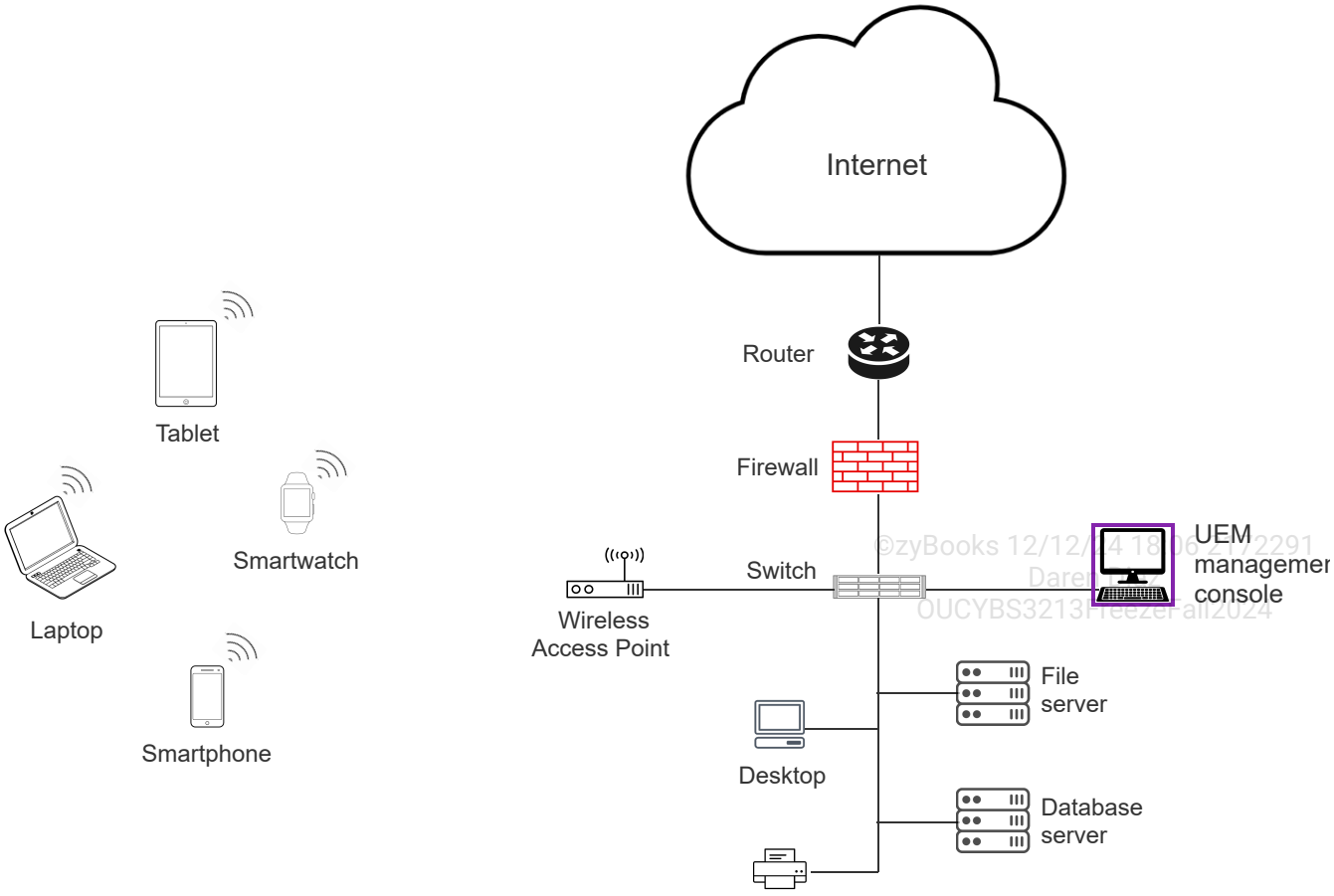
9.1 Endpoint protection: EDR and anti-malware

Endpoint

An ***endpoint*** is a hardware device that is an end point of a wired or wireless network connection. Ex: A network-connected laptop, desktop, smartphone, printer, file server, or specialized hardware such as a sensor or smart meter is an endpoint. An endpoint has firmware, an operating system, and may run application programs. An endpoint is subject to various attack types. An attacker may attempt to gain access to data stored or in process at an endpoint or use the endpoint to connect to a network.

A ***unified endpoint management (UEM)*** is a collection of software tools for controlling and securing endpoints from a single management console. UEM enables the management of endpoints, regardless of the operating system or device type. A UEM has the capability to install firmware and software updates, apply security policies, and remotely remove all applications and data from lost or compromised endpoints.

PARTICIPATION ACTIVITY 9.1.1: Endpoints.



Animation content:

Static image: A laptop, a tablet, a smartphone, and a smartwatch. A cloud labeled "Internet" is connected to a router. The router is connected to a firewall. The firewall is connected to a switch. The switch is connected to a wireless access point, a file server, a database server, a printer, a desktop, and an MDM management console.

Animation captions:

1. An endpoint is a hardware device that is an end point of a network connection. An endpoint can be a wired device, such as a desktop, printer, file server, or database server.
2. An endpoint can be a wireless device, such as a laptop, tablet, smartwatch, or smartphone.
3. A unified endpoint management (UEM) is a collection of software tools for controlling and securing endpoints from a single management console.

PARTICIPATION ACTIVITY 9.1.2: Endpoint protection.

1) A network-connected tablet is an endpoint.
   - False
   - True

2) UEM can control and secure endpoints from a single management console.
   - False
   - True

3) An endpoint has a network interface controller (NIC).
   - False
   - True

4) UEM can only manage endpoints that run the Windows operating system.
   - False

True

Endpoint protection

Endpoint protection, also known as ***endpoint security***, is the set of technologies used to protect an endpoint against various attack types. Endpoint protection technologies include antivirus and anti-malware software, endpoint detection and response (EDR), data loss prevention (DLP), next-generation firewalls (NGFW), host-based firewalls, and host-based intrusion detection and prevention systems (HIDS/HIPS).

Endpoint detection and response (EDR), also known as ***endpoint threat detection and response (ETDR)***, is an endpoint security solution that combines endpoint monitoring and log analysis capabilities with contextual information from correlated network events to detect and respond to security incidents.

Diverse endpoints, including IoT and mobile devices and traditional computing systems, each require tailored security strategies to manage distinct vulnerabilities and threats. Ex: IoT devices like smart meters constrained by limited computational power and the absence of standardized security protocols, necessitate custom-designed firmware updates and frequent vulnerability scans.

Table 9.1.1: Endpoint protection.

Endpoint	Security measures	Example
IoT devices	Customized firmware updates	Enabling automatic firmware updates for smart home security cameras
Mobile devices	Data encryption, authentication protools	Encrypting sensitive data with AES, requiring biometric authentication to access company email
Servers	Access control, regular security updates	Limiting access to authorized personnel only, applying security patches within 24 hours of release
Workstations	Endpoint security solution, regular patch management	Installing and regularly updating endpoint security software on employee workstations

PARTICIPATION ACTIVITY 9.1.3: Endpoint protection.

1) What is endpoint protection?

- A set of technologies to enhance device performance
- A set of technologies used to protect an endpoint against various attack types
- A system for improving software efficiency

2) What is the purpose of EDR?

- Detect, but not respond to security incidents at an endpoint
- Detect and respond to security incidents at an endpoint
- Respond, but not detect security incidents at an endpoint

3) Which of the following is an example of an endpoint protection technology?

- Endpoint detection and response (EDR)
- Customer relationship management system
- Web development tools

4) What is a primary security focus for workstations?

- Regular patch management
- Infrequent data backup
- Occasional software use

5) What distinguishes security measures for traditional computing systems from IoT devices?

- Traditional computing systems require more robust security measures

- IoT devices require more frequent vulnerability scans
- IoT devices require customized security measures due to limited resources and lack of standardized security protocols

PARTICIPATION ACTIVITY 9.1.4: Endpoint protection.

