جامـــــعـة المـــــســتـقـبـل

AL MUSTAQBAL UNIVERSITY

قـــســـــم الامـــــــــــــن الـــــــــــــــــــــســـيبرانــــــــــــي

Department of Cyber Security

Subject:

Software Security

Class:

second

Lecturer:

Dr. Suha Alhussieny

Lecture: (1)

Software and System Security Principles

Section One: Software and System Security Principles

Computer software, also called software, is a set of instructions and documentation that tells a computer what to do or how to perform a task. Software includes all different programs on a computer, such as applications and the operating system.

- Applications are programs that are designed to perform a specific operation, such as a game or a word processor.
- The operating system (e.g. Mac OS, Microsoft Windows, Android and various Linux distributions) is a type of software that is used as a platform for running the applications, and controls all user interface tools including display and the keyboard.

The word software was first used in the late 1960s to emphasize on its difference from computer hardware, which can be physically observed by the user. Software is a set of instructions that the computer follows. The word firmware usually refers to a piece of software that directly controls a piece of hardware. The firmware for a CD drive or a modem are examples of firmware implementation.

Software security refers to a set of practices that help protect software applications and digital solutions from attackers. Developers incorporate these techniques into the software development life cycle and testing processes. As a result, companies can ensure their digital solutions remain secure and are able to function in the event of a malicious attack.

Software Security Important: Secure software development is incredibly important because there are always people out there who seek to exploit business data. As businesses become more reliant on software, these programs must remain safe and secure. With strong software security protocols in place, you can prevent attackers from stealing potentially sensitive information such as credit card numbers and trade secrets, and build trust among users. The theft of critical data can be catastrophic for customers and businesses alike. Malicious actors can abuse sensitive information and even steal users' identities. Additionally, companies can face legal penalties in the event of a data breach and suffer reputational harm.

Businesses can work to protect critical data by implementing software security techniques into their development life cycles. Applying security techniques enables organizations to proactively identify system vulnerabilities and better protect their software.

Authentication

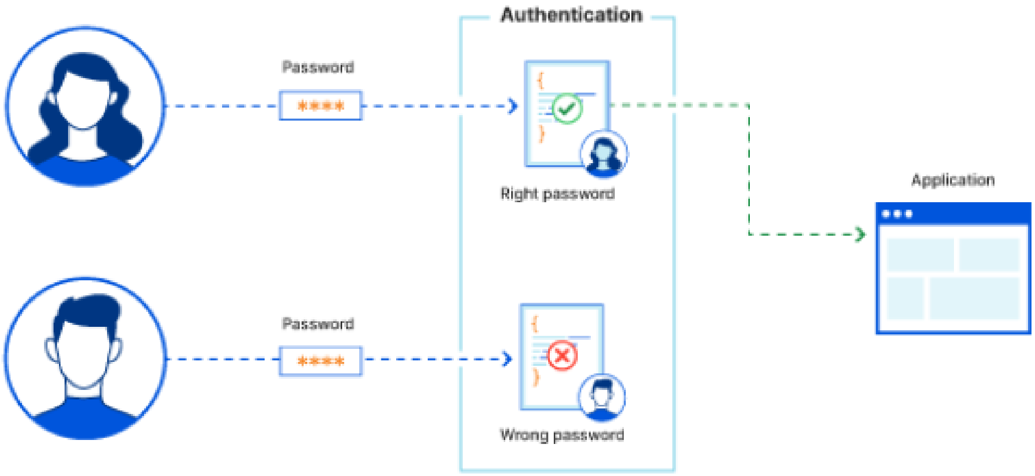
Authentication is a fundamental aspect of software security, ensuring that only authorized users or systems can access the software and its resources. It involves verifying the identity of a user, device, or other entity before granting access to the system. Here are the key components and methods of authentication in software security:

1. Password-Based Authentication

Strength and Complexity: Users are required to create strong passwords that combine uppercase and lowercase letters, numbers, and special characters to resist brute-force attacks.

2. Multi-Factor Authentication (MFA)

- **Something You Know:** This is usually a password or PIN.
- **Something You Have:** A physical token, smartphone, or a one-time password (OTP) generated by an authenticator app.
- **Something You Are:** Biometric authentication, such as fingerprint, facial recognition, or iris scanning.
- MFA adds an additional layer of security by requiring two or more forms of authentication.

