Lecture 3

Mobile Operating System Security and OWASP Mobile Top 10

- Mobile Operating System
- Operating System Security
- Mobile Application Security OWASP Mobile Top 10

1. Mobile Operating System

A mobile operating system (OS) is a software platform that serves as the foundation for smartphones, tablets, and other mobile devices. It manages hardware resources, provides a user interface (UI), and enables communication between software and a device's hardware components.

Mobile OSs are designed to be lightweight, efficient, and optimized for battery-powered devices with limited resources.

1.1. Features And Functions Of Mobile OS

- **User interface (UI):** The mobile OS provides a graphical user interface (GUI) that lets users interact with their devices through touchscreens, buttons, and other input methods.

- **App management:** Mobile OSs manage an app's installation, launching, and termination. They provide app stores or marketplaces where users can download and install new apps and manage updates and removals.

- **Hardware abstraction:** The OS abstracts the underlying hardware components, such as the processor, memory, storage, camera, sensors, and more.

- **Security:** Mobile OSs implement security measures to safeguard data, prevent unauthorized access, and safeguard against malware infections. They include features like app sandboxing, encryption, secure boot, and permission systems that allow users to control app access to sensitive data and device features.

- **Connectivity:** The OS enables various types of wireless and cellular connectivity, such as Wi-Fi, Bluetooth, Global Positioning System (GPS), Near Field Communication (NFC), and mobile data networks (3G, 4G, and

5G). These features let devices communicate with each other and access the Internet.

- **Multitasking:** Mobile OSs support multitasking, allowing users to switch between multiple apps seamlessly.

- **Notifications:** The OS manages notifications from various apps, displaying them on the lock screen or in a notifications center.

- **Updates:** Mobile OSs receive regular updates from their developers to introduce new features, fix bugs, and enhance security. Users can download and install these updates to keep their devices current.

1.2. Mobile Operating Systems Types

1- Android: Developed by Google.
2- iOS: Developed by Apple.

3- Windows 10 Mobile.
4- KaiOS.
5- Tizen.
6- Bada (Samsung Electronics).
7- Blackberry OS.
8- Symbian OS.
9- Harmony OS.
10- Palm OS.
11- WebOS (Palm/HP).

2. Operating System Security

The process of ensuring OS availability, confidentiality, integrity is known as operating system security. OS security refers to the processes or measures taken to protect the operating system from dangers, including viruses, worms, malware, and remote hacker intrusions.

Operating system security comprises all preventive-control procedures that protect any system assets that could be stolen, modified, or deleted if OS security is breached.

2.1 Android and iOS Operating System Security

Mobile devices are an inseparable part of our lives. They have made it possible to access all the information and services anywhere at any time. Almost all of the organizations try to provide a mobile device-based solution to its users. However, this convenience has arisen the risk of losing personal information and has increased the threat to security. It has been observed recently that some of the mobile device manufacturers and mobile apps developers have lost the private information of their users to hackers. Android and iOS are the major operating systems for mobile devices .

The major cause of security threats to mobile devices comes from the mobile apps installed by third-party developers or in some cases from the mobile apps installed from the app stores.

2.2 Security components for mobile operating systems

- **Application Sandboxing :** is an important layer of protection for mobile OS and increases security. Application sandboxing acts as a container and prevents the applications from gaining access to the system or other applications that may contain the virus and malicious code.

- **Application Isolation :** Applications are executed and isolated in the sandbox environments of their device. During execution, as applications are isolated in their environment, they are unable to make changes to other applications

- **Data Encryption :** Encryption is the process of converting and translating data into code to prevent unauthorized access .It is an eminent approach and well-known for being effective in data security. If the data is encrypted, then a password or special key is necessary to decrypt the file and gain access to retrieve the encrypted information. Data that is encrypted is called ciphered text and unencrypted data is simply called a plain text. Encryption is necessary for mobile devices.

- **Inbuilt Antivirus :** The most common three types of malware that are known to affect the mobile devices are spyware, viruses, and trojans.

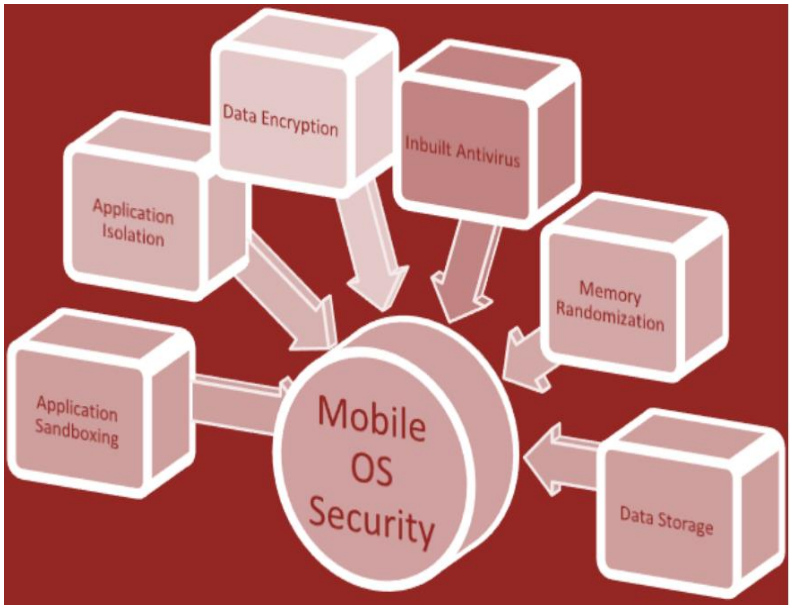
    - A virus is a software containing malicious code.
    - Spyware collects information from users without their consent and knowledge.
    - A trojan is a misleading software that depicts some normal functionality but instead is harmful to malicious content.