How to use this tool

UEM **EDR** **Endpoint** **Endpoint protection**

	A hardware device that is an end point of a wired or wireless network connection.
	A collection of software tools for controlling and securing endpoints from a single management console.
	A set of technologies used to protect an endpoint against various attack types.
	An endpoint security solution that can detect and respond to security incidents at an endpoint.

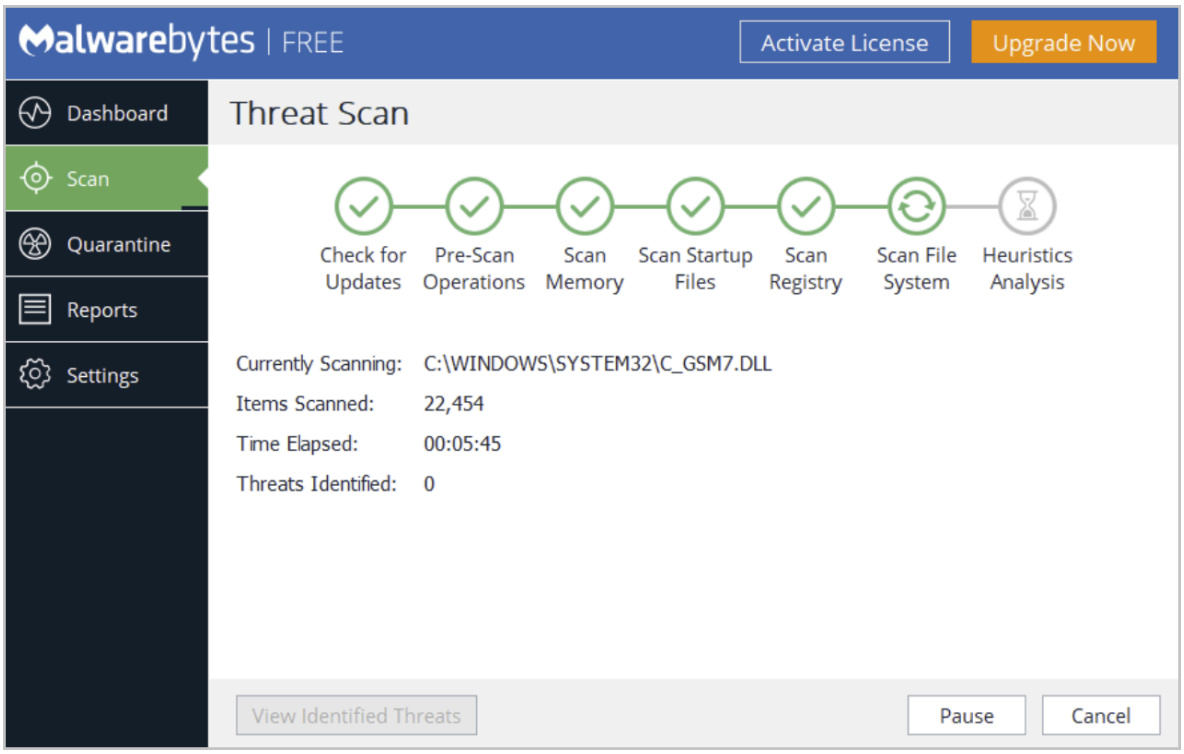
Reset

Malware protection

An endpoint can be infected by malware. ***Malware***, or ***malicious software***, is any software designed to cause damage to a computing device. Different malware types exist, including computer viruses, worms, ransomware, spyware, adware, and Trojan horses. An endpoint can be protected against malware by anti-malware software. An ***anti-malware*** software is a program that can detect and remove malware from a computing device.

A ***computer virus*** is a type of malware that modifies the way a legitimate program operates. A computer virus attaches to another program and has the ability to replicate and spread from one computer to another. An ***antivirus*** program is a program that can detect and remove a computer virus. An antivirus program continuously scans an endpoint to find a virus whose signature matches the signature of a known virus. An antivirus program can prevent a computer virus from being installed on an endpoint, or detect and remove an installed virus from an endpoint.

Example 9.1.1: Malwarebytes anti-malware program scans for malware on a computer.



Credit: Malwarebytes, Inc.[2]

PARTICIPATION ACTIVITY 9.1.5: Malware.

1) What is the purpose of malware?

- repair damages caused by a computer virus
- cause damage to a computing device
- detect network attacks

2) Which one of the following is not malware?

- virus
- worm
- IDS

3) Which malware type attaches to another program and can replicate and spread from one computer to another?

- virus
- worm
- spyware

PARTICIPATION ACTIVITY 9.1.6: Malware.

How to use this tool

Malware **Computer virus** **Antivirus** **Anti-malware**

	A type of malware that modifies the way a legitimate program operates.
	Any software designed to cause damage to a computing device.
	A program that can detect and remove a computer virus.
	A program that can detect and remove malware from a computing device.

Reset

Next generation firewall

*A **next-generation firewall (NGFW)** is a firewall that combines the functionality of a packet filtering firewall with other technologies to detect and block a network attack. A NGFW can enforce security policies at application, port, and protocol levels and can inspect encrypted traffic, detect and remove malware, and perform deep packet inspection (DPI). DPI enables a NGFW to inspect the data within a packet and identify and block a malicious packet. A NGFW may have the ability to act on real-time information provided by threat intelligence services to block new threats.*

(*2) Malwarebytes, Inc. "Malwarebtytes". https://www.malwarebytes.com.

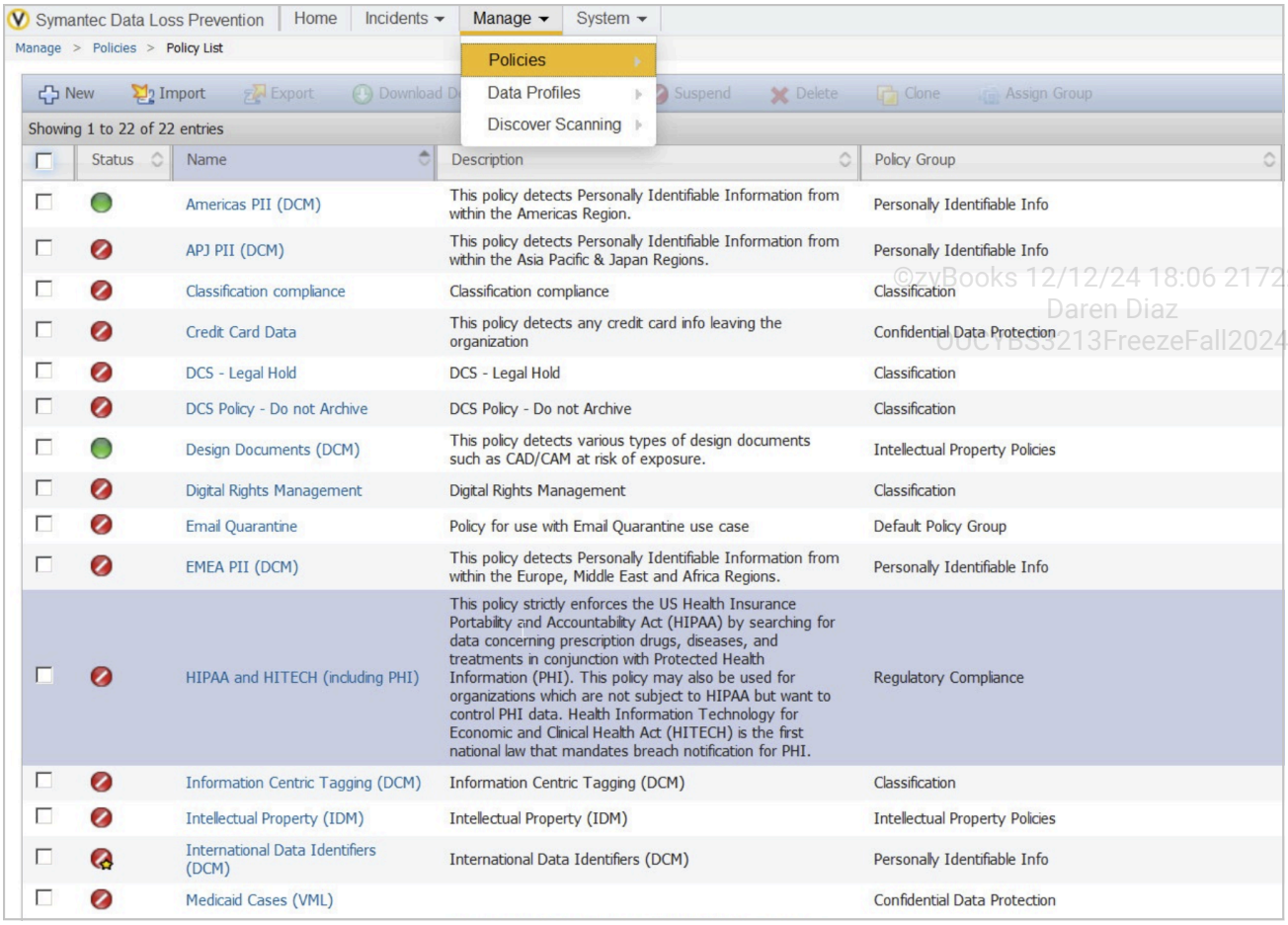
9.2 Endpoint protection: DLP and host-based systems

