Chapter 8:
Security Vulnerabilities, Threats, and Countermeasures Across Systems Layers

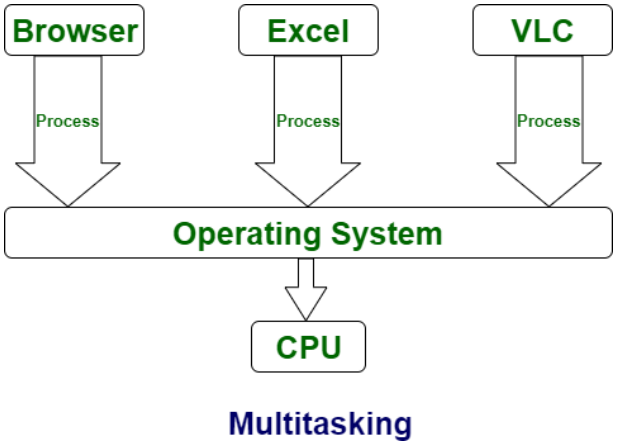
Assess and Mitigate Security Vulnerabilities

OUTLINE:

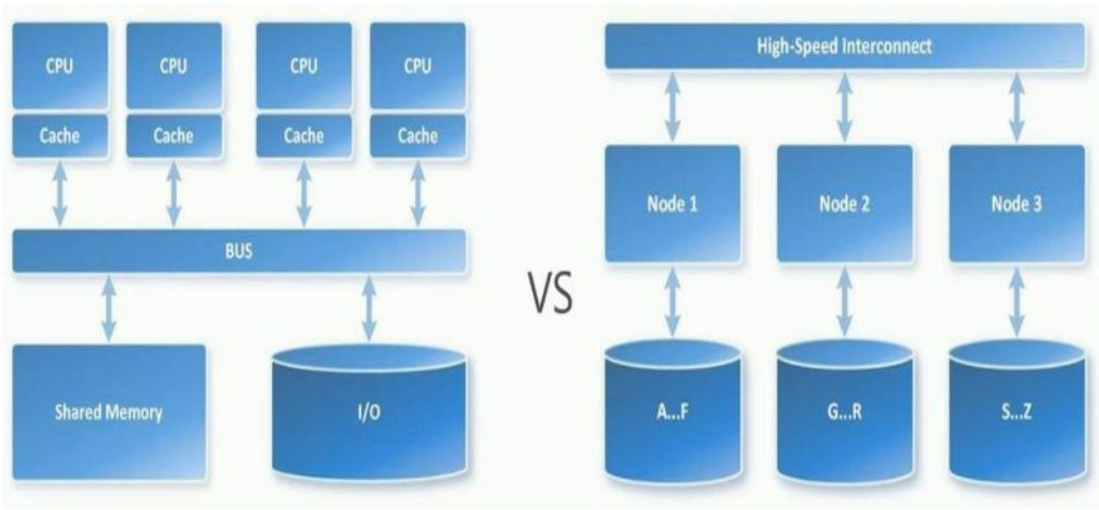
Common Security issues across system layers

- Hardware
  - Hardware Components
  - Protection Mechanisms
  - Memory Protection Mechanism
  - Input/Output Devices
  - Firmware
- Protecting Client and Server Based Systems
- Database Systems Security
- Cloud Computing Security
- Internet of Things (IoTs) Security
- Web-Based Security
- Mobile Device Security
- Operating System Security

