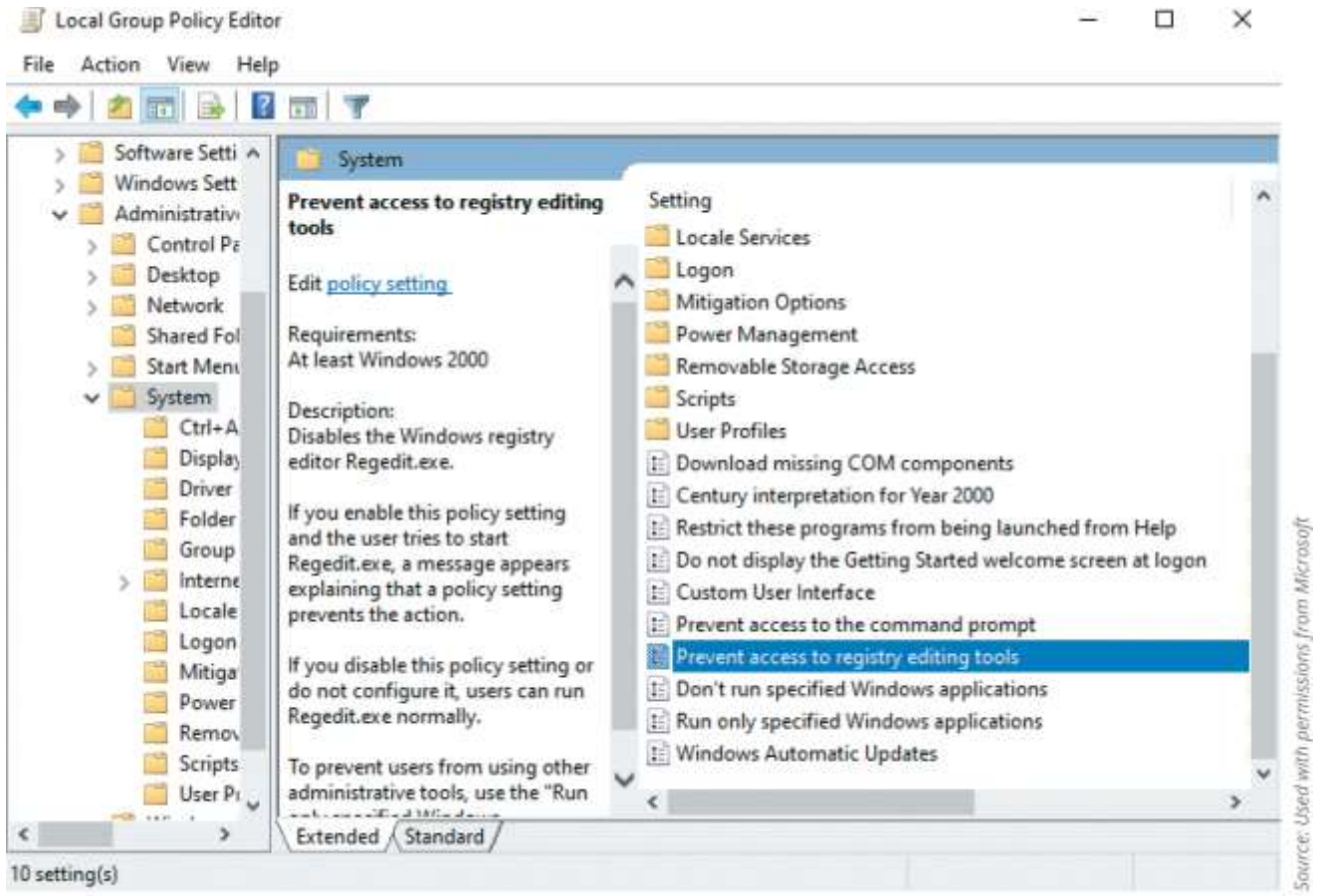


Figure 4-7 Prevent access to registry editing tools

Source: Used with permissions from Microsoft

Figure 4-7 Prevent access to registry editing tools

Harden Endpoints (4 of 6)