Data loss prevention (DLP)

Data loss prevention (DLP) refers to technologies and processes used for preventing the accidental or intentional deletion, exposure, or transfer of sensitive organizational data. The main goal of DLP is to ensure an organization is in compliance with laws and regulations that govern the use of personal, financial, and medical information of the organization's customers. Such regulations include the Health Insurance Portability and Accountability Act (HIPAA), the Payment Card Industry Data Security Standard (PCI-DSS), and General Data Protection Regulations (GDPR).

DLP's main functions include data classification and tagging, data policy management and enforcement, data monitoring, and report generation. DLP monitors endpoints and internal networks and performs content inspection and contextual analysis of data to detect activities that may violate data policies defined by an organization. Once a violation is detected, DLP takes protective actions. Ex: DLP prevents an organization's employee from sending an email containing a customer's social security number or from deleting a file containing sensitive information on the organization's finances.

Example 9.2.1: Symantec DLP policies describing the types of data that are monitored and protected to ensure compliance with various laws and regulations.



Credit: Broadcom Inc.[1]

PARTICIPATION ACTIVITY 9.2.1: Data loss prevention (DLP).

1) What is the main goal of DLP?

- protect sensitive organizational data from outside threats
- prevent data sharing within an organization
- ensure sensitive organizational data is not deleted, exposed, or transferred outside an organization

2) Which one of the following functions is not performed by DLP?

- report generation
- endpoint monitoring

- file deletion

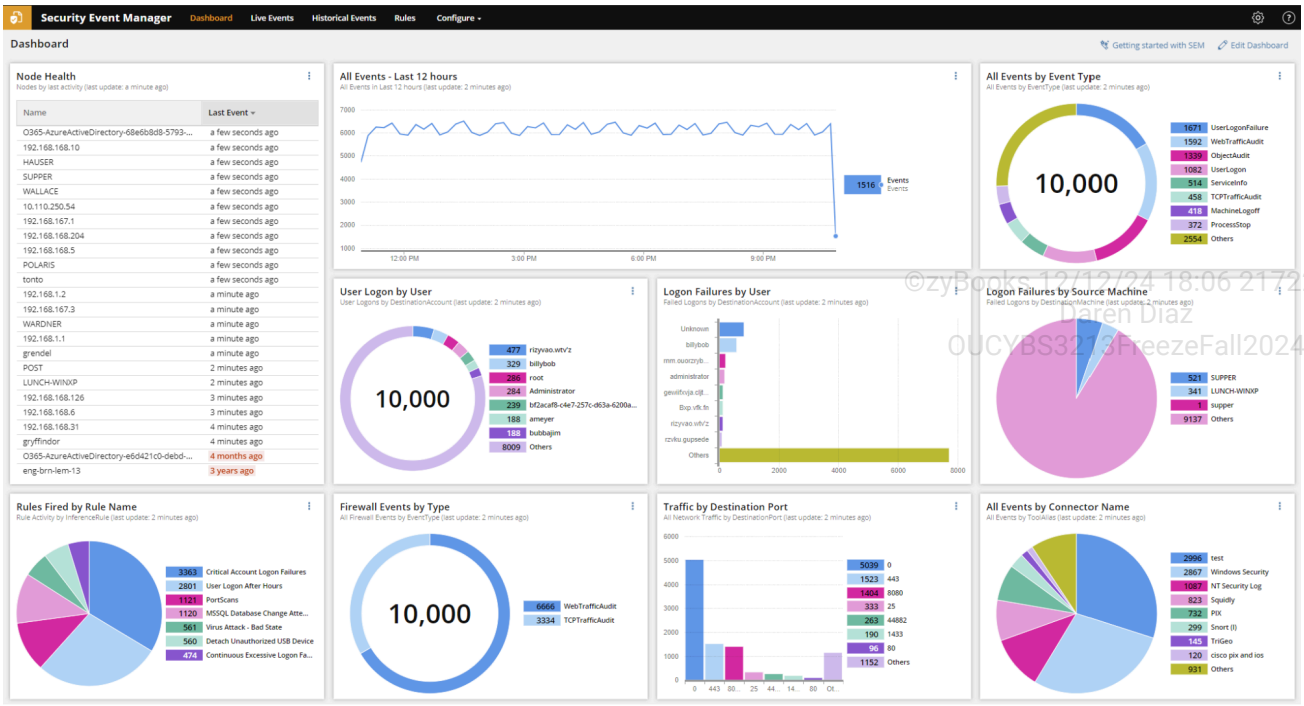
3) DLP does not prevent a company employee from sending an email that contains what type of information?
   - order history of a company's customer
   - publicly available information on the company's finances
   - credit card numbers belonging to the company's customers

Host-based systems

An endpoint, such as a network host, can be protected against attacks by host-based firewalls and intrusion detection and prevention systems. A ***host-based firewall*** is a software firewall that runs on a host and controls the host's inbound and outbound network traffic. A host-based firewall can restrict inbound and outbound traffic to a specific application running on a host and can detect and prevent malware from being installed on a host. Most modern operating systems include a host-based firewall. Ex: Windows Defender Firewall is a host-based firewall.

A ***host-based intrusion detection system (HIDS)***, or ***host-based IDS***, is a software program that runs on a host and can detect a threat to the host. A HIDS monitors and analyzes a host's running processes, network traffic, and log files to detect a threat to the host. Ex: A security information management (SIM) system running on a host is a HIDS. A ***host-based intrusion prevention system (HIPS)***, or ***host-based IPS***, is a software program that runs on a host and can detect and prevent a threat to the host. The main difference between a HIDS and a HIPS is that a HIDS can detect but not prevent a threat to a host.

Example 9.2.2: The Solarwinds Security Event Manager (SEM) monitors and detects threats to network hosts.



Credit: Solarwinds Inc.[2]

PARTICIPATION ACTIVITY 9.2.2: Host-based systems.

How to use this tool

Host-based firewall | **Host-based IPS (HIPS)** | **Host-based IDS (HIDS)**

	A software program that runs on a host and can detect a threat to the host.
	A software program that runs on a host and can detect and prevent a threat to the host.
	A software firewall that runs on a host and controls the host's inbound and outbound network traffic.

Reset

PARTICIPATION ACTIVITY 9.2.3: Host-based systems.

1) A host-based firewall can protect an endpoint against attacks.
   - ○ True
   - ○ False

2) A host-based firewall cannot control inbound and outbound traffic to a specific application running on a host.
   - ○ True
   - ○ False

3) A host-based intrusion detection system (HIDS) can prevent a threat to a host.
   - ○ True
   - ○ False

(*1) Broadcom Inc. "Symantec Data Loss Prevention (DLP)". https://www.broadcom.com/products/cyber-security/information-protection/data-loss-prevention.

(*2) Solarwinds Inc. "Solarwinds Security Event Manager (SEM)". http://www.solarwinds.com/security-event-manager.

