LACOON MOBILE SECURITY

Secure your organization from all mobile threats

Practical Attacks against Mobile Device Management (MDM) Solutions

Michael Shaulov, CEO and co-founder, Lacoon Mobile Security
michael@lacoon.com

Daniel Brodie, Sr. Security Researcher, Lacoon Mobile Security
danielb@lacoon.com

Introduction

Mobile Device Management (MDM) solutions are perceived to be the ultimate solution for mobile security in the enterprise. According to Gartner Inc's October 2012 report: "Over the next five years, 65% of enterprises will adopt a mobile device management (MDM) solution for their corporate liable users".

But do MDM solutions really provide the security that corporations are looking for?

In this whitepaper, we show how spyphones - surveillance tools surreptitiously planted on a user's handheld device – are able to circumvent common MDM security offerings, such as secure containers.

A Short Primer to MDMs and Secure Containers

Mobile Device Management (MDM)

As their names imply, MDMs are mobile policy and configuration management tools. With the rise of consumer-owned and –enabled mobile devices in the enterprise (aka BYOD), organizations have recognized the challenge of establishing and enforcing a standard policy to help them manage the influx of these devices. MDM addresses these needs by providing management across four different layers:

- **Software management.** Manages mobile applications, content and operating systems, including:
  - Provisioning and configuration
  - Updates, patches and fixes
  - Authorized software monitoring
  - Backup/restore procedures
- **Network service management.** Gains network-device information such as location, usage and cellular/ WiFi, in order to support:
  - Provisioning
  - Billing
  - Help desk/ support
- **Hardware management**. Manages the physical device components, including:
  - Provisioning
  - Inventory
  - Activation/ Deactivation

  - Performance
- **Security management**. Enforcement of security policies, including:
  - Remote wipe
  - Remote lock
  - Secure configuration enforcement
  - Encryption

Secure Containers

Secure containers separate between business and personal data on the mobile and prevent business critical data from leaking out to unauthorized individuals. This is done by encrypting the data on the phone and providing additional data security features, such as copy-paste DLP. A common scenario for secure containers is to enable companies to perform a "remote-wipe" only on an ex-employee's business data, rather than removing all mobile data. Thus relieving the anguish (and possibly, also the legal ramifications) of deleting the employee's personal photographs as well.

Popular MDM tools offered with the additional layer of a secure container include: MobileIron, AirWatch, FiberLink, Zenprise and Good Technologies.

How do secure containers work?

The secure container runs in the mobile's OS supplied sandbox, where the separation between business and personal data is implemented through encryption. All business data in the container is encrypted. In addition, all communications with enterprise assets such as the Exchange Server and cloud-based corporate apps, are performed under SSL encryption.

In particular, for iOS, Apple provides additional APIs for MDM solutions which are unavailable to regular iOS apps. These may be used to retrieve information and manage policies. However, the MDM solutions are still restricted in their enforcement capabilities.

The Mobile Threatscape

Looking at the mobile threat landscape, there are two separate categories of malicious mobile applications:

1. **Mass Mobile Malicious Apps.** These are consumer-oriented malicious applications with the obvious financial motivation. Examples of such malicious apps include apps that monetize on premium text, dialers, SMS spammers, and mobile banking trojans.
   These types of applications are not considered too sophisticated. Typically, the malware developer places the malicious tool on Google Play – or other third party application market – in hopes of reaching as many downloads as possible. Further, as a consumer-focused mass malware, a device infected with one of these apps does not have much impact on an organization.
2. **Targeted Mobile Attacks, aka Spyphones.** These are mobile surveillance software installed on particular individuals. Once installed, spyphones are privy to all data on the mobile, as well as to all communication passed on the device.

As opposed to the mass malware apps, spyphones are installed on a per-device basis. Accordingly, attackers invest heavily in discovering, creating and developing new techniques to install and hide spyphones on the user's device.
This type of malicious software is used to target the organization, with the goal of cyber-espionage. As such, the impact of such an attack on the organization is extremely high – from gaining access to corporate emails and exfiltrating memos discussing the company's roadmap, to recordings of confidential phone calls and board meetings.
Spyphones are not used only against high-end targets. Private individuals have been known too to be victims of spyphones - for example, in the case of cheating spouses.

