Let

tutorialspoint

SIMPLY EASY LEARNING

www.tutorialspoint.com

https://www.facebook.com/tutorialspointindia https://twitter.com/tutorialspoint

About the Tutorial

Mobile security is a concept that has gained a lot of importance ever since the launch of the first mobile OS, Symbian, which was launched by Nokia. It is continuing to gain significance with the massive use of Android OS.

This tutorial will take you through the simple and practical approaches to implement mobile security techniques.

Audience

This tutorial has been prepared for beginners to IT administrators to help them understand the basic-to-advanced concepts related to mobile security that they can use in daily life and in their organizations.

Prerequisites

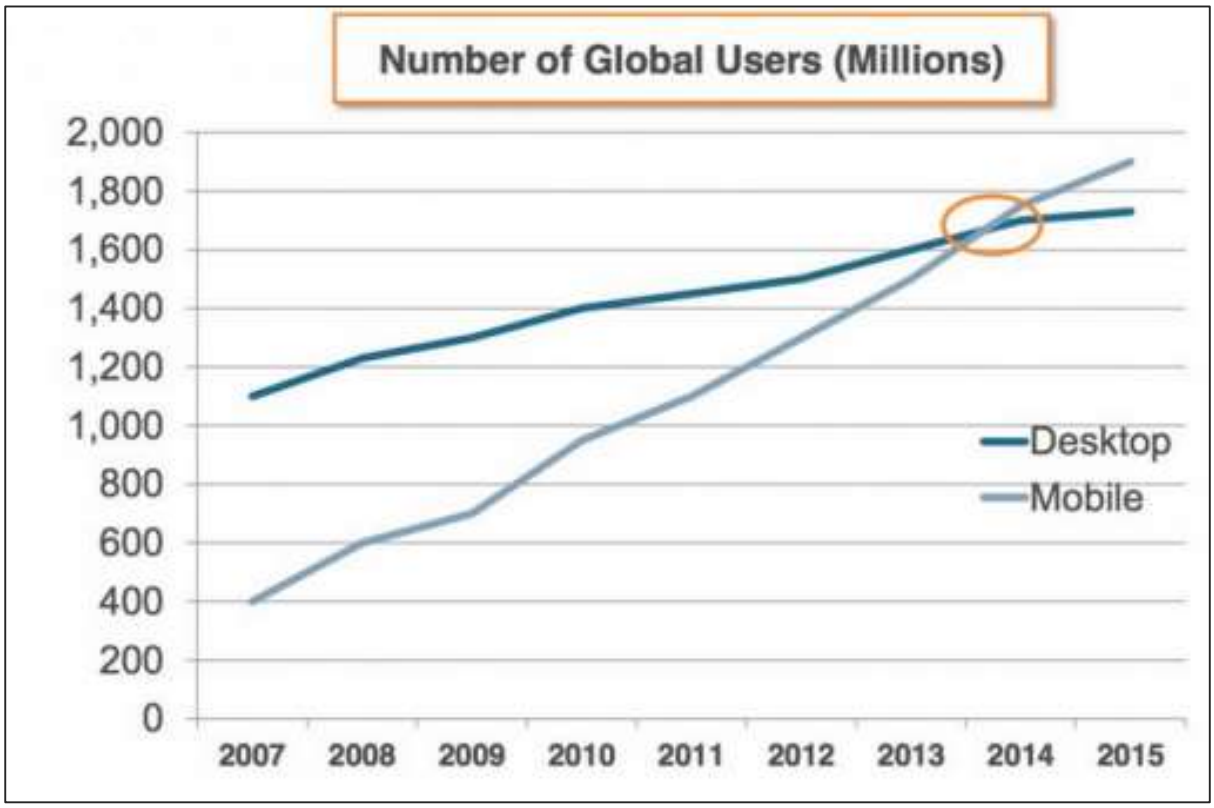
This is a very basic tutorial that should be useful for most users. Before you start practicing the various types of security options given in this tutorial, we assume that you are well-aware of the various features available in a standard smartphone.

Disclaimer & Copyright

Table of Contents

1. Mobile Security – Introduction

In this tutorial, we will deal with mobile security concepts mostly from the practical point of view. Take a look at the following graph, it illustrates the ever-growing number of mobile phone users across the world, which brings out the importance of mobile security.



The estimated number of mobile devices is around 5.8 billion, which is thought to have grown exponentially within five years and is supposed to reach nearly 12 billion within four years. Hence, it will be an average of two mobile devices per person on the planet. This makes us fully dependent on mobile devices with our sensitive data being transported all over. As a result, mobile security is one of the most important concepts to take in consideration.

Mobile Security as a concept deals with the protection of our mobile devices from possible attacks by other mobile devices, or the wireless environment that the device is connected to.