Hardware Components

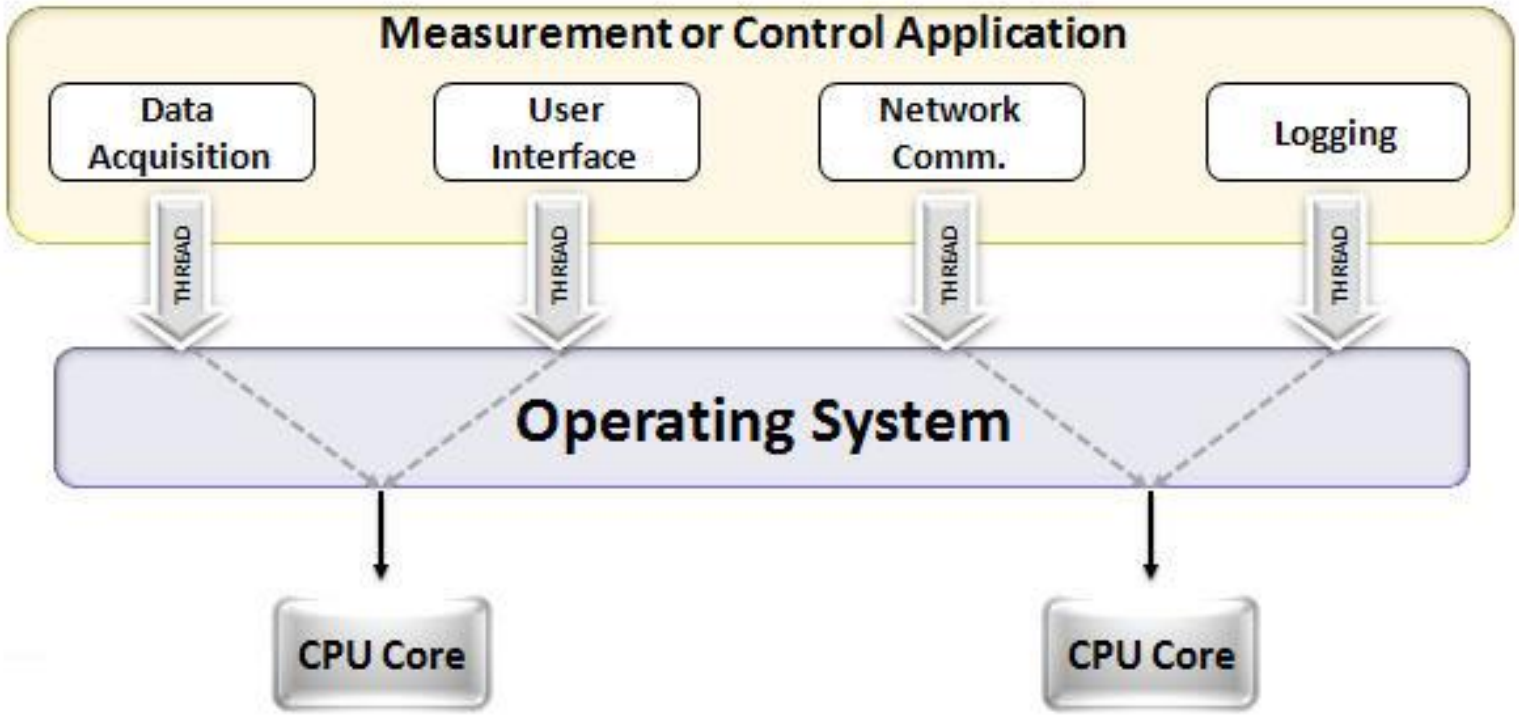


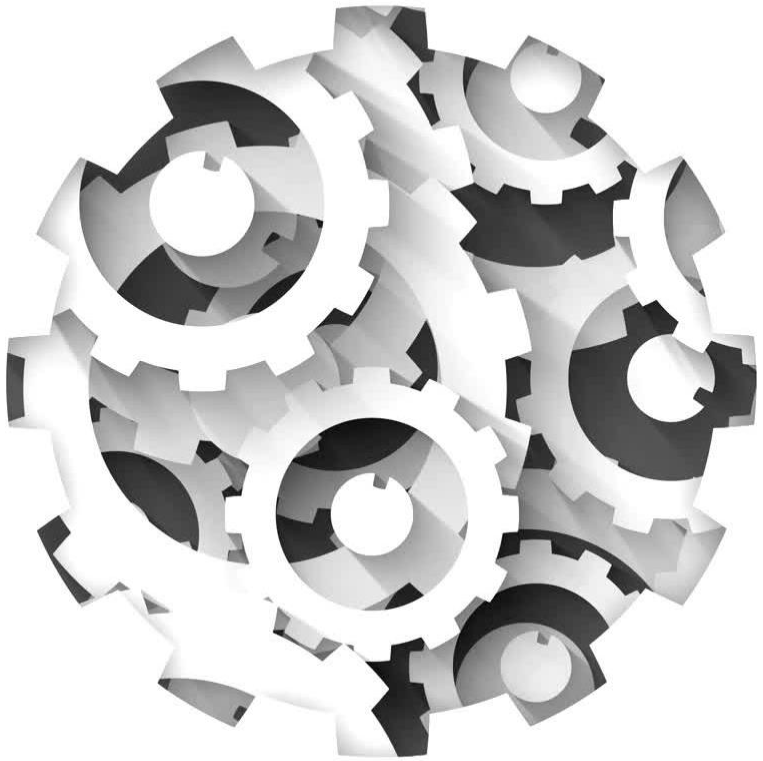
- Processor / central processing unit (CPU)
- Execution types:
  - **Multitasking:** handling two or more tasks simultaneously
  - **Multicore**: chip containing multiple independent execution cores
  - **Multiprocessing**:
    - **Symmetric Multi Processor (SMP):** shared OS & memory for all processors.
    - **Massive parallel Processing(MPP):** each processor has its own OS & memory



