CHAPTER 3

Mobile Security Countermeasures

So far I've outlined many of the mobile device threats that could lead to data loss. Fundamentally, when considering data loss one must encompass data-at-rest and data-in-motion to ensure confidentiality and integrity of the data. But a mobile device is more sophisticated than that. This involves protecting data on the device, data in the app, and data over the network (Figure 3.1).

Fortunately, mobile devices and complimentary products leverage new features in the mobile operating systems not previously found in traditional PCs. Let's continue by detailing these newer features and outline countermeasures to many of these aforementioned threats.

MOBILE OS COMPROMISE

In the previous chapter I outlined a myriad of ways in which a mobile device can become compromised. There are multiple approaches for detecting and mitigating this threat. First, the EMM client should provide ways to identify an OS compromise locally on the device, and then report that back to the console. In response, the administrator should have a policy to quarantine devices when a compromise is detected. This automation should allow the console to send down a Selective or Full Wipe of the device. A selective wipe would remove the enterprise data only, while leaving the personal data alone. A full wipe of course wipes the entire device back to factory defaults, and is typically only suited for corporate-owned devices. Selective wipes can be accomplished in a few ways. One way is to remove the previously deployed configuration profiles such as email, Wi-Fi, VPN, etc. Additionally, managed apps and/or their data can also be removed (note that this capability varies across the different mobile operating systems). When using a container, the selective wipe would purge the container itself.

Mobile Data Loss. DOI: http://dx.doi.org/10.1016/B978-0-12-802864-3.00003-9

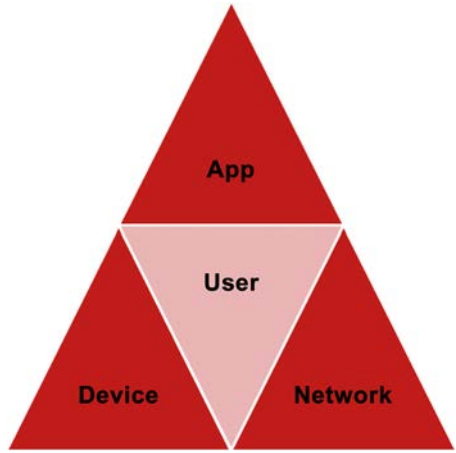

Figure 3.1 Mobile Data Loss Protection Triad.

Also, when a compromised device is detected, other lockdowns can occur. For example, the mobile device can also be automatically blocked from remote access to the network by a secure mobile gateway, until the device is brought back into compliance. The same can be done for the local network. A similar approach can be employed with NAC (Network Access Control), where the NAC solution checks in with the MDM/EMM when a device connects to the network to determine its security posture and if it's a registered device. If out of compliance, the NAC can block access similar to a secure mobile gateway. In terms of cloud services, EMM integration with Azure Active Directory can block rogue and out-of-compliance devices from accessing Office 365.

It's important to note that there's an issue not addressed by the aforementioned countermeasures that is lost or stolen devices. Assuming the lost or stolen device remains on the network, the EMM can still receive threat notifications from the EMM client and issue a quarantine to protect corporate data with a selective wipe. But if the device is a Wi-Fi-only device and it's no longer on the Wi-Fi, how does the EMM still quarantine the device? If it's off the network, the EMM loses visibility into the device.

More recently some EMM products have added *offline* policies that can reside on the device, specifically when using a container solution for your enterprise data. The local EMM client can still look for the same types of OS Compromise threats, but now when a threat is detected it doesn't need to "phone-home" to the EMM management console to

receive a quarantine command. Instead a local policy selectively wipes the container. This is particularly helpful in organizations that have many Wi-Fi-only mobile devices. In fact, the PCI Council added this to its Mobile Point-of-Sale (POS) "Mobile Payment Acceptance Security Guidelines v1.1, July, 2014.[1]"

Most recently in Windows 10, the operating system now performs a device health check to validate the integrity of the device during the bootup process. This can then be reported to the MDM or EMM and used to block access to corporate resources.

