CompTIA

CompTIA PenTest+

Exam PT0-002

CompTIA PenTest+ Exam PT0-002

Lesson 11

Targeting Mobile Devices

Objectives

- Given a scenario, research attack vectors and perform wireless attacks.
- Explain common attacks and vulnerabilities against specialized systems.

Lesson 11

Topic 11A

Recognize Mobile Device Vulnerabilities

Comparing Deployment Methods

- Many companies adhere to a structured mobile device implementation model, which can include the following:
  - BYOD, COBO, COPE and CYOD
- Access control on a mobile device must be a top priority.
  - If a threat actor can bypass the security of a smartphone or tablet, they can potentially gain access to personal and corporate information.

Controlling Access

- Access control will prevent unauthorized users from accessing the device, which can be accomplished in many ways, that include:
  - **What you know**, such as a password, passphrase, or PIN
  - **What you have**, such as a smart card or Universal Serial Bus (USB) token
  - **What you are** (biometric), such as a fingerprint
  - **What you do**, such as a swipe pattern
  - **Where you are**, such as trusted physical or logical location
  - **What conditions are present** (context-aware), such as geolocation

Enterprise Mobility Management (EMM)

- A class of management software designed to apply security policies to mobile devices and apps in the enterprise as follows:
- **Mobile device management (MDM)**—sets device policies for authentication, feature use, and connectivity
- **Mobile application management (MAM)** - prevent unauthorized apps from being installed and automatically push out updates
- The core functionality of endpoint management suites extends the concept of network access control (NAC) solutions.
  - Configure policies that prevent data transfer to personal apps.

Setting Policies and Protecting Data

- EMM allows administrators to provide remote access to managed device that ensure the security of the infrastructure:
  - Restrict certain features and services based on access control policies
  - Enroll and authenticate devices
  - Push out OS, app, and firmware updates to devices
  - Locate devices through GPS and other technologies
  - Prevent root access or jailbreaking devices
  - Compartmentalize sensitive organization data

Vulnerable Characteristics of Android Devices

- Most threats targeted at mobile platforms affect Android devices, which represent the largest market share.
- Android devices can fall victim to attack for several reasons:
  - Using older OS versions with unpatched vulnerabilities
  - Customizing the operating system
  - Using third-party apps
- Many of the threats occur because users obtain apps from an unofficial source rather than from the Google Play store.

Protecting an Apple iPhone

- More secure as you can only obtain apps from the App Store.
- To circumvent this restriction, users jailbreak their phone
    - Removes the protective seal and any OS specific restrictions to give users greater control over the device.
    - Jailbreaking poses a significant threat. Once a device is jailbroken, any application can read and write to the root file system.
    - The OS will run unsigned code, which is normally prevented by Apple.

Recognizing Threats to Business Logic

- The business logic process represents the flow of information from requesting access to when the request hits a resource.
- Numerous threats and vulnerabilities can exist that can include:
    - Deperimeterization
    - Lost or stolen devices
    - Lack of antimalware protection
    - Using known vulnerable components
    - Passcode vulnerabilities

Review Activity: Recognize Mobile Device Vulnerabilities

- Review approaches to mobile device implementation
- Discuss methods used to control access
- Explain some of the key elements of EMM
- Compare Android and iPhone security considerations
- List some of the threats to the business logic process

Lesson 11

Topic 11B

Launch Attacks on Mobile Devices

Attacking a Mobile Device

- Threats and vulnerabilities to mobile devices include:
  - **Malware**: Spyware, Trojans, Rootkits, Viruses and Worms
  - **Biometric integration** - a poorly designed device might allow a malicious actor to spoof the system by presenting a forged biometric.
  - **Execution of activities using root**, which can occur when the user roots or jailbreaks their system to improve the performance of the device.
  - **Over-reach of permissions** can occur, as it’s often up to the individual to decide what services to access when downloading and installing an app.

Use Social Engineering

- In addition to phishing, pharming, and baiting, malicious actors use other techniques that target mobile devices, such as:
  - **Vishing** is phishing using Voice over Internet Protocol (VoIP).
  - **SMiShing** uses text messages to entice users to click on a link
  - **Drive by downloads** can occur while browsing the internet
  - **Spamming** is sending unsolicited ads and calls to a mobile user
  - **Browser Hijackers** take a web request and send it to another search engine

Moving through the Attack

- When using social engineering, the attack phase will generally complete the following steps:

1. Research some type of ploy that will get the victim to click on a link or complete some action.

2. Engage the victim by leveraging the ploy, possibly sending as a phishing or SMiShing attack with the hopes that the victim will complete some action.

3. Once the victim responds, extract sensitive information, such as login credentials on a vendor account.

4. After the attack is over or has played out its useful life, remove all traces of the attack, such as any bogus ecommerce sites.

Spyware on a Mobile Device

- Spyware can pose a serious problem as they are designed to track your usage and capture passwords and data.
- Can get spyware on a phone by someone having physical contact with the device, or by sending a message with a link
  - Once installed spyware can monitor all activity such as text messages, social media posts, and phone calls, along with websites that were visited.