Hardware Components

- **Multithreading**: permits multiple concurrent program parts to be performed on a single processor (

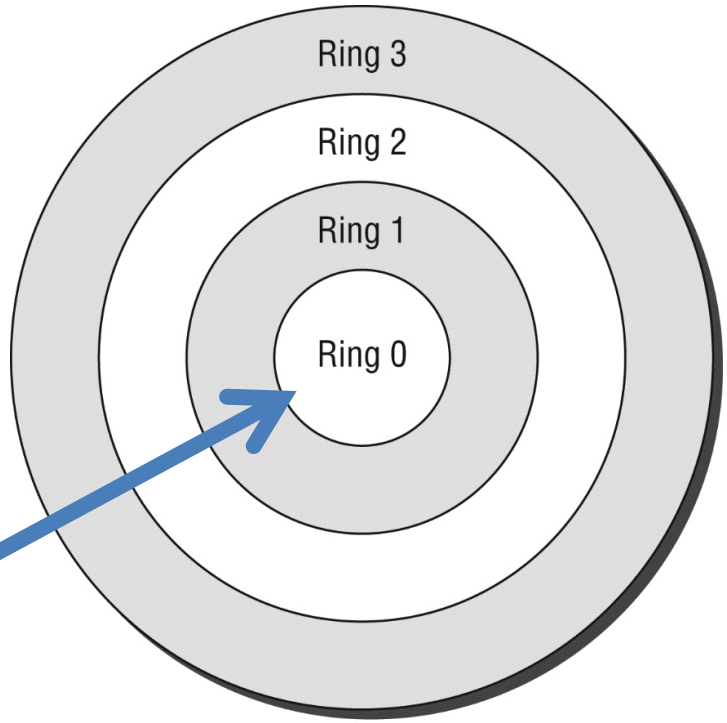




Protection Mechanisms

1. HW level: Protection rings

- Kernel mode or privileged mode
- User mode
- Mediated access/ system call



The deeper inside the circle you go, the higher the privilege level associated with the code that occupies a specific ring.

Ring 0: OS Kernel/Memory (Resident Components)
Ring 1: Other OS Components
Ring 2: Drivers, Protocols, etc.
Ring 3: User-Level Programs and Applications

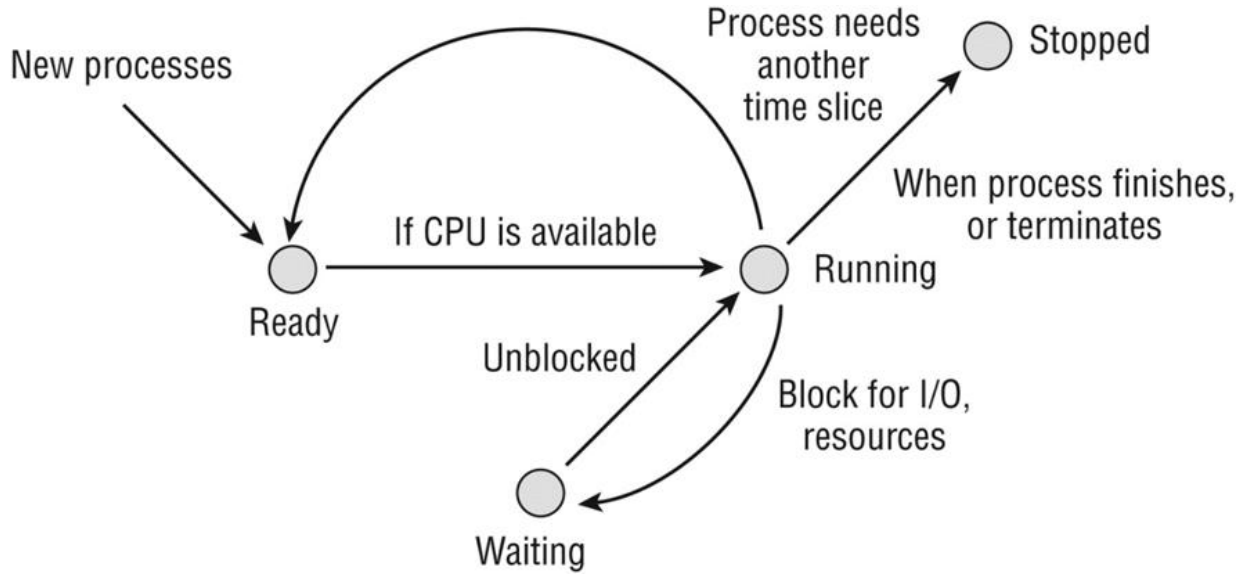
Rings 0–2 run in supervisory or privileged mode.
Ring 3 runs in user mode.