Spyphone Capabilities

Most spyphones provide, at a minimum, the following capabilities which may prove to be costly to the business:

- **Eavesdropping and surround recording.** Examples: listening in real time on customer calls and recordings of board meetings.
- **Extracting call and text logs.** Examples: text messages which contain board meetings follow-ups and voice memos.
- **Tracking location.** Examples: tracking the location of executives at key accounts meetings.
- **Snooping on corporate emails and application data**. Examples: retrieving corporate emails regarding upcoming M&A activity.

The Range of Spyphones

Lacoon's Mobile Threat Intelligence (MoTI) arm identified more than 50 families of spyphones. These spyphones run the gamut from dedicated high-end groups targeting specific nations and corporations, to low-end software targeting the private consumers.

Some publicized examples of spyphones from the high-end of the spectrum include:

- FinSpy, by The Gamma Group (August 2012, March 2013) - http://bits.blogs.nytimes.com/2012/08/13/elusive-finspy-spyware-pops-up-in-10-countries/
- LuckyCat (July 2012) - http://www.darkreading.com/mobile-security/167901113/security/attacks-breaches/240004623/luckycat-apt-campaign-building-android-malware.html
- Red October's mobile component (January 2013) - https://threatpost.com/en_us/blogs/rocra-espionage-malware-campaign-uncovered-after-five-years-activity-011413
- Android-targeted malware against Tibetan activists (March 2013) http://www.forbes.com/sites/parmyolson/2013/03/26/first-known-targeted-malware-attack-on-android-phones-steals-contacts-and-text-messages/

At the lower end of the spectrum are mobile surveillance tools which most commonly portray themselves as promoting parental controls and spouse monitoring. The operators of these tools follow a SaaS business model where the exfiltrated data is stored and managed as a dedicated cloud service. Similarly

to a well-run business, the operators of these tools promise professional world-wide support. Their GUI is simple and user-friendly to enable all users – from the tech-savvy to the technologically impaired – to run their service.

The difference between the military and non-military grade mobile surveillance tools? The device infection vectors and accordingly, their cost. Current estimates hold nation-targeted spyphones at $350K[1]. In the meanwhile, the commoners-targeted tools follow a monthly low licensing model– sometimes as low as $4.99.
The amazing part is that the end-result is essentially the same on the targeted devices. So for just a bit more than the price of a Starbucks latte, an attacker can purchase a surveillance tool with nearly identical capabilities to that of a top-end spyphone.

Spyphones in the Wild

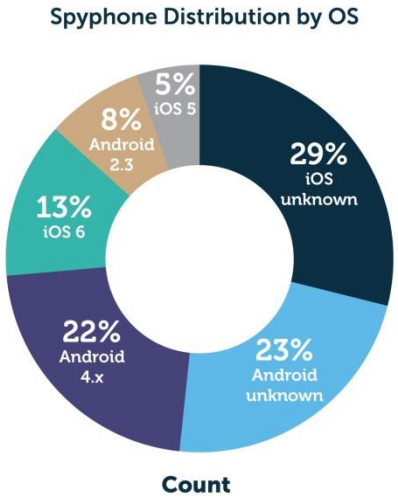
To paint a better picture of how common spyphones are in the wild, Lacoon Mobile Security partnered with global cellular network providers to sample 500,000 subscribers.

It is important to note that the samplings was done on a statistically diverse group of cellular network users and that there was nothing to suggest a higher usage of spyphones than the usual.

This type of monitoring provided real-time insights on the infection rates of the different devices. In addition, it allowed the content inspection of the communications to C&C servers of 17 different spyphone distributors and the analysis of the data that the attackers gathered from users' mobile devices.

Survey Findings

- **Infection Rates**.
  The first sampling showed that 1 of 800 devices had a spyphone installed.
- **Spyphone distribution by OS:**
  Our sampling showed that 47% of the infected devices were iOS-enabled and 53% were Android-based. The following figure details the distribution by device version.