Summary of Mobile OS Compromise Countermeasures:

- PIN or Password enforcement
- Encryption
- Containerization of enterprise data
- OS Compromise detections (Jailbreak and Root detections) and Quarantine
  - Online selective wipe
  - Offline selective wipe
- Out-of-compliance device triggers the network gateway to block access

MALWARE AND RISKY APPS

Based on the plethora of threats I outlined in chapter "Understanding Mobile Data Loss Threats," it's important to detail an approach to deterring malware and risky app behaviors. Since we know that iOS is no longer immune to malware threats, a comprehensive mobile security strategy should address these threats across all of your mobile devices.

Anti-virus alone has taken a backseat to more comprehensive mobile malware security products. The reason for this is that on a mobile device anti-virus is just another app, and therefore the sandboxing limits its ability to remove a malicious app, limiting it to alert the user and rely on them to remove it. This is very different from the PC world where we've always relied on anti-virus to both identify the threat *and remove it.*

Due to this shortcoming of anti-virus alone, a new group of products has emerged referred to as App Reputation and Mobile Threat Prevention. This is a broad exploding category of products

[1] https://www.pcisecuritystandards.org/security_standards/documents.php

designed for mobile threats. The key difference here is that they all integrate with the EMM to leverage the EMM's ability to respond to an identified threat with a quarantine.

App Reputation commonly uses the EMM app inventory of the mobile devices under management and correlates it against their database of known malicious and risky apps. It will then report on malicious or risky behaviors for each app, either in its own console or also in the EMM console to give the administrator a single monitoring dashboard. The App Reputation may then feed into an EMM App blacklist to spawn a quarantine. It may also tie into APIs to allow profiles to be removed from the device and selectively wipe corporate data.

Mobile Threat Prevention is also a broad category of products that rely largely on an anti-virus-like app on the device that may include some intrusion detection features, malicious app behaviors, and more. These products can also integrate with an EMM to kick off a quarantine when a threat is identified on a mobile device. Furthermore, some of the features between App Reputation vendors and Mobile Threat Prevention vendors have also begun to overlap. Some App Reputation vendors have added an app to analyze local behaviors on the device, thus providing a more defense-in-depth approach.

These products are changing quickly with more features always being added. App Reputation and Mobile Threat Prevention solutions are very important to an overall Mobile Security Strategy as concerns about malware continue to increase.

ACCESS CONTROL AND CONDITIONAL ACCESS

Ensuring the network is secure for remote access is key in a mobile world. Traditionally in the PC world this has been delivered through a remote access VPN. Mobile requires a more mobile aware secure gateway. This gateway can control access to resources such as ActiveSync or Lotus Notes email. In addition, it can control access to content, internal web services, and application servers. Access control is performed by authenticating the user and the device.

When a device is under MDM or EMM management, the management system can collect hardware and software information about the device. This is key to eliminating impersonation and cloned devices, and

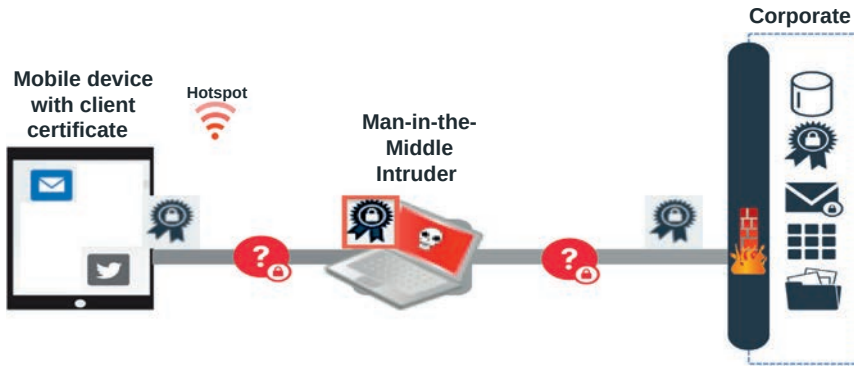

Figure 3.2 Thwarting a Man-in-the-Middle Attack.

used for authenticating the device. In addition, the security posture can be analyzed to identify when a device is outside of corporate compliance policies, as defined in the security policy. By combining this with user authentication, the device authentication provides yet another factor of authentication when a device remotely connects to the network and is far superior to traditional gateways.