2. HW level: Operating states

- Operating states
  - **Supervisory state**: privileged – all access mode
    - Can execute all machine instructions
    - Can reference all memory offsets
  - **Operating/Problem state**: User mode
    - Limited access and executions based on the authorization
    - In problem state: privileges are low and all access requests must be checked against credentials for authorization before they are granted or denied.

3. HW level: Process states

- Processes: Ready, Waiting, Running, terminated.
- Process scheduler



Memory Protection Mechanisms

Primary Memory (1/1)

- Read only memory (ROM)
  - Programmable Read-Only Memory (PROM)
  - Erasable Programmable Read-Only Memory (EPROM)
  - Electronically Erasable Programmable Read-Only Memory (EEPROM)
  - Flash
- Random access memory (RAM)
  - Real
  - Cache
  - Registers

Memory Protection Mechanisms

- Memory protection prevents one process from accessing the memory space of another.
- By controlling access to memory, an OS can maintain a safe environment where processes can execute without interference, thus preserving data integrity and system reliability.
- In modern computing, OS like Windows, Linux, and macOS rely heavily on memory protection to manage multiple processes running simultaneously.
- Each program/ process, requires its own memory space to function properly. Memory protection ensures that these processes do not interfere with one another's memory, which could lead to **data corruption, system crashes, or unauthorized data access.**

Memory Protection Mechanisms

- Memory protection is implemented through:
  - *Hardware Mechanisms:* Memory Management Unit (MMU) within the CPU translates virtual addresses to physical addresses and enforces access controls (checks access permissions in real-time)
  - *Software Mechanisms:* includes the OS kernel, which manages the allocation and protection of memory spaces.

Memory Protection Mechanisms

- **Paging & Segmentation:** Divides memory into chunks (pages or segments) with specific access rights (read, write, execute).
- **Address Space Layout Randomization (ASLR):** Randomizes memory addresses for system and application processes, making it difficult for attackers to locate target code.
- **Data Execution Prevention (DEP):** Marks memory regions as non-executable to prevent malware from running code there.
- **Protection Keys (MPK):** Assigns a numeric key to memory blocks, allowing fast permission changes.
- **Process Isolation:** Prevents one application from reading/writing the memory of another, preventing unauthorized data theft.
- **Kernel Protection:** Protects the operating system's memory space from user-level malware.
- **Buffer Overflow Mitigation:** Blocks attacks where malicious actors try to overwrite memory to change program flow.

Secondary Memory 1/2

- Secondary memory protection refers to the strategies, technologies, and practices designed to secure non-volatile storage devices (such as HDDs, SSDs, USB drives and cloud storage) against unauthorized access, data theft, malware, and tampering.
- Secondary storage retains data when power is off, making it a prime target for long-term data exfiltration.
- ***Secondary Memory Protection Mechanisms:***
  - **Encryption (At-Rest):** Utilizing full-disk encryption or file-level encryption - if a device is lost, stolen, or accessed unauthorized, the data remains unreadable.
  - **Immutable Backups & Object Locking:** Use technologies that create "read-only" copies of data, which cannot be modified, encrypted, or deleted by ransomware.
  - **Air-Gapping:** Maintaining physical separation between secondary storage and the network to isolate backups from cyber-attacks.
  - **Access Controls & Authentication:** Restricting access to sensitive storage partitions using multi-user authentication (MUA) or strict permissions to prevent accidental or malicious deletion.
  - **Physical Security:** Locking down servers and removing removable media (USB drives).

