Chapter 5

Securing Individual Systems

Episode 5.01

Episode title: **Malware**

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

Malware

- Software that is detrimental to the operation of a host

Virus

- Program that can replicate only through definite user interaction
- Activates once a user clicks or downloads
- Fileless malware/virus
  - No file, lives only in memory
  - Difficult for anti-malware to detect

Ransomware

- Cryptomalware/ crypto-ransomware
  - Uses encryption to lock a user out of a system
- Attacker hides your data until you pay a ransom

Worm

- Virus that, once started, replicates itself
- More like a pathway for replication

Trojan Horse

- A program that looks benign, but in fact hides a nefarious program inside it
- No replication by themselves
- Remote access Trojan (RAT)
  - Maliciously takes control of a system remotely

Backdoor

- Not necessarily nefarious
- Created by developers as easy maintenance entry point
- Can be exploited by attackers if left open by developers
- Can be created in a program by hackers to gain access

Potentially Unwanted Programs (PUPs)

- Software that may have negative or undesirable effects
- Crapware, adware, spyware, bloatware

Bots/Botnets

- Distributed attack using remotely-controlled malware controlling several computers
  - Often running some kind of RAT
- Hosts are called bots or zombies
- One kind of botnet attack is a distributed denial of service (DDoS) attack
  - Overload of traffic from a number of sources that makes resources unavailable for legitimate users

Bots/Botnets

- Command and control (C2)
  - Protocols that automate the control, not requiring human interaction after the initial programming

Keylogger

- Hardware
  - Device that plugs in between keyboard and computer to log keystrokes
  - Many have WAPs built in for remote access
- Software
  - Program that logs keystrokes
  - Example: monitoring kids' activity on the computer
  - Most anti-malware can find nefarious software keyloggers

Rootkit

- Can often be somewhat invisible
- Goal is to get root access to a system
- Usually installed on the boot of the systems they're attacking

Logic Bomb

- Often a script set to execute
- Created with a timer to go off at a specific time or during a specific event on a system

Quick Review

- Viruses activate and replicate only after specific user interaction
- Ransomware is when an attacker hides data from a user and demands a ransom to return it; cryptomalware is when the attacker uses encryption to hide the data
- A worm is a virus that replicates itself
- Trojans are bad software hidden inside seemingly good software
- Remote access Trojans (RATs) gain control remotely
- Backdoors lead to access to programs through non-traditional ways and can be accidentally left open by developers and exploited by attackers or purposefully created by hackers

Quick Review

- Potentially unwanted programs (PUPs) add useless or potentially negative programs without the user's consent
- Bots and botnets use automated command and control (C2) to infect hosts (zombies) and cause issues such as distributed denial of service (DDoS) attacks
- Keyloggers record keystrokes on a computer and can be hardware or software
- Rootkits hide malware at the boot level and attempt to infect critical operating system files
- Logic bombs are often scripts set to execute based on a time or event trigger

Episode 5.02

Episode title: **Weak Configurations**

Objective: **1.6 Explain the security concerns associated with various types of vulnerabilities.**

Weak Configurations

- On-premises vs. cloud solutions
- Open permissions
  - Open wireless networks
  - Guest user accounts
  - No intruder lockout settings
  - Too many file or app permissions

Weak Configuration Example

- Linux root account
  - Don’t sign in with root account
  - Use sudo to run privileged commands
  - Disallow remote access as root
  - Use su to temporarily switch to root

Insecure Cryptographic Solutions

- Wi-Fi Wired Equivalent Privacy (WEP)
  - Use WPA2 or WPA3
- Digital Encryption Standard (DES)
  - Use AES
- Secure Sockets Layer (SSL)
  - Use TLS
- Transport Layer Security (TLS)
  - Not secure
    - Versions 1.0 and 1.1
  - Secure
    - Versions 1.2 and 1.3

Change Default Settings

- IP address
- Open port numbers
- Web server root filesystem location
  - Directory traversal attacks
- Username/password policies

Quick Review

- Never allow default usernames and passwords to be used
- Modify default configurations to harden your environment
- Only use powerful accounts when performing administrative tasks
- Do not use SSL or TLS version 1.0/1.1
- Change default port numbers

Episode 5.03

Episode title: **Common Attacks**

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

Zero-Day (0-Day) Attacks

- An exploit unknown by the vendor and the public
- Zero Day Initiative (ZDI)
  - Encourages the private reporting of vulnerabilities to vendors