Following are the major threats regarding mobile security:

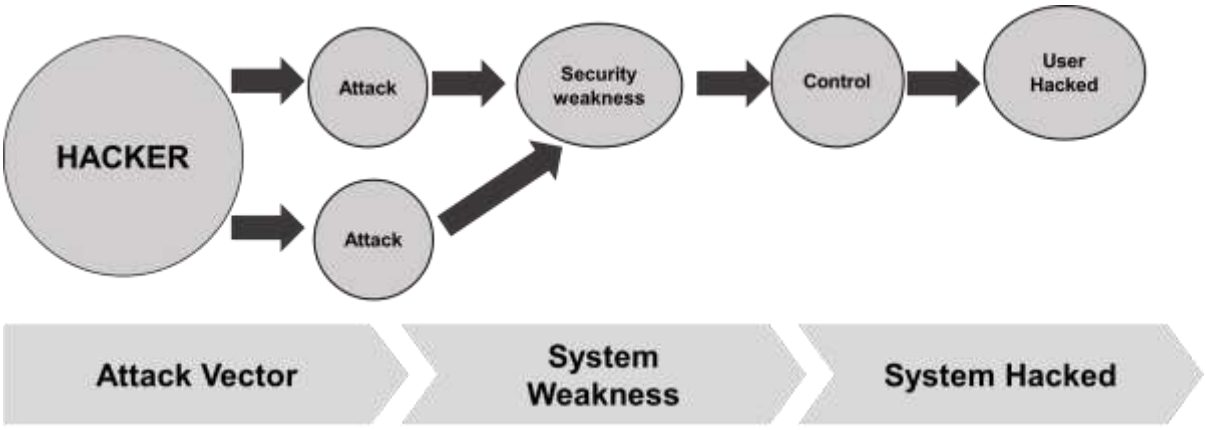
- Loss of mobile device. This is a common issue that can put at risk not only you but even your contacts by possible phishing.

- Application hacking or breaching. This is the second most important issue. Many of us have downloaded and installed phone applications. Some of them request extra access or privileges such as access to your location, contact, browsing history for marketing purposes, but on the other hand, the site provides access to other contacts too. Other factors of concern are Trojans, viruses, etc.

- Smartphone theft is a common problem for owners of highly coveted smartphones such as iPhone or Android devices. The danger of corporate data, such as account credentials and access to email falling into the hands of a tech thief is a threat.

2. Mobile Security – Attack Vectors

By definition, an **Attack Vector** is a method or technique that a hacker uses to gain access to another computing device or network in order to inject a "bad code" often called **payload**. This vector helps hackers to exploit system vulnerabilities. Many of these attack vectors take advantage of the human element as it is the weakest point of this system. Following is the schematic representation of the attack vectors process which can be many at the same time used by a hacker.



Some of the mobile attack vectors are:

- Malware
  - Virus and Rootkit
  - Application modification
  - OS modification
- Data Exfiltration
  - Data leaves the organization
  - Print screen
  - Copy to USB and backup loss
- Data Tampering
  - Modification by another application
  - Undetected tamper attempts
  - Jail-broken devices
- Data Loss
  - Device loss
  - Unauthorized device access
  - Application vulnerabilities

Consequences of Attack Vectors

Attack vectors is the hacking process as explained and it is successful, following is the impact on your mobile devices.

- **Losing your data**: If your mobile device has been hacked, or a virus introduced, then all your stored data is lost and taken by the attacker.

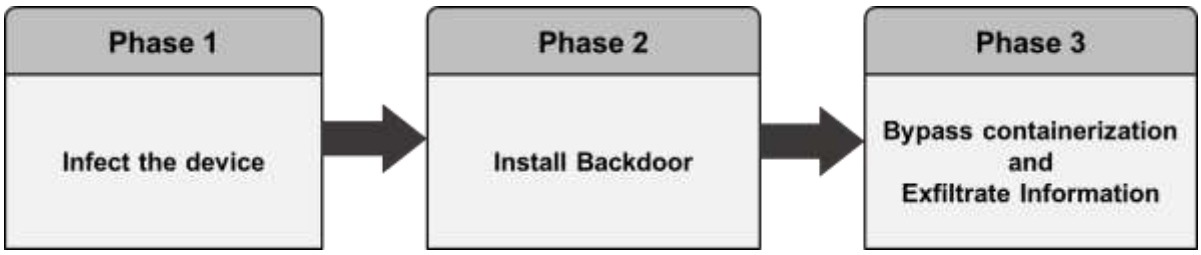
- **Bad use of your mobile resources**: Which means that your network or mobile device can go in overload so you are unable to access your genuine services. In worse scenarios, to be used by the hacker to attach another machine or network.

- **Reputation loss**: In case your Facebook account or business email account is hacked, the hacker can send fake messages to your friends, business partners and other contacts. This might damage your reputation.

- **Identity theft**: There can be a case of identity theft such as photo, name, address, credit card, etc. and the same can be used for a crime.