Input/Output Devices

- **Input:** Keyboards, mice, scanners, microphones, cameras.
- **Output:** Monitors, printers, projectors, speakers.
- **Storage/Peripherals:** USB drives, external hard drives, IoT sensors.
- **Modems/Routers:** Ubiquitous broadband and wireless connectivity is common nowadays

Can these basic devices present security risks to a system?

Input/Output Devices - Key Protection Strategies

Physical Security: Secure devices and cables to prevent tampering or unauthorized swapping. Use locks, surveillance, and access badges to limit physical access.

Port Control: Disable/ physically remove unused connection ports like USB, CD/DVD drives to prevent malware insertion or data theft.

Software Updates: Consistently update firmware on I/O devices and software drivers on computers to patch vulnerabilities.

Authentication & Access Control: Implement strong passwords or biometric authentication for accessing devices and systems.

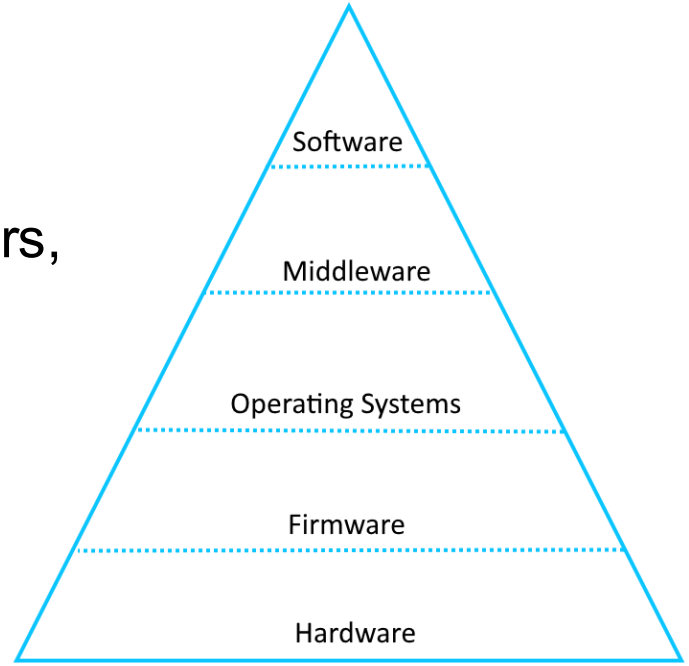
Network Segmentation: Isolate IoT and peripheral devices on a separate network to prevent a compromised device from affecting the main network.

Encryption: Utilize secure communication protocols like TLS to encrypt data transmission between devices.

Input Validation: Implement input validation to ensure integrity and protect against malicious code

Firmware

- Called Microcode: software that is stored in a ROM chip
- Basic input/output system (BIOS)
  - BIOS is usually stored on an EEPROM chip
  - The process of updating the BIOS is known as flashing the BIOS
  - malicious code embedding itself into BIOS/**phlashing the BIOS**
- Unified Extensible Firmware Interface (UEFI)
  - Used since 2011 to replace BIOS
  - UEFI is a more advanced interface between hardware and the operating system
- HW Device firmware, e.g. printers, modems
  - These devices has mini operating systems
  - Sorted as a firmware in EEPROM



Protecting Client and Server Based Systems

Protecting Client-Based System

- Client-side security secures the user endpoint from where they access the network
- A client-side attack is any attack that is able to harm a client e.g., *A malicious website that transfers malicious code to a vulnerable browser running on the client*.

Client-Side Protection Measures

It focuses on securing end-user devices (laptops, mobile devices) and browser-based applications. It includes:

Endpoint Security: Installing antivirus and Endpoint Detection and Response (EDR) solutions to protect against malware.

Browser Security: Implementing Content Security Policies (CSP) to block malicious scripts and using secure cookie attributes (HttpOnly, Secure, SameSite).

Input Sanitization: Validating all user inputs on both the client and server side to prevent script injection.

Code Obfuscation: Making client-side code (JavaScript) difficult to understand to deter reverse engineering.

Patching and Updating: Ensuring browsers, operating systems, and installed software are updated to patch vulnerabilities.