9.3 Endpoint hardening: Disk, registry, ports, and services

Data encryption

An endpoint stores data locally on a partitioned disk consisting of one or more volumes. Endpoint hardening includes data encryption measures to provide data confidentiality and prevent unauthorized data access. Data encryption can be applied to an individual file or an entire partition, volume, or disk. Ex: Linux Unified Key Setup (LUKS) is used to encrypt Linux volumes.

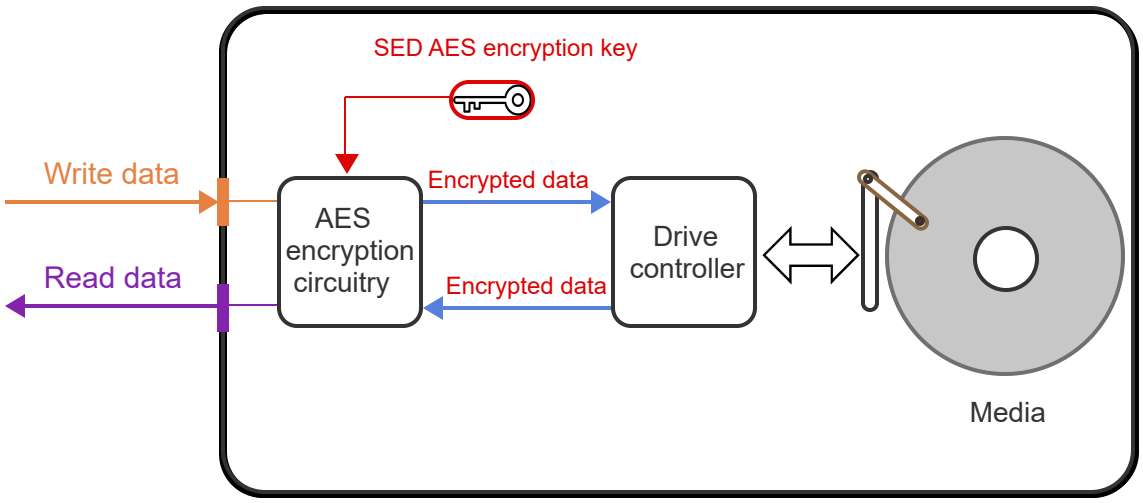
Two disk encryption methods exist:

- ***Full disk encryption (FDE)*** is a method to secure drive data using encryption. FDE encryption can be implemented in hardware or software. FDE encryption key is generated with an

authentication token based on a user password or PIN. Ex: Microsoft Windows BitLocker is a security feature that uses software or hardware-based encryption to protect drive data. BitLocker stores the drive encryption key in a trusted platform module (TPM) or on an external storage device, such as a USB drive. BitLocker uses advanced encryption standard (AES) in cipher block chaining (CBC) mode with 128 or 256-bit keys.

- ***Self-encrypting drive (SED)*** is a hard disk drive (HDD) or solid state drive (SSD) with encryption circuitry built into the drive. SED is a special form of FDE that only uses hardware-based encryption. SED is transparent to operating systems, applications, and users, and is designed to automatically encrypt and decrypt drive data without the need for disk encryption software. SED uses advanced encryption standard (AES) with 128 or 256-bit keys. SED encryption key is factory-set and stored on the drive controller.

PARTICIPATION ACTIVITY 9.3.1: Self-encrypting drive (SED).



Self-encrypting drive (SED)

Animation content:

Static image: A large box labeled "Self-encrypting drive (SED)". Within the SED is a box labeled "AES encryption circuitry", a box labeled "Drive controller", and a disk labeled "Media". An arrow labeled "Write data" outside of the SED box points to the left edge of the SED and to the AES encryption circuitry box. Inside the SED, a red key labeled "SED AES encryption key" has an arrow pointing to the AES encryption circuitry box. An arrow labeled "Encrypted data" points from the AES encryption circuitry box to the Drive controller box. A two-way arrow is between the Drive controller and the Media disk. An arrow labeled "Encrypted data" points from the Drive controller box to the AES encryption circuitry box. An arrow labeled "Read data" points from the AES encryption circuitry box to the left and outside the SED.

Animation captions:

1. When data is written to a SED, the data is encrypted with the SED AES key before the data is written to the disk
2. When data is read from a SED, the data is decrypted with the SED AES key. The encryption/decryption is transparent to OS, applications, and users.

Opal is a set of security specifications for SED which defines a hierarchy of security management standards to secure data from theft and tampering by unauthorized persons. Opal specification is published by the Trusted Computing Group (TCG) Storage Workgroup. Opal is used for applying hardware-based encryption to storage devices, such as hard-disk drives (HDD), solid-state drives (SSD), and optical drives. Ex: An Opal-compliant SED requires user authentication to unlock upon bootup.

PARTICIPATION ACTIVITY 9.3.2: Disk encryption.

How to use this tool

Endpoint hardening | **Self-encrypting drive (SED)** | **Full disk encryption (FDE)** | **Opal**

	The process of improving the security of an endpoint by reducing the endpoint's attack surface.
	A set of security specifications for SED which defines a hierarchy of security management standards to secure data from theft and tampering by unauthorized persons.

	A method to secure data residing on a drive by encryption.
	A hard disk drive (HDD) or solid state drive (SSD) with encryption circuitry built into the drive.

Reset

PARTICIPATION ACTIVITY 9.3.3: Disk encryption.

Select the disk encryption method that supports the stated feature.

1) Uses an encryption key based on a password or PIN
   - FDE
   - SED
   - FDE and SED