Common Attacks

- DNS sinkholing
  - Return false DNS query results
- Privilege escalation
  - Attacker acquires a higher level of access
  - Example: compromising an admin account that has a weak password
- Replay attack
  - Attacker intercepts and later retransmits or uses sensitive data

Common Attacks

- Pointer/object dereference
  - Attacker manipulates memory pointers to point to unexpected memory locations
  - Normally causes software to crash (DoS attack)
- Error handling
  - Improper handling can crash a system
  - Disclosure of too much information

Common Attacks

- Dynamic Link Library (DLL) injection
  - Attacker places malicious DLL in the file system
  - Legitimate running processes call malicious code within the DLL
- Resource exhaustion
  - DoS or DDoS
  - Memory leaks

Common Attacks

- Race conditions
  - Code runtime phenomenon
  - Action that might occur before security control is in effect
  - Based on timing

Quick Review

- Zero-day attacks are not patchable and are unknown to the vendor and public
- Hardening user accounts mitigates privilege escalation attacks
- Common attacks resulting from software development flaws include race conditions, resource exhaustion, and pointer/object dereferencing

Episode 5.04

Episode title: **Driver and Overflow Attacks**

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

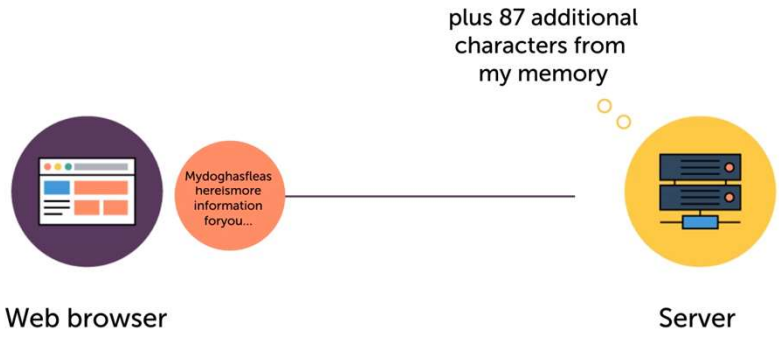
Driver Attacks

- Driver shimming
  - Normally used to allow legacy software to run
  - Can be installed by a malicious user
    - Device access
    - Injected in software development stage
  - Intercepts application programming interface (API) calls
- Driver refactoring
  - Restructures internal code while maintaining external behavior
  - Can evade signature-based antivirus

Overflow Attacks

- Integer overflow
  - Less memory than expected is allocated
  - Leads to
    - Sensitive information disclosure
    - Remote exploit privilege escalation
    - Application crash
- Buffer overflow
  - Too much data is read or written compared to allocated memory

Heartbleed Bug



Quick Review

- Driver shimming attacks occur when malicious driver shims are installed
- Driver refactoring modifies source code to evade detection while functionality remains unchanged
- Integer overflows are related to an inadequate amount of memory being allocated
- Buffer overflows are a result of integer overflows

Episode 5.05

Episode title: **Password Attacks**

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

Password Attacks

- Online vs. offline
- Tools
  - John the Ripper
  - Cain and Abel
  - Hydra
- Dictionary
  - Uses common username/ password files
  - Tries thousands or millions of likely possibilities to login to a user account

Password Attacks

- Brute-force
  - Try every possible combination of characters
- Multiple attempts should trigger an account lockout

Password Spraying

- Blast many accounts with a best-guess common password before trying a new password
- Slower (per-user account basis) than traditional attacks
- Less likely to trigger account lockout thresholds

Quick Review

- Dictionary attacks uses username/password lists
- Brute-force attacks try character combinations
- Password spraying tries a single password against many accounts before moving on to a different password
- MFA and strong password policies mitigate common password attacks

