



WHITEPAPER

# Managing Jailbreak Threats on iOS

Enterprise Mobile Security

## I. Introduction

Apple strongly discourages iOS users from jailbreaking their devices and for good reason: the practice can impair key device functionality, such as OS upgrades, and can also introduce significant security risks, such as the ability to download and execute unvetted and potentially malicious applications.

Nonetheless, many people ignore these risks and jailbreak their devices to unlock capabilities or content that Apple otherwise prohibits on iOS. A range of free tools and tutorials have made jailbreaking widely accessible, even to non-technical users. Consequently, an estimated 7.5% of all iPhones, more than 30 million devices worldwide, are jailbroken. Jailbreaking is especially prevalent in China where an estimated 13% of all iPhones are jailbroken<sup>1</sup>.

Jailbreaking entails removing restrictions on iOS by modifying system kernels to allow read and write access to the file system. This form of administrative privilege escalation permits custom software installation and device behaviour modification.

While Apple diligently patches new jailbreak vulnerabilities, it's a constant game of Whack-A-Mole with the jailbreaking community racing to find new vulnerabilities whenever Apple releases iOS upgrades. A well-known jailbreak developer, for example, recently jailbroke a beta version of one of the latest iOS versions (8.4) within six days of its release<sup>2</sup>.

Today most jailbreak-detection methods rely on client-side tests that the jailbreaking community has unfortunately managed to reverse-engineer and evade.

This whitepaper provides an overview of the jailbreaking process and its security risks and highlights Lookout's unique and innovative approach to managing this security threat on iOS devices.

Jailbreakers can easily avoid detection using free tools such as xCon<sup>3</sup> or FLEX<sup>4</sup> that fool standard jailbreak detection tests.

## II. The Path to Jailbreaking

### Motivation

A wide variety of Apple-restricted iOS customisations and content might incentivise a user to jailbreak their device, such as:

- Enabling app-multitasking with split-screen views
- Locking individual apps with Touch ID
- Creating mobile wifi hotspots without paying additional carrier charges
- Unlocking the phone to use the device internationally or with other carriers
- Installing surveillanceware on a partner's device to track their communications or location
- Consuming pirated music or video
- Downloading apps from third party app sources, such as Cydia or Lima
- Downloading pirated apps from app repositories, such as Hackulous

<sup>1</sup> "WireLurker" Malware May Have Infected 100,000+ iPhones, No Jailbreak Required". DailyTech. November 2014. <http://www.dailytech.com/WireLurker+Malware+May+Have+Infected+100000+iPhones+No+Jailbreak+Required/article36850.htm>

<sup>2</sup> "Android Wear Update Takes Aim at Apple Watch". GottaBeMobile. April 2015. <http://www.gottabemobile.com/2015/04/22/android-wear-update-takes-aim-at-apple-watch/>

<sup>3</sup> xCon entry on TheiPhoneWiki.com: <https://theiphonewiki.com/wiki/XCon>

<sup>4</sup> Forum webpage link: <http://www.sinfuliphone.com/showthread.php?t=10032183>

## Process

All jailbreaks exploit iOS vulnerabilities to either bypass, disable or patch the signature checks that run when an iOS device boots to ensure it loads only Apple-approved software. One of the most popular sources of non-Apple-approved software is Cydia, an app that helps users find and download software for their jailbroken iOS devices. While the technical mechanisms behind each jailbreak technique remain complex, the end-user experience today is relatively simple due to tools that largely automate the jailbreaking process. Popular jailbreaking tools include:

Absinthe	JailbreakMe	sn0wbreeze
blackra1n	limera1n	Spirit
Corona	Pangu	TaiG
evasi0n	PwnageTool	
greenpois0n	redsn0w	

Given their dependence on specific iOS vulnerabilities, each jailbreaking tool typically works on only a limited range of iOS versions and devices. The evasi0n tool, for example, which helped jailbreak over 7 million iOS devices in 2013<sup>5</sup>, works for iOS 6.0–6.1.2 and 7.0–7.0.6 (using evasi0n7). More recent tools include TaiG, used to jailbreak iOS 8.0–8.1.2, and Pangu, whose latest version, Pangu8, can jailbreak iOS 8.0–8.1.