- Operating Systems (continued)
  - Confinement Tools – several tools can be used to restrict malware:
    - *Application whitelisting/blacklisting*
    - *Sandbox*
    - *Quarantine*

Harden Endpoints (5 of 6)

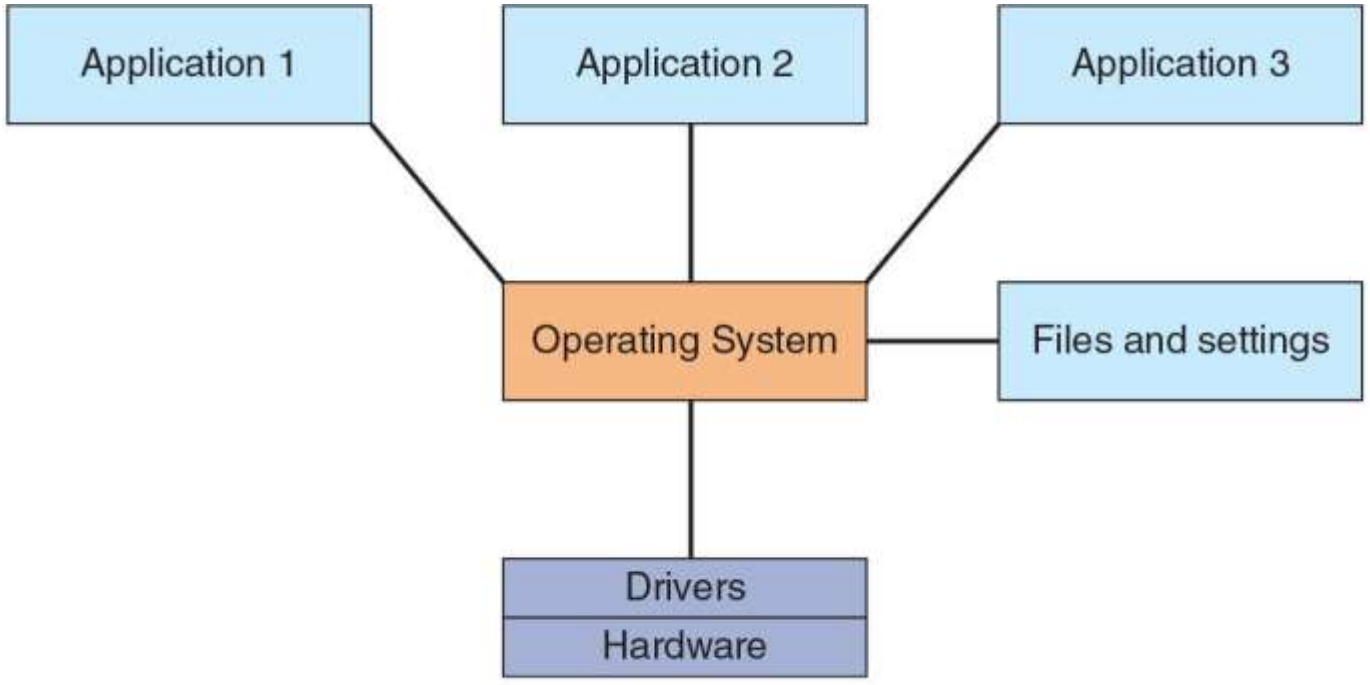


Figure 4-8 Applications interacting with an OS

Figure 4-8 Applications interacting with an OS

Harden Endpoints (6 of 6)