Episode 5.06

Episode title: **Bots and Botnets**

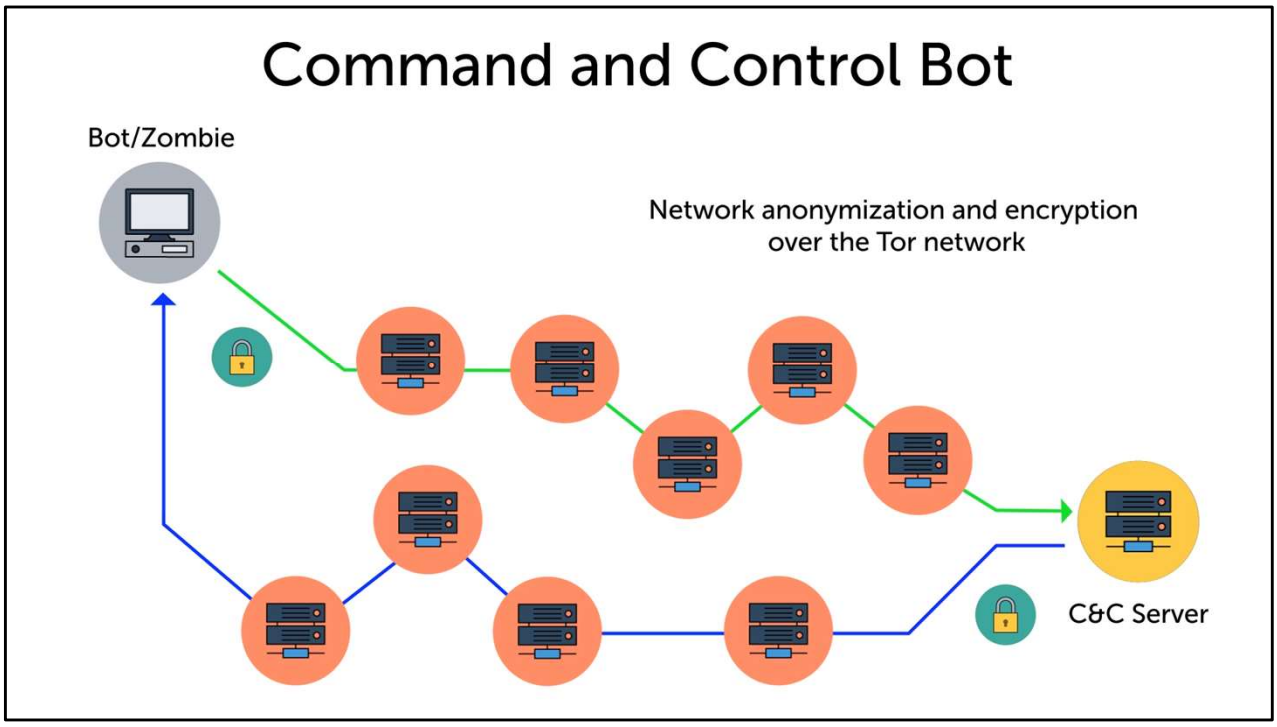
Objective: **1.2 Given a scenario, analyze potential indicators to determine the type of attack.**

Bots and Botnets

- Bot
  - Single infected device under attacker control
  - AKA "zombie"
- Botnet
  - Collection of infected devices under attacker control

Bots

- Periodically talks to command and control (C2/C&C) attacker server
  - Mitigate with IDS
  - Attacker might have directions stored in a DNS TXT record
    - Network IDS might detect this


Command and Control Bot
Bot/Zombie
Network anonymization and encryption over the Tor network
C&C Server

Quick Review

- Bots (zombies) are infected computers under malicious user control
- Bots periodically contact a command and control (C2 or C&C) server to retrieve commands
- The C&C server is also under malicious user control