[1] http://bits.blogs.nytimes.com/2012/08/13/elusive-finspy-spyware-pops-up-in-10-countries/

Myth-Busting the Security of Secure Containers

Secure containers may rely on different defense mechanisms to protect the corporate data:

- Detection of JailBreaking (iOS) and Root (Android) devices.
- Prevention of the installation of applications from third-party markets in order to protect against malware.
- Encryption of data
- The built-in Mobile OS Sandbox component

However, these measures can be easily bypassed:

- **There's a huge Internet community involved in JailBreaking/ Rooting efforts.** A quick Google[2] search will retrieve not only hacker-oriented details, but also step-by-step guidelines[3] for the layman on JailBreaking the device.
- **The JailBreaking/Rooting detection mechanisms are quite restricted.** Usually, checks are performed only against the features which signify a JailBroken/Rooted device. For example, it will check whether Cydia – an iOS app which allows the downloading of third party applications – is installed, or SU – the tool used by Android to allow privileged operations. More importantly, there are no detection mechanisms for exploitation. So even if the secure container recognizes a JailBroken/Rooted device, there are no techniques to detect the actual privilege escalation.
- **Android, for example, attempts to prevent malicious app installation.** However, these measures are placed with mass malware in mind. Furthermore, third party application restrictions should protect against malware. As a security mechanism, this has previously been proved to be defeated[4].

Behind the Scenes: Bypassing the Secure Container

In the following sections we present proof of concepts for bypassing the secure container – both for Android, and for iOS-based devices.

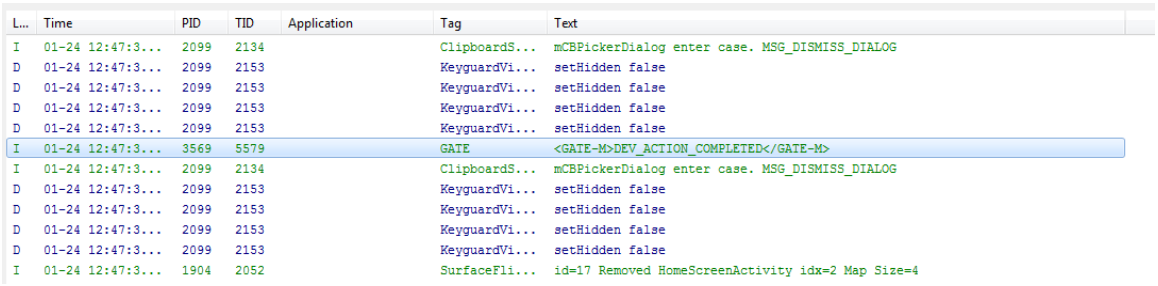
Android-based devices

A spyphone targeting Android-based devices can work in the following manner:

1. As demonstrated in BlackHat Vegas 2012, the attacker creates a "two-stage" application which bypasses the market's malicious app identification measures (e.g. Bouncer). By using the "two-stage", the attacker can publish a seemingly innocent application. Once the victim installs the app, the app refers to the malicious code which is then downloaded.
2. The app exploits a mobile OS vulnerability which allows for privilege escalation. For example, the recent vulnerability in the Exynos[5] chipset in the drivers used by the camera and multimedia devices.