2) Supports software-based encryption
   - FDE
   - SED
   - FDE and SED

3) Uses AES encryption
   - FDE
   - SED
   - FDE and SED

4) Complies with Opal security specifications
   - FDE
   - SED
   - FDE and SED

5) Includes encryption hardware
   - FDE
   - SED
   - FDE and SED

6) Has an embedded AES encryption key

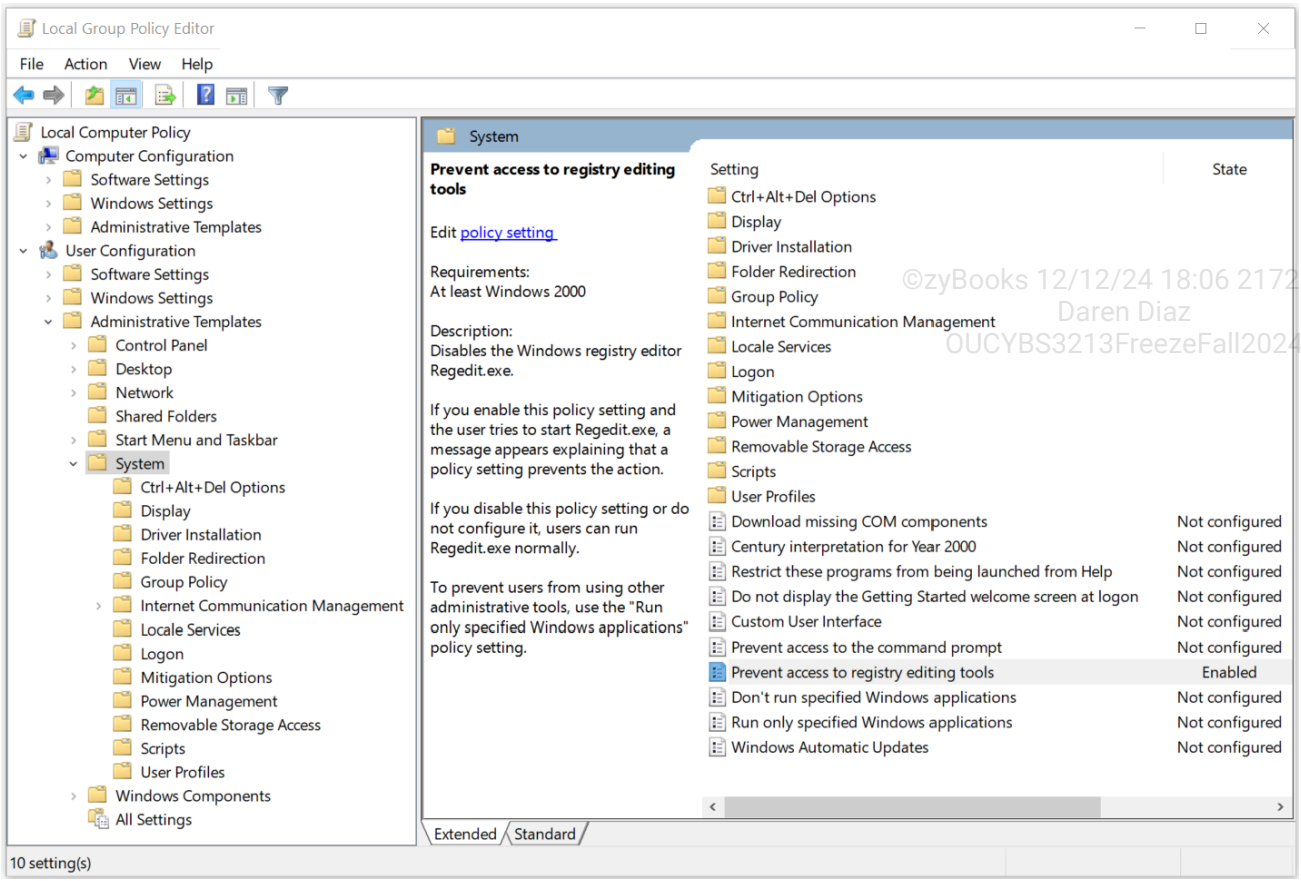
- FDE
- SED
- FDE and SED

Registry

Registry is a hierarchical database that stores configuration settings and options in Microsoft Windows operating systems. The kernel, device drivers, services, security accounts manager (SAM), and third-party programs can use the registry to store configuration settings. The registry stores information such as user passwords and file permissions, the types of documents an application can create, and the list of recently opened files.

In a Windows endpoint, the registry should be secured to protect the endpoint, operating system, and applications. Unauthorized users should be prevented from modifying registry entries by restricting access to registry editing tools such as `regedit`. User access to registry editing tools is controlled by a Group Policy setting located under the *"User Configuration\Administrative Templates\System\Prevent access to Registry editing tools"* setting in the default Group Policy Object (GPO).

Example 9.3.1: Windows Group Policy setting to prevent access to registry editing tools.



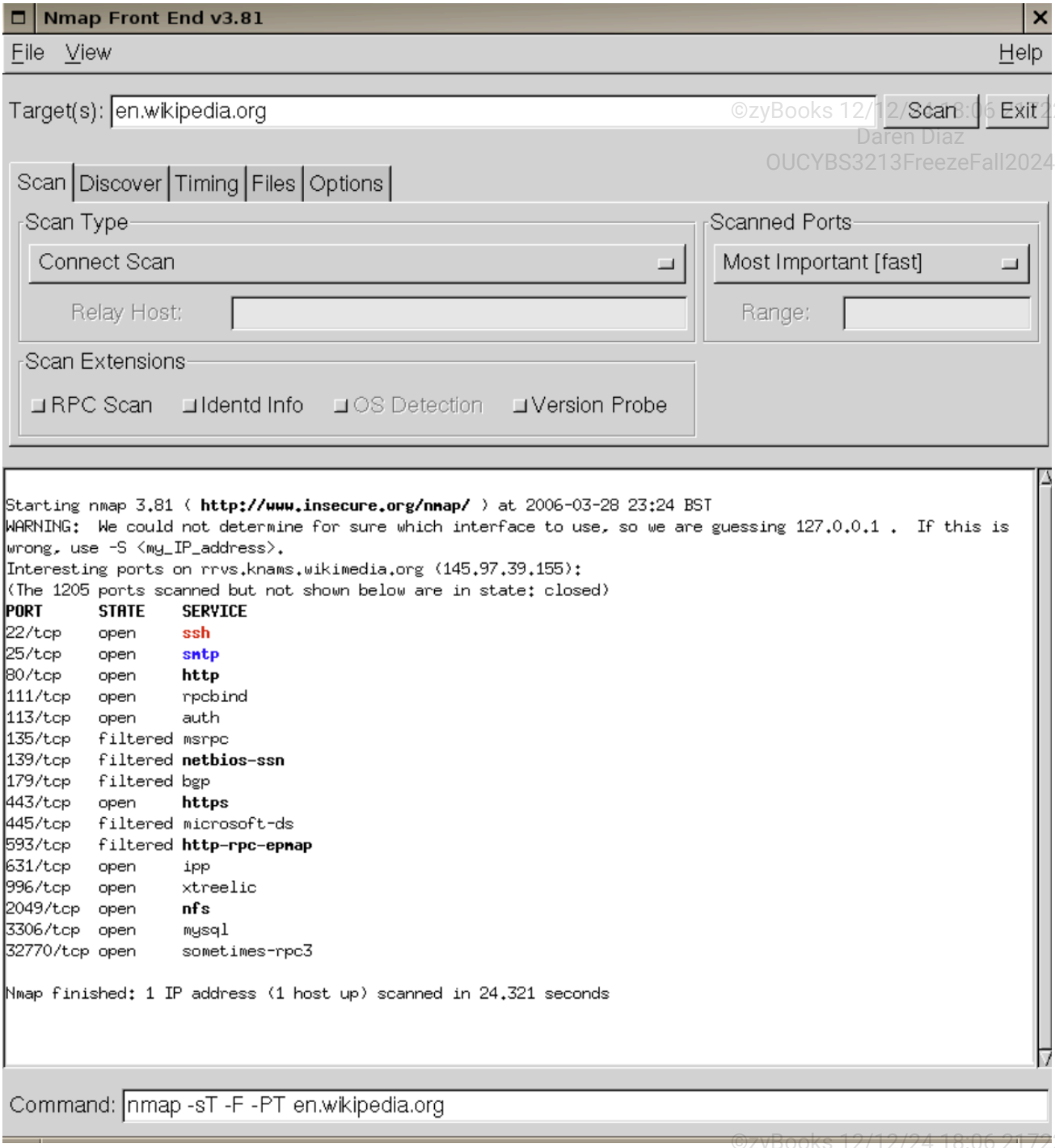
Open ports and services

An ***operating system port*** is a 16-bit unsigned number that uniquely identifies a network application or service on a host. A port number is between 0 and 65,535. A port number is of three types:

- A ***well-known port*** is a port number between 0 and 1,023 that is reserved for commonly used services. Ex: TCP port 80 is reserved for Hypertext Transfer Protocol (HTTP).
- A ***registered port*** is a port number between 1,024 and 49,151 that is registered with Internet Assigned Number Authority (IANA) for a specific use. Ex: TCP port 3,389 is registered with IANA for Windows Remote Desktop Protocol (RDP).
- A ***higher-number port***, also known as a ***private port*** or ***dynamic port***, is a port number between 49,152 and 65,535 that can be used by any application.

An IP address has a TCP and a UDP port. An ***open port*** is a TCP or UDP port that is configured to accept data packets. A ***closed port*** is a TCP or UDP port that is configured to reject or ignore data packets. A network scanner, such as Nmap, can be used for port enumeration. ***Port enumeration*** is the process of identifying a host's open ports and determining the services which use those ports.

Example 9.3.2: Nmap output showing open ports and the services using those ports.



A network service uses a port to send and receive data. Open ports expose potential vulnerabilities of services that use those ports for communication. To reduce an endpoint's attack surface, the endpoint's unused ports should be closed or filtered by a firewall, and unnecessary services should be disabled.

PARTICIPATION ACTIVITY 9.3.4: Open ports and services.

How to use this tool

Well-known port **Port enumeration** **Registered port** **Dynamic port**

Operating system port

	A 16-bit unsigned number that identifies a host and a network service.
	A port number between 0 and 1,023 that is reserved for commonly used services.
	A port number between 1,024 and 49,151 that is registered with Internet Assigned Number Authority (IANA) for a specific use.
	The process of identifying a host's open ports and determining the services which use those ports.
	A port number between 49,152 and 65,535 that can be used by any application.

Reset

PARTICIPATION ACTIVITY 9.3.5: Open ports and services.