Bluejacking and Bluesnarfing a Signal

- **Bluejacking** is used by attackers to send out unwanted messages, images, or videos to a device using a Bluetooth connection.
  - Performed by sending a message to nearby, discoverable devices using the attacker's Bluetooth app.
  - Can be used as a vector to carry out more insidious attacks.
- **Bluesnarfing** is more aggressive attack, as a malicious actor can read information from a victim's Bluetooth device.
  - The end goal is to glean sensitive data from the victim, like their contacts, calendars, email messages, text messages, etc.

Dealing with Malware

- iOS devices are more restrictive can only install apps from the official App Store, which helps prevent malware.
  - Jailbreaking an iPhone enables devices to install apps from third-party sources, which might contain malware.
- Android OS is much less restrictive. A single setting can make it possible for the device to install apps from third-party sources.
  - Rooting process reduces the device's security, as once rooted, apps will be able to run outside of their sandbox environments

Analyzing Malware

- Provides a way to see what happens when a virus executes
  - Identifies vulnerabilities, which helps prevent future attacks.
- **Reverse engineering** steps through the code to see what happens when the code is run on a device.
- **Sandbox analysis** uses virtualization to provide a safe environment to analyze malware.

Review Activity: Launch Attacks on Mobile Devices

- Outline some of the threats designed to target a mobile device
- Review Social Engineering techniques used on a mobile device
- Discus the steps taken when launching a social engineering attack
- Describe why spyware can be a serious attack
- Compare Bluejacking and Bluesnarfing
- Review how iOS and Android devices deal with malware
- Explain ways to analyze malware

Lesson 11

Topic 11C

Outline Assessment Tools for Mobile Devices

Recognizing the Testing Life Cycle

- When dealing with mobile devices, most organizations incorporate a testing framework to provide oversight and minimize risk:
  - **Mobile Device Assessment**—provides an overview of compliance and business logic issues.
  - **BYOD Approval**—selects appropriate devices and creating policies.
  - **Secure App Development**—creates organization specific apps in-line with organizational policy.
  - **Mobile APP Testing**—includes Static Application Security Testing (SAST) and Dynamic Application Security Testing (DAST).

Using Kali Linux

- A suite of tools designed to conduct penetration testing on a variety of devices. Includes applications such as:
  - **Ettercap** is a suite of tools that can be used to launch various types of Man in The Middle (or on-path) attacks.
  - **Android SDK tools** have packages so you can design, build, and test mobile apps for Android devices along with reverse engineering an existing device.
  - **Burp Suite** is an integrated platform for testing web applications along with a mobile assistant designed to test iOS devices.

Mobile Security Framework (MobSF)

- The MobSF can provide an automated evaluation of code and malware analysis using both static and dynamic analysis as follows:
  - **Static analysis** can evaluate both Android and iOS.
  - **Dynamic analysis** can assess an Android platform.
- The framework conducts an assessment to determine:
  - OS reputation, whether it has been rooted or jail broken, and app security.

Mobile Security Testing Guide

- The MSTG provides an intuitive framework that steps you through the assessment process. Key elements include:
    - A dashboard to summarize testing information along with contact information
    - Security recommendations for both Android and iOS devices
    - Specifications for testing resiliency against reverse engineering and tampering
- In addition to providing extensive checklists, you'll also find hyperlinks for external resources.

Examining Code with Using Frida and Objection

- **Frida** work with a wide range of OS and includes custom developer tools used during application testing. Features include:
  - Examine plaintext data, dump process memory, in-process fuzzing
  - Anti-jailbreak (or root) detection, change a program’s behavior
- **Objection** is a scriptable debugger that allows you to perform various security related tasks on unencrypted iOS applications.
  - The team can run custom Frida scripts and interact with the filesystems on non-jailbroken iOS devices.

Debugging Applications

- **Drozer** is an attack framework that allows you to find security flaws on Android devices and apps
  - Works as a client-server model and lets you assume the role of an app
- An APK file is an app designed to run on an Android device. Two application decompilers that work with APK files are:
  - **APKX tool** an Android APK decompiler that allows you to pull and analyze the Java source code to see what's going on inside.
  - **APK Studio** an IDE designed so you can decompile and or edit an APK file.

Evaluating with Postman

- An API is a set of commands that is used to send and receive data between systems, such as a client and a server.
- Postman is an API testing tool that provides an interactive GUI environment used to interact and test an HTTP API.
- Postman is rich with features and can do the following:
  - Explore and create an API, build and run a test suite.
  - Work with other team members, Analyze results and run reports.
  - Integrate within the DevOps life cycle.

Review Activity: Outline Assessment Tools for Mobile Devices

- Discuss some of the testing frameworks used to minimize risk
- Review some of the tools included in Kali Linux
- Outline the benefits of the Mobile Security Framework
- List key elements of the Mobile Security Testing Guide
- Compare Frida and Objection when examining code
- Describe ways Drozer, APKX tool. and APK Studio to examine code
- Explain how Postman interacts and tests an HTTP API.

Lesson 11

Summary