Episode 5.07

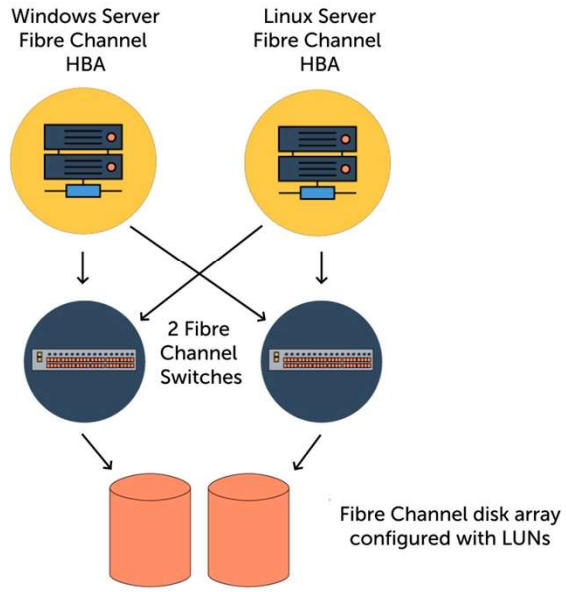
Episode title: **Disk RAID Levels**

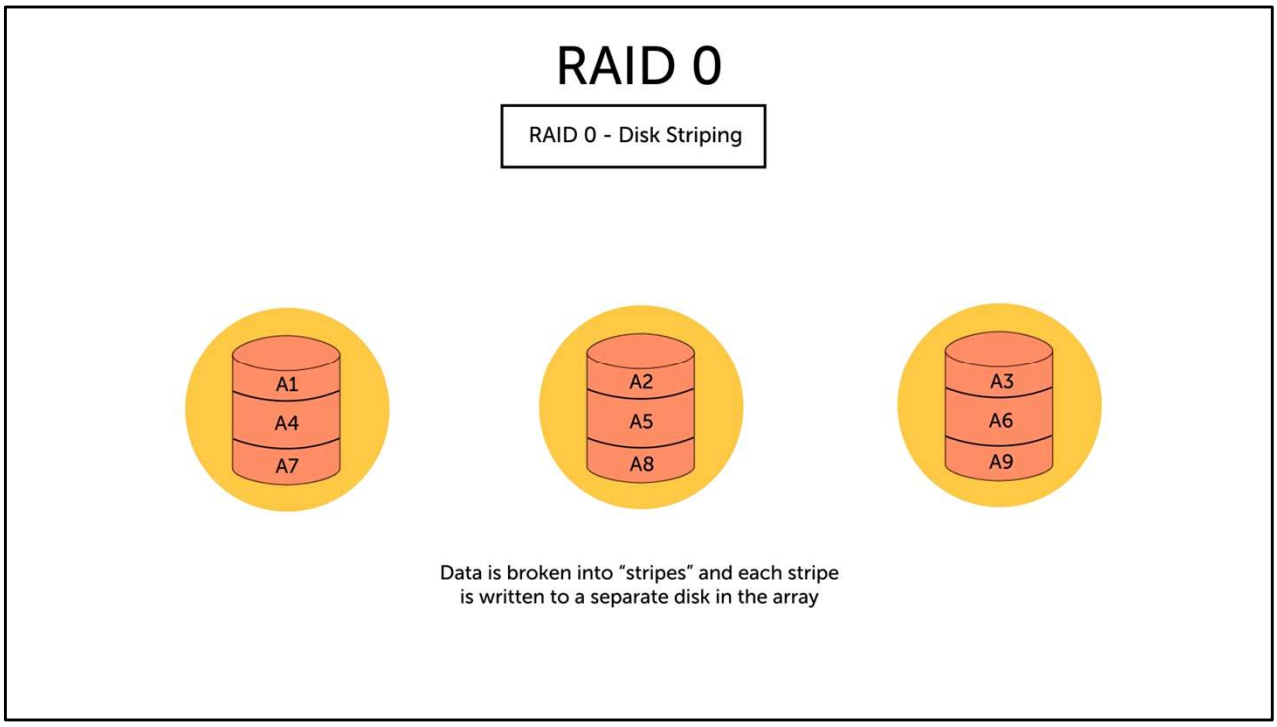
Objective: **2.5 Given a scenario, implement cybersecurity resilience.**

Redundant Array of Inexpensive Disks (RAID)

- Groups disks together to work as one
  - Better performance
  - Data high availability
- Hardware RAID controller
- Software RAID
  - Slower and less reliable than hardware RAID

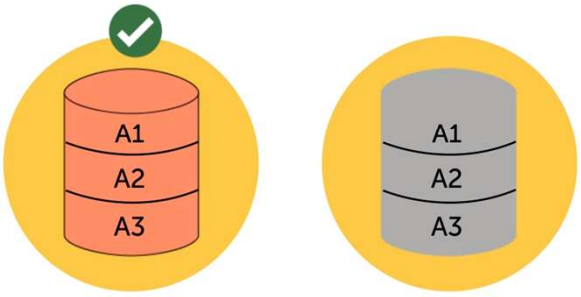
Storage Area Network (SAN) Multipathing




RAID 0
RAID 0 - Disk Striping
A1
A4
A7
A2
A5
A8
A3
A6
A9
Data is broken into "stripes" and each stripe
is written to a separate disk in the array