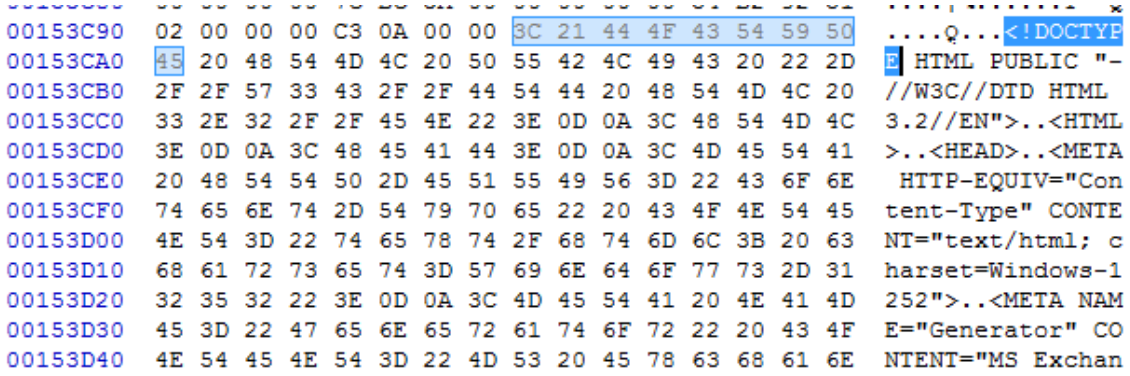
[2] http://www.cultofmac.com/177385/why-i-love-my-jailbroken-iphone/
[3] http://www.gizmag.com/how-jailbreak-ios-6-cydia-iphone-4-ipod-touch-4g/24552/
[4] Black Hat Vegas 2012: Adventures in BouncerLand - http://media.blackhat.com/bh-us-12/Briefings/Percoco/BH_US_12_Percoco_Adventures_in_Bouncerland_WP.pdf
[5] http://www.securityweek.com/samsung-patch-vulnerable-exynos-powered-devices

3. The spyphone creates a hidden 'suid' binary and uses it for privileged operations, such as reading the mobile logs (discussed in the next step). The file is placed in an execute-only directory (i.e. --x--x--x), which allows it to remain hidden from most root detectors.
4. The spyphone listens to events in the 'adb' logs. These logs, and their corresponding access permissions, differ between Android versions. For versions 2.3 or less, it's possible to simply use the logging permissions. For Android version 4.0 and higher, root permissions are required in order to view the logs.
5. The spyphone waits for a log event that signifies that the user is reading an email:



L...	Time	PID	TID	Application	Tag	Text
I	01-24 12:47:3...	2099	2134		ClipboardS...	mCBPickerDialog enter case. MSG_DISMISS_DIALOG
D	01-24 12:47:3...	2099	2153		KeyguardVi...	setHidden false
D	01-24 12:47:3...	2099	2153		KeyguardVi...	setHidden false
D	01-24 12:47:3...	2099	2153		KeyguardVi...	setHidden false
D	01-24 12:47:3...	2099	2153		KeyguardVi...	setHidden false
I	01-24 12:47:3...	3569	5579		GATE	<GATE-M>DEV_ACTION_COMPLETED</GATE-M>
I	01-24 12:47:3...	2099	2134		ClipboardS...	mCBPickerDialog enter case. MSG_DISMISS_DIALOG
D	01-24 12:47:3...	2099	2153		KeyguardVi...	setHidden false
D	01-24 12:47:3...	2099	2153		KeyguardVi...	setHidden false
D	01-24 12:47:3...	2099	2153		KeyguardVi...	setHidden false
D	01-24 12:47:3...	2099	2153		KeyguardVi...	setHidden false
I	01-24 12:47:3...	1904	2052		SurfaceFli...	id=17 Removed HomeScreenActivity idx=2 Map Size=4

6. The spyphone dumps the heap using /proc/<pid>/maps and /mem. Accordingly, it can find the email structure, extract it and send it home.



```
00153C90  02 00 00 00 C3 0A 00 00 3C 21 44 4F 43 54 59 50  ....Q...<!DOCTYP
00153CA0  45 20 48 54 4D 4C 20 50 55 42 4C 49 43 20 22 2D  E HTML PUBLIC "-
00153CB0  2F 2F 57 33 43 2F 2F 44 54 44 20 48 54 4D 4C 20  //W3C//DTD HTML
00153CC0  33 2E 32 2F 2F 45 4E 22 3E 0D 0A 3C 48 54 4D 4C  3.2//EN">..<HTML
00153CD0  3E 0D 0A 3C 48 45 41 44 3E 0D 0A 3C 4D 45 54 41  >..<HEAD>..<META
00153CE0  20 48 54 54 50 2D 45 51 55 49 56 3D 22 43 6F 6E   HTTP-EQUIV="Con
00153CF0  74 65 6E 74 2D 54 79 70 65 22 20 43 4F 4E 54 45  tent-Type" CONTE
00153D00  4E 54 3D 22 74 65 78 74 2F 68 74 6D 6C 3B 20 63  NT="text/html; c
00153D10  68 61 72 73 65 74 3D 57 69 6E 64 6F 77 73 2D 31  harset=Windows-1
00153D20  32 35 32 22 3E 0D 0A 3C 4D 45 54 41 20 4E 41 4D  252">..<META NAM
00153D30  45 3D 22 47 65 6E 65 72 61 74 6F 72 22 20 43 4F  E="Generator" CO
00153D40  4E 54 45 4E 54 3D 22 4D 53 20 45 78 63 68 61 6E  NTENT="MS Exchan
```