1) What type of port is TCP port 1,194?

- ○ well-known
- ○ registered
- ○ dynamic

2) What type of port is UDP port 52,312?

- well-known
- registered
- dynamic

3) What type of port is TCP port 443?

- well-known
- registered
- dynamic

4) An endpoint runs an unused FTP server. Why should the FTP server be disabled?

- To make TCP port 21 available to the endpoint's other services
- To protect the endpoint from attacks that may exploit the vulnerabilities of the FTP server
- To increase the endpoint's attack surface

5) What is the impact of closing an endpoint's unused ports on the endpoint's attack surface?

- Increases the endpoint's attack surface
- Decreases the endpoint's attack surface
- Has no impact on the endpoint's attack surface

Default passwords and unnecessary software

Updating default passwords and removing unnecessary software are critical steps in enhancing computer system security and mitigating risks. Devices and software are often equipped with default passwords that are widely known or easily searchable, posing significant security risks. Updating default passwords to strong, unique alternatives, can prevent access by unauthorized entities.

Uninstalling software that is no longer necessary is equally important for reducing security vulnerabilities and improving system performance. Such software not only

increases the attack surface of systems, but also consumes valuable resources that could affect system functionality. Conducting regular audits to identify and remove unused programs reduces the risk of security threats and optimizes functionality, creating a more secure and efficient computing environment.

CHALLENGE ACTIVITY 9.3.1: Endpoint hardening.

581480.4344582.qx3zqy7

Start

Select all attributes of SED.

- [ ] Requires a user password or PIN
- [ ] Is invisible to the operating system
- [ ] Is hardware-based only
- [ ] Can be Opal-compliant by adhering to Opal standards

1 2 3

Check Next

9.4 Endpoint hardening: Software updates, patch management, and operating system vulnerabilities

Software updates

Endpoint software is updated to remove security vulnerabilities, fix errors, add features, and improve performance and functionality. Three software update types exist:

- A ***software patch***, or ***patch***, is a set of code changes to a program that update, improve, or fix the program. ***Patching*** is the process of applying software patches to ensure a program's continuous improvement in functionality, performance, security, or usability. Ex: A software patch may add new features to a word processor program.
- A ***hotfix***, also known as a ***quick-fix engineering update*** or ***QFE update***, is a software update to address a specific problem in a program. A hotfix is developed and released quickly to fix a recently found problem. A hotfix is typically a temporary measure until a new full release of the software becomes available. Ex: A hotfix may resolve a buffer overflow in a program that may create exploitable vulnerabilities.
- A ***service pack*** is a collection of software patches and hotfixes combined into a single package. A service pack aims to bring a system up-to-date through a single update process. A service pack installation is simpler and less error-prone than the installation of a large number of software patches and hotfixes individually. Ex: A Windows service pack includes all the hotfixes and software patches released since the last service pack.

PARTICIPATION ACTIVITY | 9.4.1: Operating system hardening.

How to use this tool

Service pack **Hotfix** **Software patch**

	A piece of code that removes or fixes a vulnerability, or improves or adds a feature to an application.
	A minor software update to address a specific problem.

	A collection of software patches and hotfixes that are combined into a single package.

Reset

PARTICIPATION ACTIVITY 9.4.2: Operating system hardening.

Select the software update type in each scenario.

1) A software update that fixes a recently found security vulnerability.
   - patch
   - hotfix
   - service pack

2) A software update that fixes an application vulnerability that enables SQL injection attacks.
   - patch
   - hotfix
   - service pack

3) A software update that contains a large number of fixes for an operating system and applications.
   - patch
   - hotfix
   - service pack

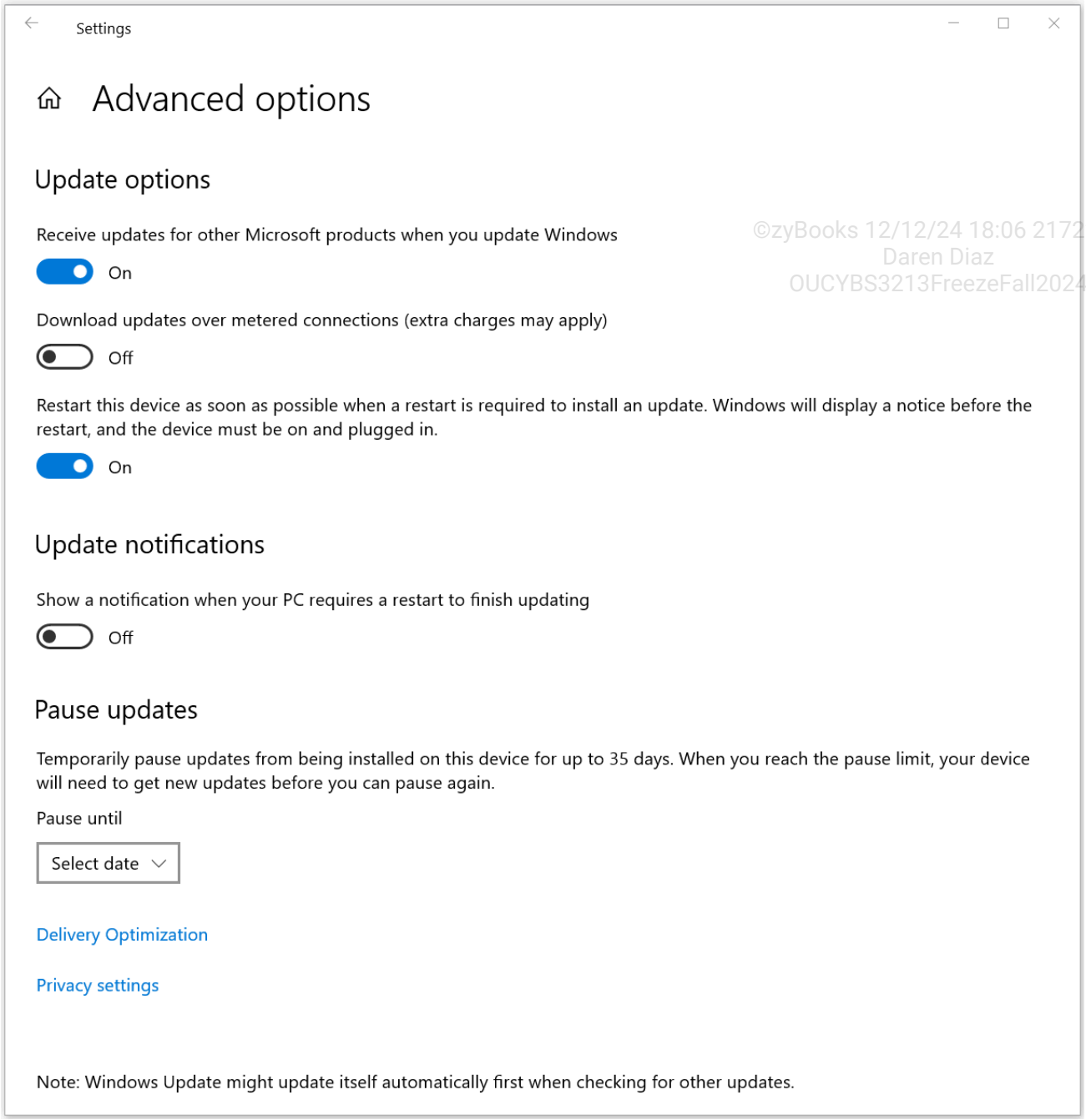
Patch management

Patch management is the process of distributing and applying software updates, which includes monitoring patch availability to ensure timely application of critical fixes. Patch management can be automated to expedite the deployment of security patches and reduce the chance of running outdated software. However, the inability to patch certain systems can pose risks, particularly when software vendors discontinue support or when systems cannot be temporarily shut down for updates. ***Auto-update*** is a patch management option that enables the automatic download and installation of software patches, hotfixes, and service packs without user intervention.