Protecting Server Based Systems

- Server-side security protects data and applications
- Main concern in such systems include:
  - Data flow control
  - Management between processes, devices, networks, or communication channels with minimal delay
  - Load balancing
  - Efficient transmission with minimal delays or latency
  - Reliable throughput using hashing and confidentiality protection with encryption

- **A denial-of-service & DDoS attacks can be a severe detriment to data flow control.**

Protecting Server Based Systems

Servers are often the primary target as they house sensitive data, making them critical points for attack. It Includes:

- **Firewalls & WAFs:** Next-Generation Firewalls (NGFW) monitor network traffic and block dangerous traffic. Web Application Firewalls (WAF) protect against application-layer attacks like SQL injection and cross-site scripting (XSS).
- **Patch Management:** Regular updates for operating systems and applications.
- **Access Control:** Implementing Multi-Factor Authentication (MFA) and Role-Based Access Control (RBAC) to ensure only authorized personnel can access critical systems.
- **Encryption:** Protecting sensitive information if physical drives are stolen.
- **Intrusion Detection/Prevention (IDS/IPS):** Monitoring network traffic in real-time to identify and block suspicious activities.
- **Server Hardening:** Disabling unnecessary services, closing unused ports, and using hardened OS configurations.
- **DDoS Mitigation:** Tools and strategies to protect against DoS & DDoS attacks that aim to make services unavailable

Database Systems Security

Database Systems Security

- Database systems security protects databases from cyber threats, ensuring data confidentiality, integrity, and availability.
- It involves securing the database management system (DBMS), applications, physical servers, and network infrastructure through access controls, encryption, and regular auditing.
- Key practices include robust authentication, patch management, and monitoring for threats like SQL injection and unauthorized access

Common Database Threats

- **SQL/NoSQL Injection Attacks:** Attackers insert malicious code into queries to manipulate the database.
- **Insider Threats:** Malicious or careless employees with legitimate access.
- **Misconfigured Databases:** Improper settings, such as default passwords or open ports, leading to vulnerabilities.
- **Weak/Missing Encryption:** Allowing sensitive data to be read if the system is breached.
- **Data Loss/Ransomware:** Attackers encrypt or steal data, demanding ransom.

Best Practices for Protecting Database Systems

- **Apply Least Privilege:** Grant users only the minimum access necessary for their job roles.
- **Use Data Masking:** Hide sensitive data in non-production environments.
- **Secure Backups:** Regularly back up data and encrypt the backups to ensure availability.
- **Vulnerability Scanning:** Frequently test for vulnerabilities and perform penetration testing.
- **Ensure Physical Security:** Secure the physical or virtual servers hosting the database

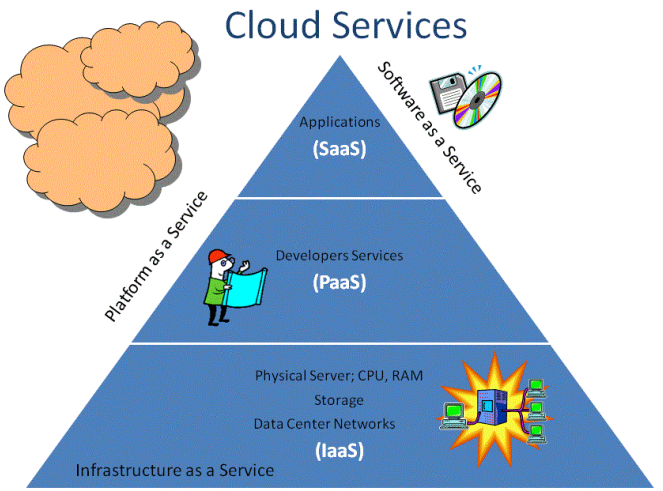
Cloud Computing Security

Cloud-Based Systems & Cloud Computing

- Cloud security involves protecting data, applications, and infrastructures in cloud environments (AWS, Azure, Google, Alibaba) from threats.
- It uses tools like encryption, firewalls, and IAM to manage risks such as data breaches, misconfigurations, and API vulnerabilities.
- **Hypervisor, virtual machine monitor (VMM)**

  Is the component of virtualization that creates, manages, and operates the virtual machines.
- **Elasticity:**

  Flexibility of virtualization and cloud solutions to expand or contract based on need
- **Cloud computing**
  - PaaS
  - SaaS
  - IaaS



Cloud Security Components & Strategies

Shared Responsibility Model: The Cloud Service Provider (CSP) secures the infrastructure ("of" the cloud), while the customer secures their data, applications, and access ("in" the cloud).

Identity and Access Management (IAM): Crucial for controlling user access to resources, often requiring Multi-Factor Authentication (MFA) to prevent account hijacking.

Data Protection: Involves encrypting data at rest and in transit, and ensuring secure backups.

Visibility and Compliance: Utilizing tools like Cloud Security Posture Management (CSPM) to monitor configurations and ensure regulatory compliance.