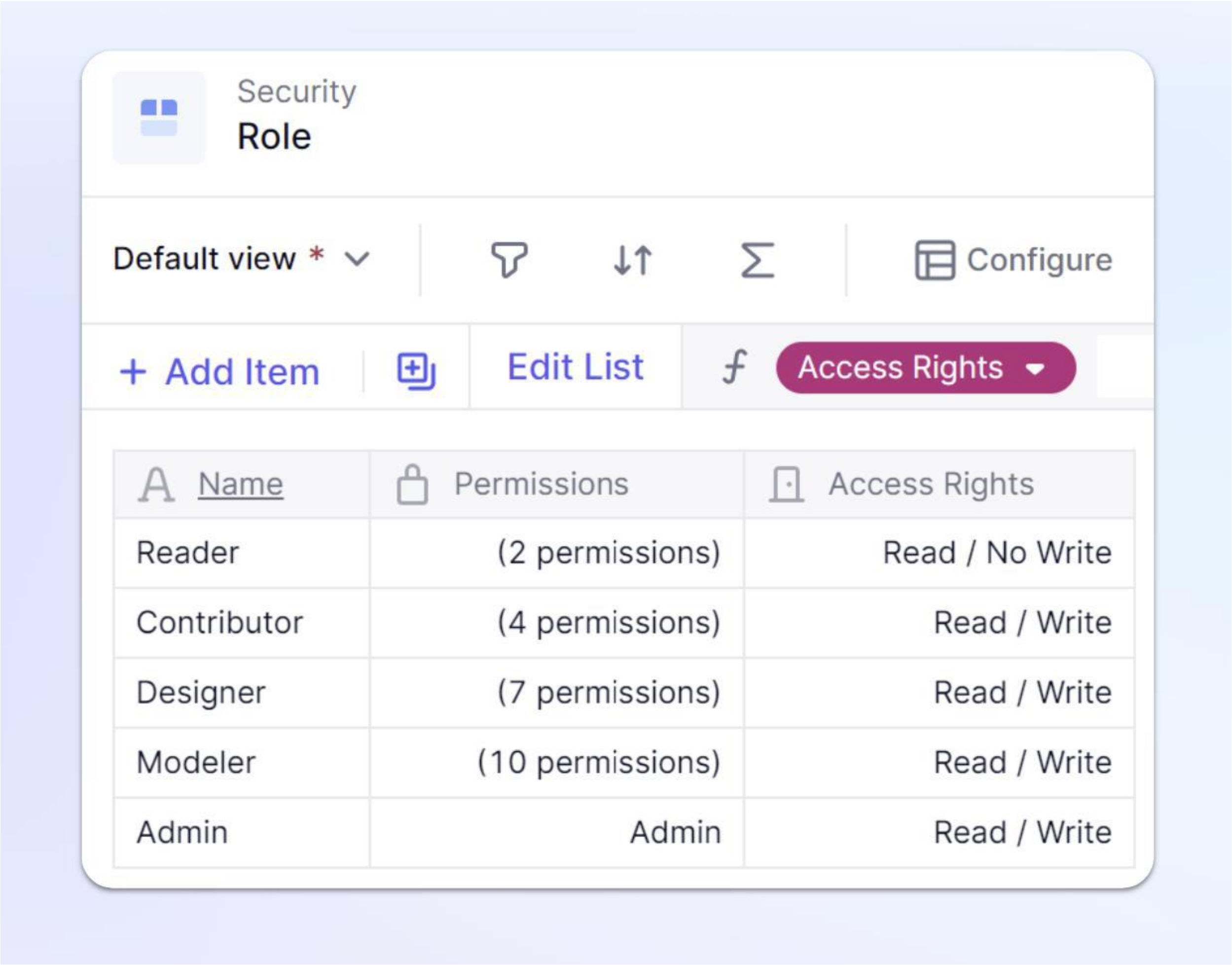


Access Rights

Access control policies are a fundamental component of software development that governs the permissions and restrictions placed on users accessing a system or its resources. These policies define the rules and guidelines for granting or denying access to different functionalities, data, or areas within the software. There are several types of access control policies that can be implemented in software development to manage and enforce access to resources. These policies determine how permissions are granted or denied based on various factors, such as user roles, attributes, or predefined security levels.