In the early days of iOS, many jailbreak techniques could not survive device reboots. These so-called “tethered jailbreaks” required users to connect devices to a computer if they wanted to reboot it in a jailbroken state. Most modern jailbreak tools (e.g. evasi0n, TaiG and Pangu) are “untethered”, however, and allow devices to independently reboot in a jailbroken state.

The automation provided by these tools has put jailbreaking within the technical reach of most iOS users and reduced jailbreak processing time to a matter of minutes. Most modern jailbreak tools require users to take only the following steps:

1. Manually verify the iOS version (via: Settings > General > About > Version)

2. Back up device data (an optional precaution)
3. Manually enable or disable a few basic device settings (e.g. “turn off device passcode”)
4. Download the jailbreak software to a computer
5. Connect the device to a computer via USB, open the jailbreak app, and run it
6. Wait for the device to automatically reboot

## Jailbreak Detection Evasion

The enterprise security risks posed by jailbreaking compound in the face of tools (e.g. xCon and FLEX) that can help users easily evade common jailbreak detection methods. A user can download xCon, for example, directly from the third-party app store Cydia.

Mobile device management (MDM) services provide jailbreak detection, as do many media and financial service apps that want to limit content pirating and account compromise, respectively. Unfortunately, these jailbreak detections rely on a combination of relatively straightforward and evadable tests, such as:

- Checking for files or directories common to jailbroken devices, such as Cydia
- Checking for elevated directory permissions (i.e. more directories with “write” permission)
- Checking to see if an app can successfully write files outside of its sandbox

The fundamental limitation with these and comparable detection tests is that as client-side tests they can be accessed, reverse-engineered, and evaded by attackers. In addition, the apps performing these jailbreak detection tests (e.g. the MDM app) must go through Apple’s app review process, limiting the scope of data they can collect to analyse a device’s jailbreak status.

<sup>5</sup> “Evasi0n Is The Most Popular Jailbreak Ever: Nearly Seven Million iOS Devices Hacked In Four Days”. Forbes. February 2013.  
<http://www.forbes.com/sites/andygreenberg/2013/02/08/evasi0n-is-the-most-popular-jailbreak-ever-nearly-seven-million-ios-deviceshacked-in-four-days>

### III. The Jailbreaking Threat

#### Apps on Jailbroken Devices

Jailbroken devices create a major enterprise risk given their ability to run apps developed outside of Apple's review, which may be malicious or contain vulnerabilities. Jailbreaking removes the normal signing certificate checks that prevent these apps from executing and gives them unrestricted access to the device, including the ability to use undocumented APIs that Apple otherwise prohibits. These private APIs can empower apps with a wide range of dangerous capabilities on jailbroken devices, such as the ability to install or launch additional code or collect location data without notification.

When attackers target jailbroken iOS devices with malware they often distribute these threats through third-party app marketplaces and software repositories that have minimal to non-existent app vetting policies. Surveillanceware, a type of malware that conducts comprehensive data collection on compromised devices, represents one of the greatest app-based threats to jailbroken iOS devices. Most documented iOS surveillanceware to date has specifically targeted jailbroken devices, including the recently discovered XAgent threat<sup>6</sup>.

#### OS Vulnerabilities

Jailbroken devices can also introduce enterprise security risk by creating new OS vulnerabilities that attackers can exploit, such as:

- Escalated admin privileges provided by jailbreaking are an open door that can also be exploited by attackers to insert or extract files from the file system, as happened with the Xsfer mRAT trojan.
- Some jailbreaking methods leave SSH enabled with a well-known default password (e.g. alpine) that attackers can use for Command & Control purposes<sup>7</sup>.
- Apps on a jailbroken device can run with escalated privileges and can access sensitive data belonging to other apps, enabling widespread device surveillance by an attacker who could steal credentials by installing a keystroke logger on the device.