RAID 1

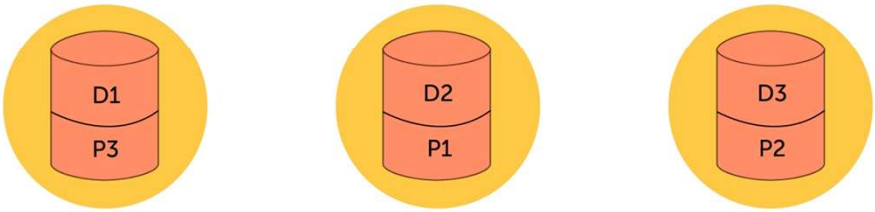
RAID 1 - Disk Mirroring



Data in its entirety is written to two separate disks

RAID 5

RAID 5 - Disk Striping with Distributed Parity



Data stripes (D) and the related parity (P) are stored on separate disks

RAID

- RAID 6
  - Requires at least 4 disks
  - Stores 2 parity stripes on each disk
  - Can tolerate failure of 2 disks
- RAID 10
  - RAID level 1, then 0
  - Disk mirroring, then striping
  - Requires at least 4 disks

Quick Review

- RAID groups disks together to improve fault tolerance and/or performance
- Hardware RAID is more reliable than software RAID
- RAID 0 and 5 use striping to increase disk I/O performance
- RAID 1 mirrors data from disk 1 to disk 2 to increase resilience to a single disk failure
- RAID 6 can tolerate a failure of 2 disks

Episode 5.08

Episode title: **Securing Hardware**

Objective:

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

2.5 Given a scenario, implement cybersecurity resilience.

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

Securing Hardware

- Limit physical access
  - Alarms, sensors, locks
  - Card cloning/skimming
- Use vendor/technology diversity
- Limit USB storage device use

Securing Hardware

- Apply firmware patches
- Use USB data blocker
  - Mitigate infection via USB port
    - Prevents data transfer from other devices
    - USB Ninja cables
  - Allows recharging but not data transfer

Trusted Platform Module (TPM)

- Used as basis for hardware root of trust
- Boot integrity
  - UEFI secure boot
  - Measured boot
  - Boot attestation
- Disk volume encryption
  - Microsoft BitLocker

Failed Machine Boot

- Causes
  - File corruption
  - Malware
  - Failing disks
  - Misconfiguration
- Remediation
  - Boot from alternative media
    - Live boot media
    - Be sure to require password
    - Revert to known state or last known-good configuration

Hardware Redundancy

- RAID
- NIC teaming
- Uninterruptible power supply (UPS)
- Power distribution unit (PDU)
- Dual power supplies

Cloud Redundancy

- Network connection to the cloud
- Load balancing
- Cross-region storage replication

Quick Review

- The first step to securing hardware is restricting physical access
- USB data blockers allow USB device power charging but prevent data transfer
- TPM detects boot-time anomalies and can store disk encryption keys
- Hardware component redundancy increases resilience against hardware component failures

Episode 5.09

Episode title: **Securing Endpoints**

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

Antivirus/Anti-Malware

- Endpoint detection and response (EDR)
  - Alarms for detected anomalies or malware infections
  - Shows up in central SIEM console
- Host-based firewalls

Host Intrusion Detection System (HIDS)

- Looks for suspicious activity
- Analyze host activity/ logs
- Detect and alert on anomalies
  - Write to log
  - Send notifications
    - E-mail
    - SMS text
    - SIEM console

Host Intrusion Prevention System (HIPS)

- HIDS functionality plus ability to block suspicious activity

Next-Generation Firewall (NGFW)

- Packet filtering firewall
  - Up to OSI layer 4
- Deep packet inspection firewall
  - Up to OSI layer 7
- Intrusion detection
- Intrusion prevention

Allow Lists

- Sometimes called whitelist
- Lists only allowed
- Can be circumvented with DLL injection attacks
- Can prevent users from
  - Installing and running malware
  - Making Windows registry changes

Block/Deny Lists

- Sometimes called blacklist
- Lists only disallowed

Quick Review

- Host-based firewalls control traffic into and out of a specific host
- A HIDS detects and reports anomalies
- A HIPS detects, reports, and blocks anomalies
- App allow lists specify which apps are allowed to run
- App deny/block lists specify only disallowed apps