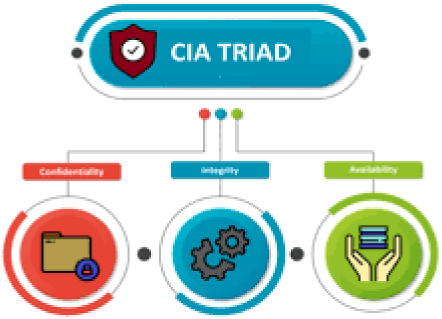
1. **Role-Based Access Control (RBAC). In RBAC**, access rights are assigned to users based on their roles within the system. For example, an administrator may have full access to all functionalities, while a regular user may only have access to specific features.

2. **Attribute-Based Access Control (ABAC)** is another type of access control policy that considers additional attributes or characteristics of users when granting or denying access. These attributes can include user location, time of access, device used, or any other relevant information.



Name	Permissions	Access Rights
Reader	(2 permissions)	Read / No Write
Contributor	(4 permissions)	Read / Write
Designer	(7 permissions)	Read / Write
Modeler	(10 permissions)	Read / Write
Admin	Admin	Read / Write

Confidentiality, Integrity, and Availability



Confidentiality, Integrity, and Availability

Confidentiality, Integrity, and Availability (CIA) are the three core principles of information security, often referred to as the CIA triad. These principles form the foundation for designing and evaluating the security of systems, data, and processes. Here's a detailed overview of each component:

1. Confidentiality

- **Definition:** Confidentiality ensures that sensitive information is accessed only by authorized individuals or systems and is protected from unauthorized disclosure.
- **Purpose:** To protect information from being disclosed to unauthorized parties, thereby preventing breaches of privacy and security.
- **Key Concepts and Practices:**
    - o **Encryption**
    - o **Access Controls**
- **Examples of Confidentiality Breaches:**
    - o **Data Leaks:** Sensitive information, like personal data or trade secrets, being exposed due to inadequate access controls.
    - o **Unauthorized Access:** Hackers gaining access to confidential information through phishing, malware, or other attack vectors.

2. Integrity

- **Definition:** Integrity ensures that data is accurate, consistent, and has not been tampered with or altered by unauthorized parties.
- **Purpose:** To maintain the trustworthiness and accuracy of information, ensuring that it remains unchanged from its original state unless properly authorized.
- **Key Concepts and Practices:**
    - o **Checksums and Hashing:** Using checksums or cryptographic hashing (e.g., SHA-256) to detect alterations in data. Any changes to the data will result in a different hash value.

o **Digital Signatures:** Applying digital signatures to data to verify its origin and ensure it has not been modified during transmission.

- **Examples of Integrity Breaches:**

o **Data Tampering:** Unauthorized modification of data, such as altering financial records, which can lead to fraud or misinformation.

o **Man-in-the-Middle Attacks:** Attackers intercepting and altering data during transmission, potentially compromising the integrity of communication.

3. Availability

- **Definition:** Availability ensures that information and systems are accessible and usable when needed by authorized users.
- **Purpose:** To ensure that systems and data are available to users in a timely manner, supporting the continuity of business operations.
- **Key Concepts and Practices:**

o **Redundancy:** Implementing redundant systems, such as backup servers, failover clusters, and data backups, to ensure continuous availability even if a component fails.

o **Load Balancing:** Distributing workloads across multiple systems or servers to prevent overload and ensure that resources are available even under heavy usage.

- **Examples of Availability Breaches:**

o **DDoS Attacks:** Overloading a system with traffic to make it unavailable to legitimate users.

o **Hardware Failures:** System crashes or server outages leading to unavailability of critical services.

H.W/

1. What is computer software?

2. Which of the following is an example of application software?

3. The operating system is mainly responsible for:

4. The term 'software' was first used in:

5. Firmware is best described as:

6. What is the main goal of software security?

7. Why is secure software development important?

8. Which of the following is a consequence of poor software security?

9. Authentication is the process of:

10. Which is an example of something you KNOW in MFA?

11. Which MFA factor represents 'something you HAVE'?

12. Biometric authentication is based on:

13. What does MFA improve?

14. Access control policies mainly define:

15. RBAC assigns access based on:

16. ABAC differs from RBAC because it considers:

17. The CIA triad stands for:

18. Confidentiality ensures that data is:

19. Which technique supports confidentiality?

20. Integrity is mainly concerned with:

21. Which technique is used to verify data integrity?

22. A man-in-the-middle attack mainly affects:

23. Availability ensures that systems are:

24. Which practice improves availability?

25. A DDoS attack targets which CIA principle?