Cloud-Native Protection: Adopting platforms like CWPP (workload protection) and CNAPP (cloud-native application protection) for real-time monitoring and threat detection.

Common Cloud Security Threats

Misconfigurations: The leading cause of breaches, where cloud services are improperly secured.

Unauthorized Access/Account Hijacking: Exploiting weak credentials to gain entry to cloud environments.

Insecure APIs: Vulnerabilities in the interfaces used to manage cloud services.

Insider Threats: Risks from within the organization

Internet of Things (IoTs) Security

Internet of Things (IoTs) Security

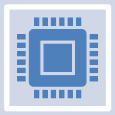
- IoT security is the specialized cybersecurity practice of protecting internet-connected devices—like cameras, sensors, and smart appliances—from cyberattacks, as these devices often lack robust built-in security.
- Key threats include malware, data theft, and botnets used for DDoS attacks.
- Effective defense requires robust authentication, regular firmware updates, network segmentation, and encryption.

Key IoT Security Challenges & Threats

Vulnerabilities in Firmware/Software: Many IoT devices are rushed to market with little, if any, security considerations, leading to weak or outdated operating systems.

Lack of Encryption: A vast majority of IoT device traffic is unencrypted, making data transmission vulnerable to interception.

Default Credentials: Attackers easily compromise devices using default usernames and passwords.

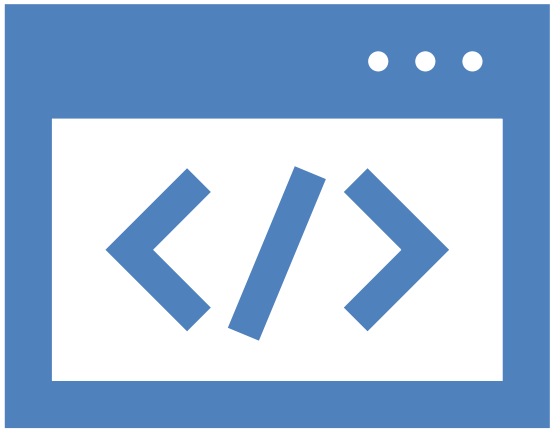


IoT as Attack Vectors: Once compromised, devices act as gateways, allowing attackers to access the broader, more sensitive corporate or home network.

Unsupported Software: 83% of medical IoT devices, for example, run on unsupported operating systems, providing a significant target for hackers.

Best Practices for Securing IoT Devices

- **Change Default Passwords:** Immediately set strong, unique passwords for every device.
- **Update Firmware/Software:** Regularly update device firmware to patch known vulnerabilities.
- **Network Segmentation:** Isolate IoT devices on a separate network to prevent attackers from accessing main computers, servers, or personal data.
- **Implement Zero Trust:** Treat every device, user, and connection as a potential threat by requiring continuous authentication.
- **Disable Unused Features:** Turn off unused features like Bluetooth, remote management to reduce potential entry points.
- **Use Enhanced Authentication:** Enable Multi-Factor Authentication (MFA) whenever available.



Web-Based Security

Assess and Mitigate Vulnerabilities in Web-Based Systems

- Web-based cybersecurity involves protecting websites, web applications, and APIs from malicious attacks, preventing data breaches, and ensuring service availability.
- **Key components of Web Security include:**
  - Web Application Firewalls (WAF): Filter and monitor HTTP traffic between web apps and the internet to block threats.
  - Vulnerability Scanning: Automated tools that scan web applications for known weaknesses.
  - Secure Coding Practices: Developing code with security in mind to prevent common vulnerabilities, often aligned with OWASP Top 10 guidelines.
  - API Security: Protecting application programming interfaces from unauthorized access or data leaks

Planning a security evaluation or penetration test of an organization's web services includes:

- View hosted web pages
- Explore what automation technologies in use
- Look for information that should not have been posted
- How files and backups are handled
- evaluating the site's transmission security
  - Secure Sockets Layer (SSL)/Transport Layer Security (TLS)
- Evaluate authentication and session management
- Evaluating the cryptography of the site and the methods used for data validation and sanitization
- Other activities include: checking for DoS defences, evaluating risk responses, and testing error handling.