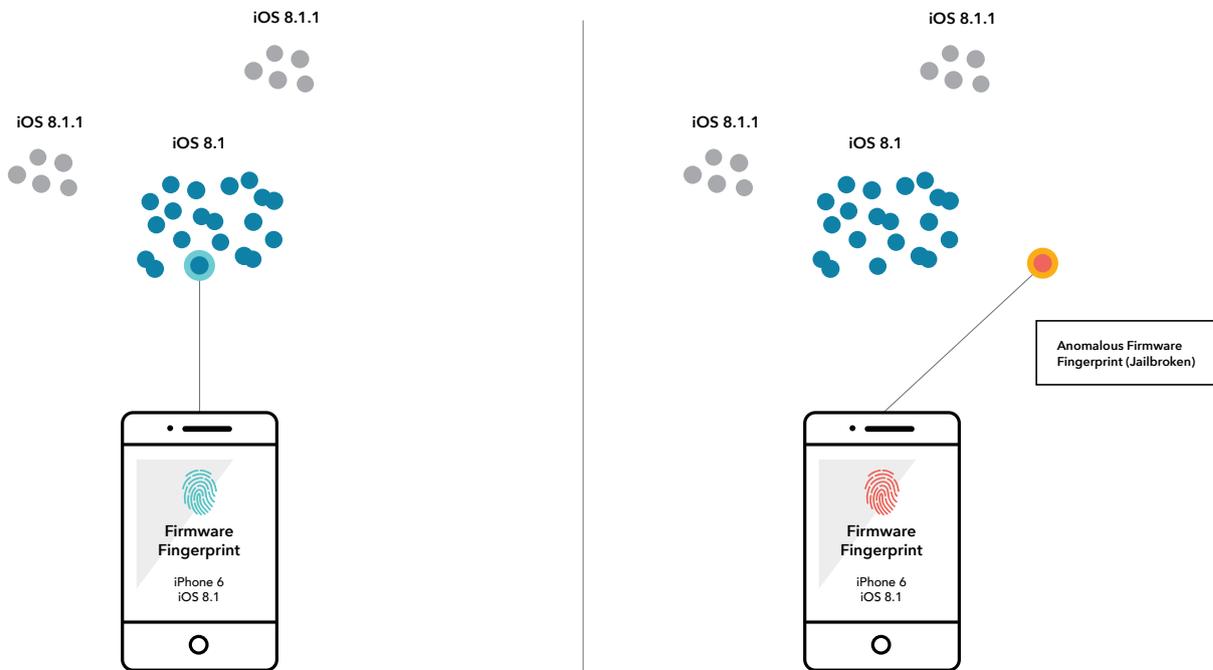
<sup>6</sup> "iOS spyware steals texts, photos, contacts, switches on voice recorder". ZDNet. February 2015.

<http://www.zdnet.com/article/ios-spyware-steals-texts-photos-contacts-switches-on-voice-recorder/>

<sup>7</sup> "First iPhone worm discovered - ikee changes wallpaper to Rick Astley photo". Naked Security Blog by Sophos. November 2009.

<https://nakedsecurity.sophos.com/2009/11/08/iphone-worm-discovered-wallpaper-rick-astley-photo/>

#### IV. Lookout's Approach to Jailbreak Protection



To better detect compromised operating systems on iOS devices, the Lookout Security Platform collects a range of device security telemetry to form a digital firmware fingerprint of each device. This security telemetry includes a range of OS file metadata, such as file size, and also OS configuration data, such as build properties.

After collecting this data the platform then reassembles it in the cloud to form a device OS fingerprint. It correlates the various data points of this fingerprint against Lookout's mobile intelligence dataset to identify when a device is vulnerable or has been compromised through jailbreak, predicting device risk based on anomalies or correlations to known signals of compromise.

Through this unique approach Lookout can offer more comprehensive jailbreak detection thanks to two key differentiators: first, Lookout distributes its iOS security app using enterprise provisioning, enabling Lookout's iOS app to analyse a much wider range of security telemetry to assess a device's jailbreak status. Second, Lookout analyses this security telemetry in the cloud, which makes it substantially more difficult for attackers to reverse-engineer and evade with tools like xCon as it would require them to mimic every single security signal of a legitimate device, as opposed to avoiding a specific client-side test.