Most of the mobile operating systems have native support for certificates, making it quite easy for certificates to be deployed with an EMM profile automatically for authentication, unlike their PC counterparts, which normally required cumbersome manual techniques for deploying certificates to users PCs and laptops. Therefore, when a profile is deployed to a device for services such as email, SharePoint, and intranet web access, a certificate can be generated and deployed to the device automatically. This also eliminates hassles such as required password changes every 90 days. It also allows an organization to meet security or compliance requirements requiring strong factor or two-factor authentication. When combined with a secure mobile gateway, it also provides *proactive* protections against MitM attacks by offering both mutual authentication, and certificate pinning on the secure mobile gateway (Figure 3.2).

Steps to thwarting a MitM attack:

1. Attacker presents fake server-side certificate (impersonating the network back at corporate)
2. Certificate pinning prompts the fake certificate to be compared to what has previously been sent to the device and quickly identifies that they don't match

3. Client certificate mutual authentication handshake fails
4. No per-App VPN tunnel is set up
5. No data communicated
6. Data breach is prevented

A secure mobile gateway also can support mobile-specific encrypted protocols, such as per-App VPN over SSL/TLS. This was released in iOS 7, and gained mass support across public apps in iOS 8 and iOS 9. Supporting a VPN at the app-level allows the administrator to further refine what apps can access the corporate network. In contrast, a VPN typically allows all apps to access the network, including malicious apps. A per-App VPN provides additional layers or security as well as better efficiencies and ease-of-access for the user.

LOCKDOWNS AND RESTRICTIONS

Lockdown and restriction APIs have been available from device manufacturers for some time, and allow EMM solutions to leverage these APIs to disable features. These include unwanted network services (Bluetooth, IRDA, NFC, etc.), device level features (camera, screenshot, etc.), and a plethora of other lockdowns. These vary across the different mobile operating systems.

Furthermore, many EMM solutions allow these to be applied to manage mobile devices in different ways. For example, for a mobile POS, unwanted services such as Bluetooth or NFC can be disabled to avoid targeted attacks. But disabling these on BYOD devices may not be desirable since users commonly use these services for Bluetooth headsets, NFC-based retail purchases, and more. It's important to ensure when implementing these controls to evaluate each of the use-cases and perhaps different lockdown and restriction policies for each scenario.

LIVE MONITORING, AUDIT LOGS, EVENTS, AND REPORTING

EMM solutions provide inherent live monitoring of mobile devices. This can be mobile device monitoring, device security posture monitoring, network access monitoring, and more. Additionally, EMM can integrate with SIEM, Big Data Analytic products, App Reputation, Mobile Threat Prevention, Network Access Control, and proxy solutions. All of these provide the ability for logging, alerting, correlation, and reporting.

The administrator can force a device check-in to check the security posture or location of the device. Per-device logs can be stored in the EMM to allow deep analysis by the administrator. While this may be helpful for troubleshooting, it can also be helpful for security analysis. Furthermore, an EMM can provide information about when a device is connected to a network, and to what resources.

INCIDENT RESPONSE AND FORENSICS

In the event of a breach or incident, investigators are quick to perform an acquisition on a mobile device. But on-device mobile forensics is becoming increasingly difficult. Many of the mobile device forensic acquisition tools have historically required vulnerabilities, hacks, or even formal jailbreak or root to bypass protections to gain access to the data. As previously outlined, many of these techniques may perform a wipe or selective wipe, even if the device is off the network (or in a Faraday bag).

