Lookout | Mobile Threat Vectors 101

App-based threats

Malware can be installed from a number of sources including being sideloaded from the web, via infected websites, in-app ads, push notification ads, emails & SMS messages such as fake system updates, and pirated versions of legitimate apps.

Non-compliant apps, or those apps that may violate a company's policies based on the data it accesses or collects, can also pose a threat. Vulnerabilities within apps can also be exploited to collect data.

Device-based threats

iOS and Android, like their PC counterparts, contain vulnerabilities in their operating systems that can be exploited locally or remotely. If exploitation of an OS vulnerability leads to root or kernel-level privilege escalation, an attacker can then compromise any application on the device, including those that encrypt data at rest, such as enterprise containers.

A note: Not all jailbreaking/rooting is malicious. Sometimes users opt to jailbreak or root their device in order to gain deeper control over the system.

Network-based threats

Attackers can use malware or socially engineer users to configure a device to route all network traffic through a malicious proxy or VPN connection. Active man-in-the-middle attacks can be used to exploit OS or app vulnerabilities. Man-in-the-middle attacks can also leverage these types of vulnerabilities to steal data.

POTENTIAL DAMAGE

When installed on a device, malware can use multiple techniques to cause damage:

Abuse of legitimate APIs to steal data, monitor the device's sensors, access protected Wi-Fi/VPN networks, or perform other malicious actions. OS vulnerability exploitation to gain full access to the device.

If an enterprise includes an app as part of its product offering, malicious applications could also compromise brand reputation. (Marchcaban, for example, places an invisible overlay on top of Paypal's application to steal user data).

Depending on the data collected, non-compliant apps could violate an enterprise's compliance obligations or otherwise jeopardize sensitive information. App vulnerabilities can be exploited to cause similar damage.

EXAMPLES: Mobile malware, such as Not-Compatible, Malapp.d, and many others.

Rooting and jailbreaking breaks the trust model of a device, exposing enterprise containers and other apps to data theft from any app that runs under elevated privileges.

EXAMPLES: OS vulnerabilities such as "Stagefright" let attackers exploit the native Android media player to remotely steal data.

While most iOS and Android mobile apps and websites use SSL to encrypt data in motion, an attacker who is able to become a man-in-the-middle can use multiple techniques to exploit improper SSL configurations and decrypt and steal data.

EXAMPLES: Fake Root CA attacks occur when an attacker introduces a malicious certificate authority into the trusted root certificate authority store of the victim device. sslstrip is a tactic that effectively strips out the "S" in HTTPS connections, allowing normally encrypted data to be viewed in plaintext. TLS Protocol Downgrade occurs when an attacker manipulates the negotiated connection to downgrade the protocol or cipher suites.

SOLUTIONS

A technology that detects mobile threats using signature-, behavioral-, and machine learning technology. This solution will alert both users and IT admins and provide education on the threat, as well as remediation options.

The technology should be powered by an encompassing dataset of the world's mobile code as obtained from millions of devices that report device state and application data into that data set. The technology should run in the cloud, automatically scaling with the size of the dataset to ensure that it is powerful, but lightweight on the end-user device.

A technology or service that can detect vulnerabilities in apps and OSes. The solution may also be able to test the environment in which a corporate app lives to check for existing vulnerabilities or vulnerable apps. It should then provide education and remediation options.

The technology should be powered by a large, crowdsourced dataset of mobile code as obtained from millions of devices that report device state and application data into that data set. The technology should run in the cloud, automatically scaling with the size of the dataset to ensure that it is powerful, but lightweight on the end-user device.

A technology that automatically detects device connections to various networks and either tests those networks to determine if they are secure or detects attacks in real time. The solution should alert users and IT admins to the presence of a threat, and offer remediation options.