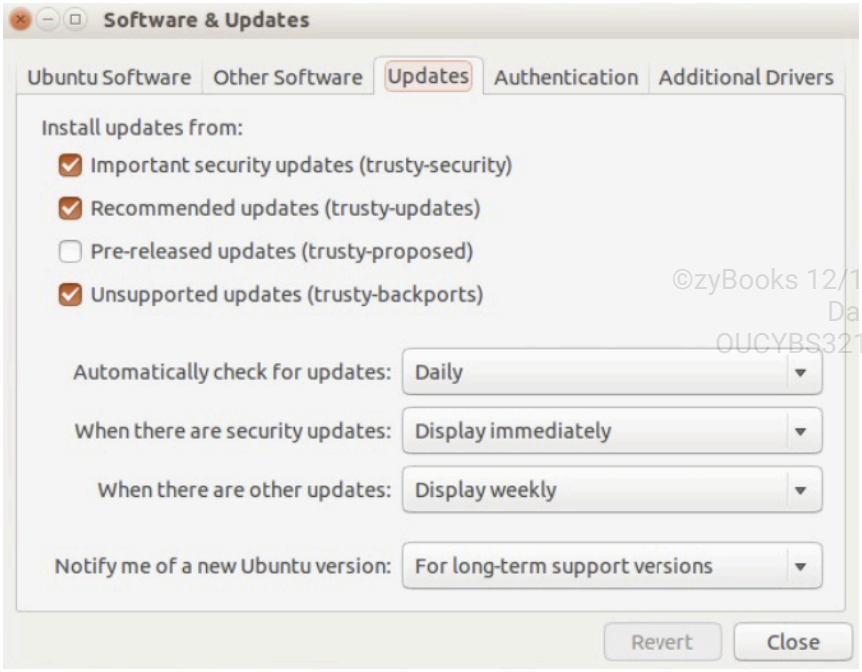
A ***third-party software*** is any software that is not included with an operating system. Ex: Bitdefender Antivirus is a third-party software to Microsoft Windows since the Bitdefender Antivirus software is not included with Windows. Similar to operating system software, third-party software should be regularly updated. ***Third-party updates*** refer to software patches that are distributed by third-party software manufacturers.

Patch management in modern operating systems, such as Microsoft Windows and Linux, scans for and detects missing patches and downloads and installs patches, hotfixes, and operating system updates.

Example 9.4.1: Advanced Windows Update options.



Example 9.4.2: Ubuntu Linux Software Updater.



PARTICIPATION ACTIVITY 9.4.3: Patch management.

How to use this tool

Third-party updates **Patch management** **Auto-update**

	The process of distributing and applying software updates.
	A patch management option that enables the automatic download and installation of software patches without user intervention.
	Software patches that are distributed by third-party software manufacturers.

Reset

PARTICIPATION ACTIVITY 9.4.4: Patch management.

1) Software updates to Microsoft Windows Defender Firewall are considered third-party updates.

- True
- False

2) Auto-update is a patch management option that requires user intervention.

- True
- False

3) Patch management in modern operating systems downloads and deploys patches, hotfixes, and operating system updates.

- True
- False

Operating system vulnerabilities

Operating system (OS) vulnerabilities encompass a range of security flaws inherent to the OS software that can be exploited to undermine system integrity, confidentiality, and availability. OS vulnerabilities include buffer overflows, privilege escalations, unpatched security flaws, and configuration errors, often triggered by unauthorized data input, improper access controls, or exploitation of outdated system components. Ex: A buffer overflow occurs when a buffer is sent more data than the buffer can hold, potentially executing malicious code or causing a system crash.

Due to the OS's critical role in managing hardware and software resources, the exploitation of OS vulnerabilities can undermine the entire system's security, potentially leading to unauthorized data access, system corruption, or a complete system compromise. A defense-in-depth approach that includes regular updates, patch management, robust access controls, firewalls, and intrusion detection systems provides comprehensive protection against OS vulnerabilities.

Table 9.4.1: Operating system-based vulnerabilities.

Vulnerability category	Impact	Example
System	Privilege escalation, arbitrary code execution, system compromise	Exploit Windows API flaws, Linux system call vulnerabilities

Memory	Arbitrary code execution, unauthorized access, memory corruption	Stack and heap overflows, exploits in file system access control lists
Network	Network compromise, unauthorized access, protocol manipulation	Exploits in Windows TCP/IP stack, vulnerabilities in Linux network protocols
Configuration	Unauthorized access, privilege escalation	Improper Windows registry settings, misconfigured Linux permissions
Authentication	Privilege escalation, unauthorized access, credential exposure	Credential theft in Windows, authentication bypass in Linux

PARTICIPATION ACTIVITY 9.4.5: Operating system vulnerabilities.

1) What is a common consequence of buffer overflow vulnerabilities in an operating system?
   - Arbitrary code execution
   - Improved user access controls
   - Enhanced system stability

2) What could be a consequence of misconfigured Linux permissions?
   - Unauthorized access
   - Faster file system access
   - Improved hardware interoperability

3) What is a potential impact of exploiting network vulnerabilities in an operating system?
   - Network compromise
   - Increased network coverage
   - Faster internet connections

4) What is the impact of a defense-in-depth approach to security on the management of OS vulnerabilities?

- Focuses solely on external threats
- Reduces the efficiency of the operating system
- Layers multiple security measures to protect at different levels

5) Which mitigation technique is most effective against privilege escalation due to system vulnerabilities?

- Installing antivirus software
- Regular software updates and patches
- Monitoring network traffic

Malicious software updates

*A **malicious software update** is a fraudulent software update that appears legitimate but contains harmful code designed to exploit trust in the update process. Such updates may be distributed through various means, including phishing emails, infected downloads, compromised update servers, or exploited vulnerabilities. Once installed, the updated code may perform actions such as creating backdoors, stealing sensitive data, disabling security features, or enabling remote control of the affected system.*

9.5 Boot integrity

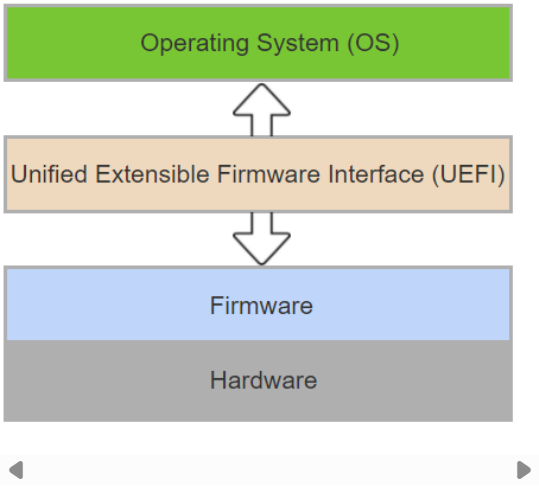
Unified extensible firmware interface (UEFI)

An endpoint should be securely booted to prevent the installation of malware on the endpoint before an operating system is loaded on the endpoint. Unified extensible firmware interface (UEFI) enables a

secure boot process. ***Unified extensible firmware interface (UEFI)*** is a specification that defines a software interface between an operating system and platform firmware. The interface consists of data tables containing platform-related information, and boot and runtime service calls available to the boot loader and the operating system. A ***boot loader***, or ***bootstrap loader***, is a program that loads an operating system into the computer memory.

UEFI replaces the legacy Basic Input/Output System (BIOS) firmware interface. ***BIOS*** is a program stored in non-volatile memory that performs start-up procedures when a computer is powered on.

Figure 9.5.1: Unified Extensible Firmware Interface (UEFI).



PARTICIPATION ACTIVITY 9.5.1: Unified extensible firmware interface (UEFI).

How to use this tool

UEFI **BIOS** **Boot loader**

	A specification that defines a software interface between an operating system and platform firmware.
	A program that loads an operating system into the computer memory.
	A program stored in non-volatile memory that performs start-up procedures when a computer is powered on.

Reset

PARTICIPATION ACTIVITY 9.5.2: Unified extensible firmware interface (UEFI).