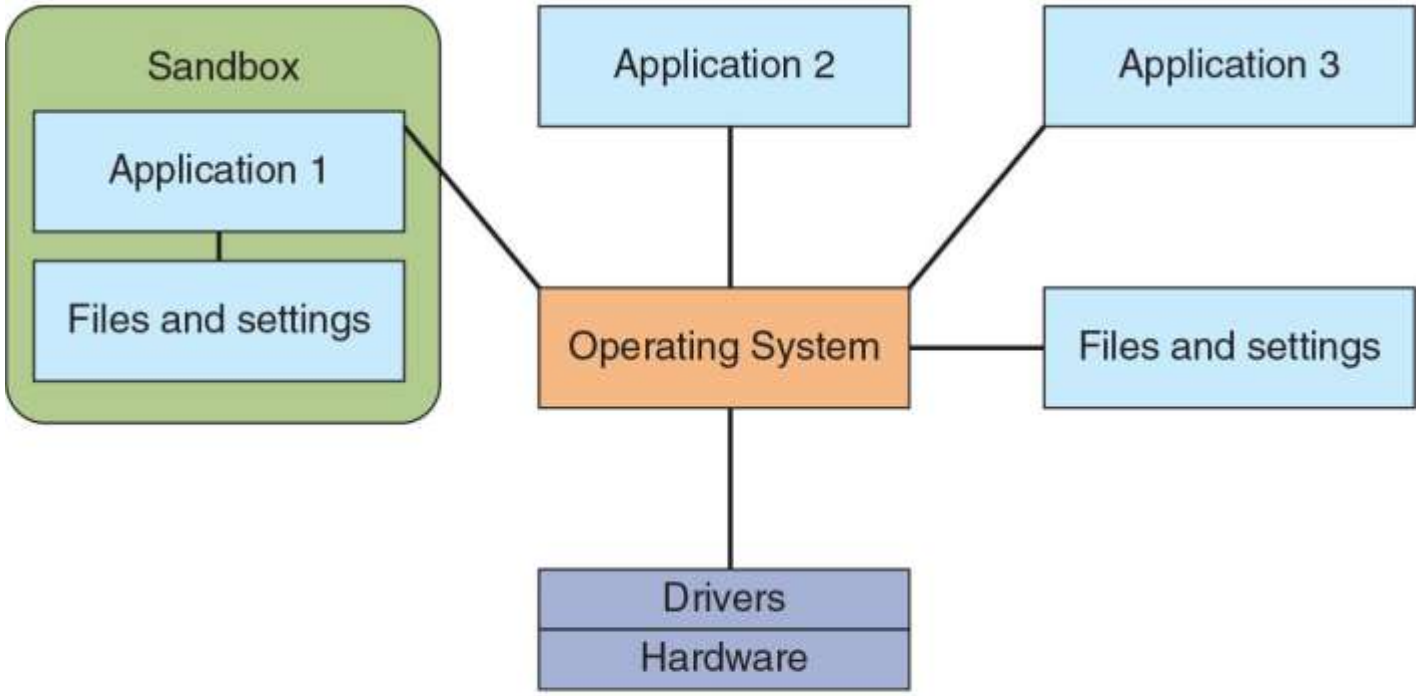


Figure 4-9 Using a sandbox

Figure 4-9 Using a sandbox

Knowledge Check Activity 2

Which of the following is NOT a tool that can be used to confine or restrict malware?

a. Whitelisting
b. Quarantine
c. Sandbox
d. Legacy boot

Knowledge Check Activity 2: Answer

Which of the following is NOT a tool that can be used to confine or restrict malware?

Answer: Legacy boot

A computer configured for legacy boot uses the BIOS to boot the system, which has little or no security features.

Creating and Deploying SecDevOps (1 of 2)

- An unsecure application can open the door for attackers to exploit the application, the data that it uses, and even the underlying OS
- 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 could allow an unauthorized users to view confidential files or enter commands to execute on a server known as *command injection*
- Other dangerous weaknesses in an application can create vulnerabilities in computer memory or buffer areas that can be easily exploited
  - **Poor memory management vulnerabilities** result in attacks such as buffer overflow, integer overflow, pointer/object deference, and DLL injection attacks

Creating and Deploying SecDevOps (2 of 2)