A blind spot for many investigators is that an EMM may hold some significant evidence. It also doesn't require breaking into the mobile device. The following list outlines just a sampling of the EMM data available to the investigator:

- Remote unlock of the mobile device
- Device hardware and software information
- An inventory of Apps on the device
- Last known location of the device or a bread crumb trail of where the device has been
- When the mobile device connected to the corporate email
- When the mobile device connected to corporate app
- What malicious apps are installed on the device that may have led to a breach
- When a malicious app was installed on a device
- If the device is compromised
- When the device was compromised
- Audit logs of files uploaded to personal cloud services

As you can see, an EMM solution provides a wealth of information to the investigator to answers questions such as when, where, what, why, and how. While although on-device forensic acquisition is valuable, EMM may provide answers more quickly and easily. This is

especially important in the event of a breach and time-to-resolution; and is incredibly helpful in a liturgical and nonliturgical forensic investigations. Mapping out these EMM data points is key to updating your incident response lifecycle and response procedures.

MOBILE DEVICE UPDATES AND PATCHING

In sharp contrast to the PC world, users are in control of when mobile device updates and patches are applied. This is very much the case with iOS and Android, and is becoming more of the case with Windows such as in Windows Phone 8/8.1 and forthcoming in Windows 10. This can be problematic for specific use-cases where an organization would like to test an update with all of their apps to avoid software issues. Single-app mode (iOS) or Kiosk-mode (Android) can limit the user from performing an update.

EMM solutions can provide a way to enforce updates to ensure that vulnerabilities are patched. This can be performed through a security policy that blocks network access or other enforcements to encourage users to perform the update. But there are obstacles and a lack of APIs (Application Program Interface) to enforce the mobile operating system updates from the EMM.

WEARABLES

Wearables and smartwatches didn't become a topical risk concern for most organizations until the release of the Apple Watch. There was certainly the fear of the unknown. Are these devices risky or not? What happens if a device is hacked or lost? The fact is that wearables and smartwatches have been around for years prior to the Apple Watch, and some can be paired with an iOS device in addition to Android and Windows devices.

There are fundamental differences between wearables and smartwatches versus their mobile device counterparts. These smartwatches typically require a pairing app on the mobile device to allow the smartwatch to be paired over the air. The most important difference of a smartwatch versus a mobile device is that the built-in security for smartwatches is more proximity based rather than PIN or passcode-based. With mobile devices typically the first security requirement

most organizations have is to enforce a PIN or passcode on the device to protect the data in the event that the device is lost or stolen. Smartwatches typically use a proximity-based approach. This can rely on identification of it residing on the user's wrist, and when it's removed a PIN or Passcode prompt is enabled to protect it. In other smartwatches, this proximity-based protection is based on whether the device is communicating over Bluetooth to the paired mobile device. When the Bluetooth connectivity is lost, the PIN or passcode is enabled.[2]

Management APIs are starting to appear for the Apple Watch. Apple has provided the ability to detect when an Apple Watch has been paired to an Apple iPhone. Other controls include blocking access to enterprise data using containerization as well as blocking the smartwatch pair apps. Look for this area to mature over the next few years. For now, considering using App-level security or containerization to mitigate the syncing of enterprise data to smartwatches. In the case of the Apple Watch, there are methods of embracing the Watch Kit extension for those enterprise apps that you would like to sync with a smartwatch, and level the encryption capabilities in combination with this.

DEVICE ENCRYPTION AND CONTAINERS

Most of the devices today across iOS, Windows, Android, and more provide operating system-level encryption either enabled by default or as an option. Furthermore, this can be enforced by the EMM as part of the enforcement policy. This is one of the fundamental requirements of most mobile security strategies.

But encryption alone doesn't prevent users from sharing data. To accomplish that requires a container to control sharing of corporate data through separate encryption and data loss prevention controls. This container can include email, secure access to corporate content (fileshares), web browsing, and corporate apps and data. Data can be shared across the apps within the container, but can block unwanted cloud services or sharing of data with apps outside of the container. In addition, it should provide controls to copy/paste, open-in, sharing, and other behaviors that allow moving of data to and from the corporate container.