Anatomy of a Mobile Attack

Following is a schematic representation of the anatomy of a mobile attack. It starts with the infection phase which includes attack vectors.



Infecting the device

Infecting the device with mobile spyware is performed differently for Android and iOS devices.

Android: Users are tricked to download an app from the market or from a third-party application generally by using social engineering attack. Remote infection can also be performed through a Man-in-the-Middle (MitM) attack, where an active adversary intercepts the user's mobile communications to inject the malware.

iOS: iOS infection requires physical access to the mobile. Infecting the device can also be through exploiting a zero-day such as the JailbreakME exploit.

Installing a backdoor

To install a backdoor requires administrator privileges by rooting Android devices and jailbreaking Apple devices. Despite device manufacturers placing rooting/jailbreaking detection mechanisms, mobile spyware easily bypasses them:

Android: Rooting detection mechanisms do not apply to intentional rooting.

iOS: The jailbreaking "community" is vociferous and motivated.

Bypassing encryption mechanisms and exfiltrating information

Spyware sends mobile content such as encrypted emails and messages to the attacker servers in plain text. The spyware does not directly attack the secure container. It grabs the data at the point where the user pulls up data from the secure container in order to read it. At that stage, when the content is decrypted for the user's usage, the spyware takes controls of the content and sends it on.

How Can a Hacker Profit from a Successfully Compromised Mobile?

In most cases most of us think what can we possibly lose in case our mobile is hacked. The answer is simple - we will lose our privacy. Our device will become a surveillance system for the hacker to observer us. Other activities of profit for the hacker is to take our sensitive data, make payments, carry out illegal activities like **DDoS attacks**. Following is a schematic representation.

OWASP Mobile Top 10 Risks

When talking about mobile security, we base the vulnerability types on OWASP which is a not-for-profit charitable organization in the United States, established on April 21. OWASP is an international organization and the OWASP Foundation supports OWASP efforts around the world.

For mobile devices, OWASP has **10 vulnerability classifications**.

M1-Improper Platform Usage

This category covers the misuse of a platform feature or the failure to use platform security controls. It might include Android intents, platform permissions, misuse of TouchID, the

Keychain, or some other security control that is part of the mobile operating system. There are several ways that mobile apps can experience this risk.

M2-Insecure Data

This new category is a combination of M2 and M4 from Mobile Top Ten 2014. This covers insecure data storage and unintended data leakage.

M3-Insecure Communication

This covers poor handshaking, incorrect SSL versions, weak negotiation, clear text communication of sensitive assets, etc.

M4-Insecure Authentication

This category captures the notions of authenticating the end user or bad session management. This includes:

- Failing to identify the user at all when that should be required
- Failure to maintain the user's identity when it is required
- Weaknesses in session management

M5-Insuficient Cryptography

The code applies cryptography to a sensitive information asset. However, the cryptography is insufficient in some way. Note that anything and everything related to TLS or SSL goes in M3. Also, if the app fails to use cryptography at all when it should, that probably belongs in M2. This category is for issues where cryptography was attempted, but it wasn't done correctly.

M6-Insecure Authorization

This is a category to capture any failures in authorization (e.g., authorization decisions in the client side, forced browsing, etc.) It is distinct from authentication issues (e.g., device enrolment, user identification, etc.)

If the app does not authenticate the users at all in a situation where it should (e.g., granting anonymous access to some resource or service when authenticated and authorized access is required), then that is an authentication failure not an authorization failure.

M7-Client Code Quality

This was the "Security Decisions Via Untrusted Inputs", one of our lesser-used categories. This would be the catch-all for code-level implementation problems in the mobile client. That's distinct from the server-side coding mistakes. This would capture things like buffer overflows, format string vulnerabilities, and various other code-level mistakes where the solution is to rewrite some code that's running on the mobile device.

M8-Code Tampering

This category covers binary patching, local resource modification, method hooking, method swizzling, and dynamic memory modification.

Once the application is delivered to the mobile device, the code and data resources are resident there. An attacker can either directly modify the code, change the contents of memory dynamically, change or replace the system APIs that the application uses, or modify the application's data and resources. This can provide the attacker a direct method of subverting the intended use of the software for personal or monetary gain.

M9-Reverse Engineering

This category includes analysis of the final core binary to determine its source code, libraries, algorithms, and other assets. Software such as IDA Pro, Hopper, otool, and other binary inspection tools give the attacker insight into the inner workings of the application. This may be used to exploit other nascent vulnerabilities in the application, as well as revealing information about back-end servers, cryptographic constants and ciphers, and intellectual property.

M10-Extraneous Functionality

Often, developers include hidden backdoor functionality or other internal development security controls that are not intended to be released into a production environment. For example, a developer may accidentally include a password as a comment in a hybrid app. Another example includes disabling of 2-factor authentication during testing.