iOS-based devices

A spyphone targeting iOS-based devices generally needs to first Jailbreak the device, and then installs the container-bypassing software.

1. The attacker installs a signed application on the targeted device, through the Enterprise/ Developer certificate.
2. The attacker uses a Jailbreak exploit in order to inject code into the secure container. We use the standard DYLD_INSERT_LIBRARIES technique to insert our libraries into the shared memory. In this manner, our (signed) dylib will be loaded into memory when the secure container executes.
3. The attacker removes any trace of the Jailbreak.
4. The spyphone places hooks into the secure container using standard Objective-C hooking mechanisms.
5. The spyphone is alerted when an email is read and pulls the email from the UI elements of the app.
6. Finally, the spyphone sends every email loaded to the spyphone's C&C server

Conclusions

The underlying notion of the secure container is that they depend on the integrity of the host system. This encourages us to deliberate the added value of the secure container:

- If the host system is *uncompromised*, what is the added value?
- If the host system is *compromised*, what is the added value?

Since the security of these secure containers is dependent on the integrity of the host system, it is enough for the attacker to target the host system.

In fact, we have seen this movie already. Desktop applications which have attempted to secure themselves, were targeted through the underlying OS. Although mobile OSes attempt to circumvent similar attacks by blocking off the OS to attackers and users alike, common and ever increasing JailBreaking/Rooting methods are rendering this safety mechanism irrelevant to targeted attacks.

In a similar fashion, the lessons learnt from the desktop equivalent may be applied here. If today the security industry understands that controls on devices themselves are not sufficient anymore to the real-world, we can expect the same in the mobile world.

It is important to recognize that infection is inevitable. As demonstrated throughout this whitepaper, MDMs cannot provide absolute security. They are certainly a beneficial tool for certain use cases such as management, compliance enforcement, DLP and physical loss. However, MDM is static and thus, inefficient against the dynamic nature of cyber-crime.

When introducing mobile devices into the organization, security professionals must ensure that they have enough visibility of the device's behavior in order to assess risk in real-time. Only then will they have enough information to take the right action required for mitigating the threat.

Risk metrics that assist in gaining visibility include:

- **Vulnerabilities and device usage.** Questions to ask include: is the device up to date? Are there any known vulnerabilities pertaining to the device's OS? Is the device connecting to a public hotspot?
- **App behavior.** For example, information obtained by static and dynamic analyses. Static analysis provides the necessary information on common app behavior. Dynamic analysis shows the actual app flow.
- **Correlation of events.** For example, outgoing text messages when the phone is locked should certainly raise an alert.

Risk metrics for real-time assessment is obtained by looking at network behavioral analysis and include:

- **Anomaly detection of communications of apps.** For example, communication to unknown servers.
- **Outgoing content inspection**. Inspecting what data is being extracted. Of course, this can only be done when the content is unencrypted.

- **Blocking of exploit and drive-by attacks**. This can be performed, for example, by testing app downloads and blocking links which lead to rogue sites.

About Lacoon Mobile Security

Lacoon has pioneered a mobile security solution to provide enterprises multi-layered protection against emerging and targeted threats to mobile devices. Going beyond the simple known-attack detection of traditional anti-virus, Lacoon combines sophisticated network and device protection capabilities to deliver a truly comprehensive mobile security solution for your organization.

Lacoon Mobile Security is founded by mobile security experts from the Defense and Telco industries and is backed by security industry veterans such as the co-founders of Checkpoint, Imperva and Trusteer.