[2]https://www.mobileiron.com/en/whitepaper/smartwatches-wearables-and-mobile-enterprise-security MobileIron Analysis of Smartwatch Security Risks to Enterprise Data

Typically the container is separately encrypted from the rest of the device. This autonomous encryption can prevent the container data from being exposed, even if the device is compromised or infested by malware. And furthermore, when a device is compromised the container can wipe the container data in real-time. People frequently ask about targeting data in memory on a mobile device. Aside from some device specific vulnerabilities, most device compromises require jailbreak or rooting behaviors, which additionally require a reboot of the device. Therefore to complete the compromise, you reboot the device thus wiping volatile memory. So previously viewed documents are gone, and not exposed to memory analysis tools such as IDA Pro, *after the compromise.* There are always exceptions to every scenario, so it's important to embrace the other outlined layered security to eliminate any single point of exposure. In this case, app reputation, mobile threat prevention, requiring mobile operating system updates and patches before accessing corporate data, operating system compromise detections, quarantine, and numerous other controls can further protect against these types of exposures.

PINs, PASSWORDS, AND PASSCODES

Determining passcode enforcement policy can be challenging for some organizations. It typically stems from traditional PC and Server 8-character password policies that require various complexities to achieve compliance or traditional security best practices. This is a prime example of traditional policies that just don't work well in the mobile world. Requiring a user to enter an 8-character complex password to unlock their mobile device makes for a horrible user-experience.

Users are accustomed to a 4-character PIN. Most EMM policies can then enforce various complexities or wipe a device after 10 bad PIN entries. Many security conscious organizations have embraced App-level or Container-level passcodes to protect corporate data. And in those cases, some have incorporated a 6-character PIN or passcode at an App-level or Container-level.

Bottomline: it comes down to the organization, but it's very important to consider the broader mobile security controls not found in the typical PC world (eg, Wipe after 10 bad passcode entries). It's important to balance that with the user-experience to avoid lack of mobile

adoption or causing users to circumvent security controls in other ways, commonly referred to as Shadow IT. Some of these options can include fingerprint authentication through Apple's Touch ID or Samsung's fingerprint scanner. This can be use to authenticate at a device or container level.

CLOUD

One of the key questions most people ask is how can an organization separate personal cloud from enterprise cloud (Enterprise File and Sync Share services). Early on, mobile administrators would blacklist the personal cloud apps, but this is like playing "whack-a-mole." If you block one personal cloud repository, the users will just find another.

At a device level it's important to provide an enterprise solution to users. Some of the most popular solutions have an Enterprise version of their app, which can also embed the SDK provided by an EMM. This allows that app to then work in unison with the EMM containerization to require users to upload enterprise data (in the container) using only that app versus personal cloud apps. Another approach is to leverage a containerized documentation collaboration app that allows webdav access to the enterprise cloud repository. For additional tips, see the "File-Level Security" section in this chapter.

FILE-LEVEL SECURITY

Users want to store documents and files in personal cloud services. In many cases they don't distinguish between personal and corporate files; therefore it's common for an employee to upload a file to share with another employee, business partner, or prospective client. Mobile Data Loss Prevention (DLP) controls and containerization are designed to prohibit such behaviors to avoid mobile data loss. But when these controls ruin the user-experience, employees will attempt to circumvent those controls resulting in Shadow IT. To overcome this issue, another approach is to embrace the personal cloud services rather than block them.

File-level security is about tying security to a corporate document. With this approach, a user can use their favorite cloud service for uploading and sharing corporate documents. When a file is shared to a

personal cloud service, the file is first encrypted before being uploaded. If the file is then shared with another employee, the key escrow at their company allows the file to be downloaded and decrypted for the employee to access and use the document. But when the file is shared with a nonemployee, the file remains encrypted and unusable by the nonemployee. This is a nice compliment to a defense-in-depth mobile strategy and creates a great user-experience for mobile users.

SUMMARY

Threats will always exist and continue to evolve. Implementing a layered security approach is key to succeeding in your mobile security strategy. But success is not always about avoiding a breach altogether, but also being prepared to respond to it. A thorough incident response plan can mitigate data loss and prepare you for when a breach occurs. If your security team doesn't have a mobile-specific incident response methodology, they should. Engage your team to ensure the vetted processes are in-place to response. Many times we've heard "it's not a matter of if, but a matter of when"; be prepared. All of the recommendations outlined should also have a tie-in to your incident response plan. The countermeasures defined in the chapter should help in implementing your defense-in-depth mobile security strategy.