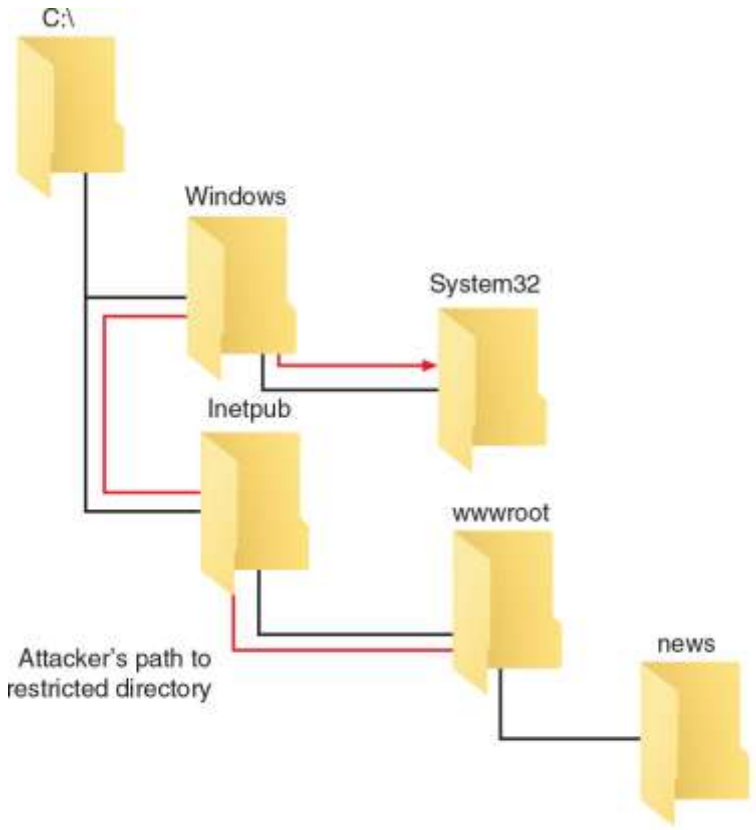


Figure 4-10 Directory traversal attack

Figure 4-10 Directory traversal attack

Application Development Concepts (1 of 3)

- The two levels of application development concepts include general concepts that apply to all application development and those that apply to a rigorous security-based approach
- General Concepts
  - Developing an application requires completing the following stages:
    - *Development*
    - *Testing*
    - *Staging*
    - *Production*
  - **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

Application Development Concepts (2 of 3)

- General Concepts (continued)
  - **Provisioning** is the enterprise-wide configuration, deployment, and management of multiple types of IT system resources
  - **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 that an application is running only a set of known and approved executables
- SecDevOps
  - An *application development lifecycle model* is a conceptual model that describes the stages involved in creating an application and are usually one of the following two:
    - *Waterfall model* – uses a sequential design process
    - *Agile model* – takes an incremental approach

Application Development Concepts (3 of 3)

- SecDevOps (continued)
  - *SecDevOps* is the process of integrating secure development best practices and methodologies into application software development and deployment processes using the agile model
  - SecDevOps applies **automated courses of action** to develop code as quickly and securely as possible
  - This automation enables:
    - **Continuous monitoring**
    - **Continuous validation**
    - **Continuous integration**
    - **Continuous delivery**
    - **Continuous deployment**

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
  - Ensuring that memory management is handled correctly so as not to introduce memory vulnerabilities

Code Testing (1 of 3)

- Testing is one of the most important steps in SecDevOps
- Testing should be performed during the implementation and verification phases of a software development process
- Testing involves static code analysis and dynamic code analysis
- Static Code Analysis
  - **Static code analysis** are tests ran before the source code is even compiled and may be accompanied by manual peer reviews
- Dynamic Code Analysis
  - Security testing performed after the source code is compiled is called **dynamic code analysis**
  - **Fuzzing** is used by dynamic code analysis tools and provides random input to a program in an attempt to trigger exceptions

Code Testing (2 of 3)

Figure 4-11 Automated static code analysis tool

Figure 4-11 Automated static code analysis tool

Code Testing (3 of 3)