1) UEFI consists of data tables containing platform-related information and boot and runtime service calls.
   - ○ True
   - ○ False

2) BIOS is a program stored in volatile memory.
   - ○ True
   - ○ False

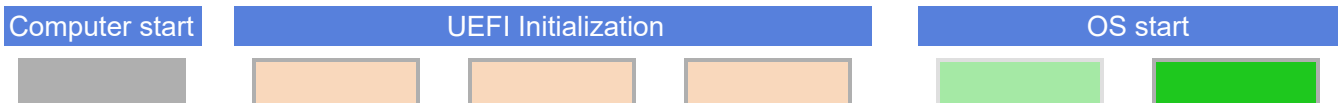
3) A boot loader loads the firmware into memory.
   - ○ True
   - ○ False

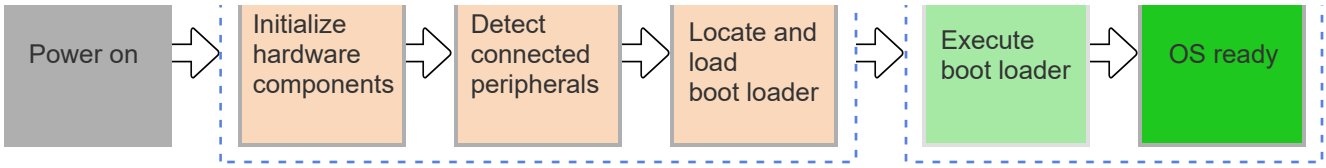
Unified extensible firmware interface (UEFI) functions

UEFI improves BIOS by providing security features, faster boot times, and supporting graphical user interfaces and mice for pre-boot configuration. UEFI's networking features enable remote configuration, diagnostics, and troubleshooting of computers.

UEFI initializes system hardware, detects connected peripherals, and locates and executes a boot loader to load an operating system into memory. UEFI can be loaded from a hard disk or solid-state drive, non-volatile memory, or a network share.

PARTICIPATION ACTIVITY 9.5.3: UEFI initialization.





Animation content:

Static image: A flowchart showing UEFI initialization. The first box labeled "Power on" is below the title "Computer start". The title "UEFI Initialization" is above the next three steps: "Initialize hardware components", "Detect connected peripherals", and "Locate and load boot loader". The title "OS start" is above the final two steps: "Execute boot loader" and "OS ready".

Animation captions:

1. When a computer is powered on, UEFI initializes the hardware components, including the processor, chipset, and motherboard.
2. Next, UEFI detects connected peripherals, including mouse, keyboard, and drives.
3. UEFI then cycles through all storage devices to locate and load a boot loader.
4. The boot loader is executed to load the operating system into memory.
5. After the operating system is ready, user processes can be executed.

PARTICIPATION ACTIVITY 9.5.4: Boot integrity.

1) UEFI defines an interface between an operating system and _____.
   - application programs
   - firmware
   - device drivers

2) UEFI initialization begins with _____.
   - detecting peripherals
   - loading the boot loader
   - initializing hardware components

3) UEFI cannot be loaded from _____.
   - solid-state drive (SSD)

- ○ non-volatile memory
- ○ a website

4) UEFI initialization ends after UEFI loads a _____.

- ○ device driver
- ○ operating system
- ○ boot loader

Boot integrity

UEFI provides security features that support boot integrity. Boot integrity ensures that only verified software is used in a boot process. Two boot integrity methods exist:

- ***Secure boot*** is a boot method that ensures only programs trusted by the system manufacturer are loaded and executed during the boot process. A system that supports secure boot maintains a signature database of valid programs and only executes a boot program that has a valid signature. The signature database is updated by the system manufacturer. Secure boot is supported by Windows Server, Windows 10, and various Linux distributions, including Ubuntu and Fedora.
- ***Measured boot*** is a boot method in which the hash of each boot program is computed (measured) and stored in a trusted platform module (TPM). ***Boot attestation*** is the process of sending the measured boot logs stored in a TPM to a remote administrative server for evaluation. An administrative server compares each software hash in the boot chain with the hash of a valid boot program. If a hash does not match, the administrative server may restrict the system's access to resources or take remedial actions to secure the system.

Secure boot and measured boot can be used together in the same system to improve the security of the boot process.

PARTICIPATION ACTIVITY 9.5.5: Boot integrity.

How to use this tool

Boot attestation **Secure boot** **Measured boot**

A boot method that ensures only firmware and software trusted by the

	system manufacturer is used in a boot process.
	A boot method in which the hash of each firmware and software in a boot process is computed and stored.
	The process of sending measured boot logs to a remote server for evaluation.

Reset

PARTICIPATION ACTIVITY 9.5.6: Boot integrity.

Select the boot integrity method that performs the stated task.

1) Stores the hashes of each boot program in a TPM.
   - ○ Secure boot
   - ○ Measured boot

2) Checks the validity of boot programs by using a signature database.
   - ○ Secure boot
   - ○ Measured boot

3) Sends boot logs to an administrative server for boot attestation.
   - ○ Secure boot
   - ○ Measured boot

4) Ensures that malware is not installed on a computer before an operating system is loaded.
   - ○ Secure boot
   - ○ Measured boot

Root of trust

The ***root of trust (RoT)*** is a system's set of highly reliable hardware, firmware, and software components that perform critical security functions. A ***hardware root of trust (HRoT)*** is a root of trust that is implemented in hardware only. An HRoT is based on a hardware-validated boot process that ensures a system is started using immutable (unchangeable) code that is not affected by malware.

A root of trust anchors a chain of trust that is the foundation of all secure operations of a system. In a chain of trust, the signature of each piece of firmware is validated before the firmware is executed. Upon the execution of all validated firmware, system control is handed over to UEFI's secure boot process.

PARTICIPATION ACTIVITY 9.5.7: Root of trust.

1) A hardware root of trust is implemented in hardware to ensure start up code is modifiable.
   - ○ True
   - ○ False

2) A root of trust is the basis for UEFI's secure boot process.
   - ○ True
   - ○ False

3) A chain of trust is broken when the signature of a piece of firmware cannot be validated.
   - ○ True
   - ○ False

Hardware vulnerabilities

Hardware vulnerabilities, including those present in firmware, pose significant risks to endpoint security. Outdated firmware can contain known vulnerabilities that can be exploited to gain unauthorized access or compromise the integrity of the system. Additionally, end-of-life hardware, no longer receiving security updates or patches, becomes increasingly susceptible to exploitation over time.

Legacy hardware, still in use for compatibility or cost reasons, may lack modern security features and thus be more vulnerable to attacks. Hardware lifecycle management practices, including regular firmware updates, timely replacement of end-

of-life hardware, and evaluation of security implications when using legacy hardware can minimize the associated risks.

CHALLENGE ACTIVITY | 9.5.1: Boot integrity.

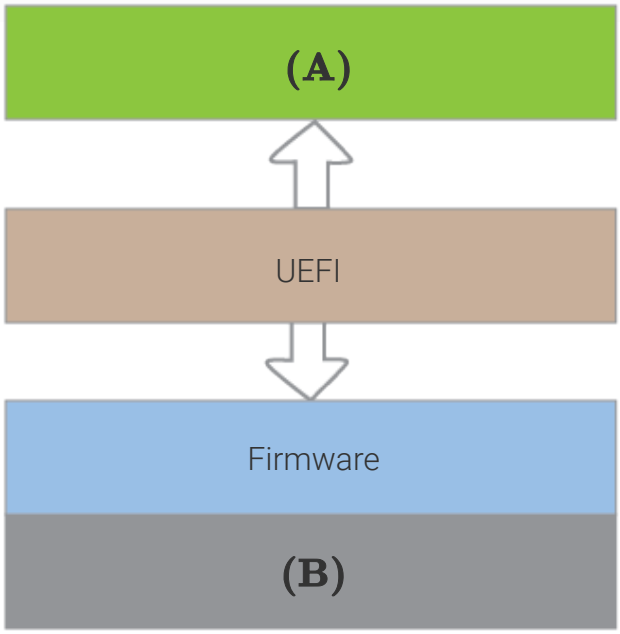
581480.4344582.qx3zqy7

Start

Select the terms for the missing labels.

(A) Pick

(B) Pick



1 2 3

Check Next

9.6 LAB: Host security (Walkthrough)