Assess and Mitigate Vulnerabilities in Web-Based Systems

Common Threats and Defense Techniques:

- **SQL Injection (SQLi):** Attackers manipulate database queries.
  - Mitigation: Validating and sanitizing all user inputs.
- **Cross-Site Scripting (XSS):** Injecting malicious scripts into trusted websites.
  - Mitigation: Escaping user-provided content.
- **DDoS Attacks:** Flooding servers with traffic.
  - Mitigation: Utilizing DDoS mitigation services.
- **Broken Access Control:** Unauthorized access to user accounts or data.
  - Mitigation: Implementing strict authentication mechanisms

Mobile Devices Security

Assess and Mitigate Vulnerabilities in Mobile Systems

- **Device Security**
  - The range of potential security options or features available for a mobile device which are useless if not activated
- **Application Security**
  - Focuses on the security of applications and functions used on those devices
- **BYOD Concerns**
  - *Bring your own device (BYOD)* is a policy that allows employees to bring their own personal mobile devices into work and use those devices to connect to (or through) the company network to business resources and/or the internet.
  - This may improve employee morale and job satisfaction

Device Security

Be sure to consider the security options of a new device before you make a purchase decision.

- Full device encryption
- Remote wiping
- Lockout
- Screen locks
- GPS
- Application control
- Storage segmentation
- Disabling unused features

Application Security

By default, No app can execute on the device but those I give them the privilege

Security concerns include:

- Key management: Most of the failures of a cryptosystem are based on the key management rather than on the algorithms
- Credential management: storage of credentials in a central location
- Authentication: good authentication include: passwords and biometrics
- Geotagging: can be used to determining when a person normally performs routine activities
- Encryption: When this is available in the mobile device, it should be enabled
- Application whitelisting: It is a security options prohibits unauthorized software from being able to execute (deny by default and allow by exception). It is the opposite of blacklisting: allow by default and deny by exception

BYOD Concerns

Bring your own device (BYOD)

- Not all mobile devices have security features
- Allowing such devices to connect to the production network introduces security risks
- A BYOD policy mandates specific devices to reduce such risks
- What about employees who are unable to purchase a compliant device?

Company owned, personally enabled (COPE)

Choose your own device (CYOD)

Virtual desktop infrastructure (VDI)/ virtual mobile infrastructure (VMI)

BYOD Concerns

Data ownership

- When a personal device is used for business tasks, mixing of personal data and business data is likely to occur.
- Some devices can support storage segmentation, but not all devices can provide datatype isolation
- If the device is lost or stolen, shall the company trigger a remote wipe?

Support ownership

- When an employee's mobile device experiences a failure, a fault, or damage, who is responsible for the device's repair, replacement, or technical support?

Patch/update management

- The mobile device policy should define the means and mechanisms of patch management for a personally owned mobile device. Who is responsible to do update user/organization? Shall the organization test updates prior to installation?

Antivirus management

- The mobile device policy should dictate whether antivirus, antimalware, and antispyware scanners are to be installed on mobile devices.

Forensics

- The mobile device policy should address forensics and investigations as related to mobile devices.

Privacy

- When a personal device is used for business tasks, the user often loses some or all of the privacy they enjoyed prior to using their mobile device at work.

Onboarding/offboarding

- The mobile device policy should address personal mobile device onboarding and off-boarding procedures.

Operating System (OS) Security

Essential Security Protection Mechanisms

- **The OS must employ protection mechanisms to keep the computing environment stable and to keep processes isolated from each other.**

- Computer system designers should adhere to a number of common protection mechanisms when designing secure systems includes:
  - **Technical Mechanisms**
  - **Policy Mechanisms**

Technical Mechanisms

- **Layering:** Implement a structure similar to the ring model used for operating modes
- **Abstraction:** access controls and operation rights are assigned to groups of objects
- **Data hiding:** important characteristic in multilevel secure systems. data hiding relies on placing objects in security containers that are different from those that subjects occupy to hide object details from those with no need to know about them.
- **Process Isolation:** requires that the operating system provide separate memory spaces for each process's instructions and data.
- **Hardware segmentation:** similar to process isolation. The main difference is that hardware segmentation enforces these requirements through the use of physical hardware controls rather than the logical process isolation controls imposed by an operating system.

Policy Mechanisms

- The policy mechanisms are extensions of basic computer security policy, but the applications described in this section are specific to the field of computer architecture and design.

  - least privilege : minimal required privileges
  - Separation of duties: task should be assigned to more than one person
  - Accountability: Who do what? Recording activities that include administrators or other trusted individuals with high levels of privilege activities on the system

OS Hardening?

THE END…