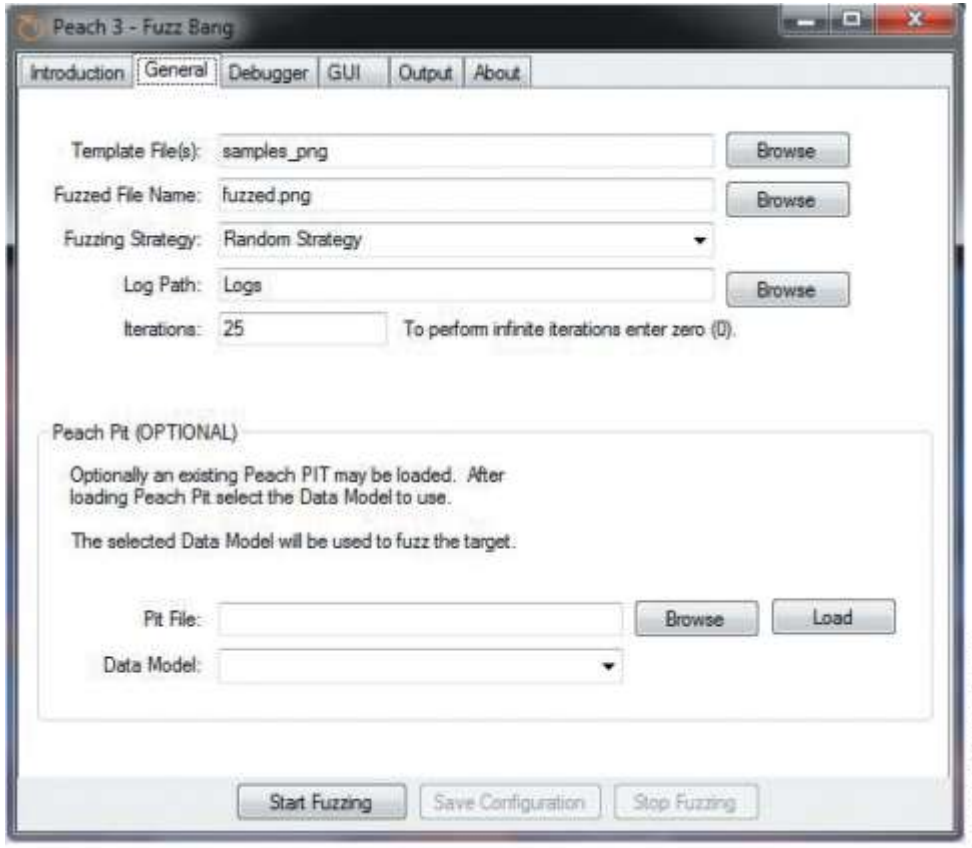


Figure 4-12 Fuzzer input generator

Figure 4-12 Fuzzer input generator

Knowledge Check Activity 3

Which of the following best describes static code analysis?

a. Testing uses a suite of pre-built attacks.
b. Tests are run before the source code is compiled.
c. Random input is used to trigger exceptions.
d. Used after all components are integrated.

Knowledge Check Activity 3: Answer

Which of the following best describes static code analysis?

Answer: b. Tests are run before the source code is compiled.

Static code analysis is performed before the source code is compiled and may be accompanied by manual peer reviews. Dynamic code analysis is performed on a running system.

Self-Assessment

1. Consider the importance of the system boot process in the security of endpoint computers. What are some of the dangers of using legacy boot procedures? Why are computers that use legacy boot firmware more susceptible to attacks?

Summary (1 of 2)

- 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
- Several sources of threat intelligence are useful: a vulnerability database, a cybersecurity threat map, and file and code repositories are examples
- 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
- 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
- Web browsers have a degree of security that can protect endpoint computers
- 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

Summary (2 of 2)

- One of the most important steps in securing an endpoint computer is to promptly install patches
- An unsecure application can open the door for attackers to exploit the application, the data that it uses, and even the underlying OS
- Testing is one of the most important steps in SecDevOps