  An antivirus usually protects against such type of attacks. Android and iOS provide their inbuilt antivirus support system.

- **Memory Randomization :** To prevent malicious attack or virus from locating the exact position or memory region, memory is allocated randomly. This is done as the running application's memory allocation becomes hard to find for the malicious code. In memory allocation not only the memory of running application is allocated randomly but also the shared libraries and other as such things in regard to the device ..

- **Storage Format :** Data can be stored both internally and externally on mobile devices. This process is known as data storage. Mobile devices come with built-in storage space. The user must choose what kind of sensitive data they want to store on it as well as to make sure it is protected and secure.

    - In Android devices, both internal and external (if necessary) storage device is provided. External storage can be stored on an SD card installed additionally.
    - iOS does not support its devices to have external storage, it only allows built-in storage. Even within the internal storage, it requires permission to make use or manipulate the data.The following figure illustrates the security components for mobile operating systems.



2.3 Sandboxing and Permission Model

- Permissions on Android

App permissions help support user privacy by protecting access to the following:

- **Restricted data**, such as system state and users' contact information
- **Restricted actions**, such as connecting to a paired device and recording audio

- Types of permissions on Android

Android categorizes permissions into different types, including install-time permissions, runtime permissions, and special permissions. Each permission's type indicates the scope of restricted data that your app can access, and the scope of restricted actions that your app can perform, when the system grants your app that permission.

1.Normal permissions : These permissions allow access to data and actions that extend beyond your app's sandbox but present very little risk to the user's privacy and the operation of other apps.

2.Signature permissions: The system grants a signature permission to an app only when the app is signed by the same certificate as the app or the OS that defines the permission.

3.Special permissions: Special permissions correspond to particular app operations.

4.Runtime permissions: Runtime permissions, also known as dangerous permissions, give your app additional access to restricted data or let your app perform restricted actions that more substantially affect the system and other apps.

5.Install-time permissions: Install-time permissions give your app limited access to restricted data or let your app perform restricted actions that minimally affect the system or other apps.

2.4 Ios permission model

Ios uses a Fine-grained permissions

Granularity: Fine-grained permissions provide a high level of granularity, with specific access controls for individual resources.

Flexibility: With fine-grained permissions, you have a lot of flexibility in granting access. You can give a user permission to access just one file in a folder, or a single row in a database table.

Management overhead: The downside to fine-grained permissions is that they require more effort to manage. When you have permissions set at a very granular level, there are more permissions to keep track of and maintain.

Use cases: Fine-grained permissions are best used for sensitive information or when you need to comply with strict regulatory requirements, such as in healthcare applications where privacy is paramount.

2.5 Sandboxing:

iOS, Android uses a sandboxing mechanism to restrict each app's access to the device's resources and limit the amount of data users can share between apps. It helps prevent malicious apps from accessing user data or executing unwanted actions on the device.

Security Benefits of Sandboxing & Permissions

1- Prevents Malware Spread : Isolates malicious apps from critical system components.
2- Protects User Data : Limits access to personal information like contacts, messages, and location.
3- Enhances Privacy : Users can control permissions dynamically.
4- Reduces Attack Surface .

3. Mobile Application Security OWASP Mobile Top 10

3.1. The Importance Of Mobile Application Security

The importance of mobile application security is difficult to overestimate. If sensitive personal and corporate data is not sufficiently protected, bad actors may take action.

Consequences are often dramatic: the stolen data has a negative impact on the company's reputation, which leads to cost loss, and the stolen information might be used for fraudulent acts. It is important to implement the best security practices and take into account all possible risks, including malware attachments, API threats, malicious code injection, phishing attacks, and others.

3.2. The Key Mobile App Vulnerabilities And Threats

The OWASP Mobile Top 10 identifies the most critical vulnerabilities affecting mobile apps today. These include the following:

1. **Improper Credential Usage:** This vulnerability arises from poorly managed credentials, such as hardcoded passwords or storing sensitive information in plaintext. Such practices can lead to unauthorized access and potential data breaches.

2. **Inadequate Supply Chain Security:** When third-party libraries or components are integrated into apps without thorough security checks, it opens the door for attackers to exploit these weak points and insert malicious code.
3. **Insecure Authentication/Authorization:** Weak or flawed authentication processes can allow unauthorized users to gain access to restricted areas of an app, compromising sensitive information.

4. **Insufficient Input/Output Validation:** Apps that do not properly validate user inputs and outputs are vulnerable to various attacks, including data manipulation and code injection, which can undermine the app's functionality and security.

5. **Insecure Communication:** If an app transmits sensitive data over unencrypted channels or uses weak encryption, that data is susceptible to interception, leading to privacy breaches and potential exploitation.

6. **Inadequate Privacy Controls:** This risk arises when apps do not sufficiently protect user data, leading to unauthorized access and privacy violations, which can erode user trust.

7. **Insufficient Binary Protections:** The lack of measures to protect the app's binary code makes it easier for attackers to reverse engineer or tamper with the app, potentially leading to unauthorized modifications or data extraction.

8. **Security Misconfiguration:** Poorly configured security settings, such as default passwords or unnecessary features, can make an app more vulnerable to attacks.

9. **Insecure Data Storage:** Storing sensitive information in easily accessible locations on a device, or failing to encrypt data, leaves it exposed to potential breaches.

10. **Insufficient Cryptography:** Using weak or improperly implemented cryptographic techniques can make sensitive data more vulnerable to attacks, compromising the app's security and user privacy.

    Combating these threats requires using the best practices and standards to minimize the attack surface area.

3.3. Mobile Application Security Standards And Best Practices

1- Secure your code
2- Implement strong authentication
3- Protect data
4- Regularly update and patch
5- Conduct thorough security testing
6- Monitor for threats
7- Educate users