Security Configuration Recommendations for Apple® iOS 5 Devices

Revision 0

March 28, 2012

The Mitigations Group

of the

Information Assurance Directorate

National Security Agency

9800 Savage Rd. Suite 6704

Ft. Meade, MD 20755-6704

Warnings

- As with any other information system, do not attempt to implement any of the recommendations in this guide without first testing in a non-production environment.
- This document is only a guide containing recommendations. It is *not* meant to replace well-structured policy or sound judgment. Furthermore this guide does not address site-specific configuration concerns. Care must be taken when implementing this guide to address local operational and policy concerns.
- The configuration settings described in this document apply only to the following devices running Apple iOS 5, which is at version 5.0.1 at the time of this writing. The guidance may not translate gracefully to other systems or versions, although applying system updates is always recommended.

Apple Device
iPhone 3GS
iPhone 4
iPhone 4s
iPod Touch (Gen. 3 or 4)
iPad
iPad 2

- Internet addresses referenced were valid as of 9 February 2012.

Disclaimer

This Guide is provided "as is." Any express or implied warranties, including but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the United States Government be liable for any direct, indirect, incidental, special, exemplary or consequential damages (including, but not limited to, procurement of substitute goods or services, loss of use, data or profits, or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this Guide, even if advised of the possibility of such damage.

The User of this Guide agrees to hold harmless and indemnify the United States Government, its agents and employees from every claim or liability (whether in tort or in contract), including attorneys' fees, court costs, and expenses, arising in direct consequence of Recipient's use of the item, including, but not limited to, claims or liabilities made for injury to or death of personnel of User or third parties, damage to or destruction of property of User or third parties, and infringement or other violations of intellectual property or technical data rights.

Nothing in this guide is intended to constitute an endorsement, explicit or implied, by the U.S. Government of any particular manufacturer's product or service.

Trademark Information

Table of Contents

1. Introduction

Purpose. This document provides security-related usage and configuration recommendations for Apple iOS devices such as the iPhone, iPad, and iPod touch. This document does not constitute Department of Defense (DoD) or United States Government (USG) policy, nor is it an endorsement of any particular platform; its purpose is solely to provide security recommendations. This guide may outline procedures required to implement or secure certain features, but it is also not a general-purpose configuration manual.

The guidance provides recommendations for general-purpose business use of iOS devices for processing data that is UNCLASSIFIED, and possibly Sensitive But Unclassified. Such data may carry various designations such as For Official Use Only, Law Enforcement Sensitive, or Security Sensitive Information. Approval for processing such Sensitive But Unclassified data is dependent upon risk decisions by Designated Approving Authorities (or their analogs in non-DoD entities).

Audience. This guide is primarily intended for network/system administrators deploying Apple's iOS devices or supporting their integration into enterprise networks. Some information relevant to IT decision makers and users of the devices is also included. Readers are assumed to possess basic network and system administration skills for Mac OS X or Microsoft Windows systems, and they should have some familiarity with Apple's documentation and user interface conventions.

Scope. Apple's mobile devices, including the iPhone and iPad, are prominent examples of a new generation of mobile devices that combine into a single device the capabilities of a cellular phone, laptop computer, portable music player, camera, audio recorder, GPS receiver and other electronics. The capabilities of such devices are considerable but, as with any information system, also pose some security risks. Design features can seriously mitigate some risks, but others must be considered as part of a careful, holistic risk decision that also respects the capabilities enabled by such devices. Major risks, and available mitigations, are discussed in Section 1.3.

Security guidance for mobile devices must cut across many previously discrete boundaries between technologies. For example, scrupulous deployment of an iPhone includes consideration not just the settings on the device itself, but those of the Wi-Fi networks to which it will connect, the VPNs through which it will tunnel, and the servers from which it will receive its configuration. This guide provides recommendations for the settings on an iOS device itself, as well as closely-related information for the network and configuration resources upon which deployment of iOS devices depends.

1.1 How to Use This Guide

1.1.1 Read Sections Completely and in Order

Each section tends to build on information discussed in prior sections. Each section should be read and understood completely. Instructions should never be blindly applied. Relevant discussion may occur immediately *after* instructions for an action, so be sure to read the whole section before beginning implementation. Careful consideration is essential for deploying iOS devices in an enterprise environment where multiple supporting devices and software components may need to be configured properly in order to function.

1.1.2 Understand the Purpose of this Guidance

The purpose of the guidance is to provide seurity-relevant configuration recommendations. It does not imply the suitability or unsuitability of any product for any particular situation, which entails a risk decision.

1.1.3 Limitations

This guide is limited in its scope to security-related issues. This guide does not claim to offer comprehensive configuration guidance. For general configuration and implementation guidance refer to other sources such as Apple[1].

1.1.4 Test in Non-Production Environment

To the extent possible, guidance should be tested in a non-production environment before deployment. Ensure that any test environment simulates the configuration in which the devices will be deployed as closely as possible.

1.1.5 Formatting Conventions

Commands intended for shell execution, file paths, and configuration file text, are featured in a `monospace font`. Menu options and GUI elements will be set in a **Bold, sans-serif font**. Settings appropriate to the device itself will be typeset in-line (i.e. **Settings** ▷ **Airplane Mode**). Actionable instructions are typically embedded in a box.

1.2 General Principles

The following general principles motivate much of the advice in this guide and should also influence any configuration decisions that are not explicitly addressed.

1.2.1 Encrypt Transmitted Data Whenever Possible

Data transmitted over a network, whether via wire or wirelessly, is susceptible to passive monitoring. Whenever practical mechanisms exist for encrypting this data-in-transit, they should be applied. Even if data is expected to be transmitted only over a local network, it should still be encrypted if possible. Encrypting authentication data, such as passwords, is particularly important. iOS's support for SSL, WPA2, and VPN protocols demonstrates its capabilities, when such features are activated, to adhere to this principle.

1.2.2 Encrypt Stored Data Whenever Possible

Data on mobile devices is particularly susceptible to compromise due to loss of physical control. Whenever practical solutions exist, they should be employed to protect this data. The Data Protection API on iOS devices is used by some applications, and demonstrates the devices' capability to provide such protection. Drawing on applications which use this capability (and ensuring that internally-developed enterprise applications also use it), and setting an appropriately complex passcode, follow this principle.

1.2.3 Minimize Software to Minimize Vulnerability

The easiest and simplest way to avoid the vulnerabilities in a particular piece of software is to avoid installing the software altogether. Hundreds of thousands of 3rd-party applications, or "apps," written by thousands of different developers are available for iOS devices. These developers may have willfully or accidentally introduced vulnerabilities. For some environments, a particular app may fulfill a mission-critical need. In other cases an app might needlessly introduce additional risk to the system. Certain risk scenarios may call

for minimizing apps. BYOD scenarios, on the other hand, generally imply the consideration and acceptance of such risks.

1.2.4 Leverage Security Features, Never Disable Them

Security features should be effectively used to improve a system's resistance to attacks. These features can improve a system's robustness against attack for only the cost of a little effort spent doing configuration. For example, iOS's enforcement of code signing of apps provides assurance of integrity both during installation and at runtime. Disabling this feature through the use of "jailbreaking" tools provided by the hacker community significantly decreases an iOS device's resistance to attack.

1.2.5 Grant Least Privilege

Grant the least privilege necessary for users to perform tasks. The more privileges (or capabilities) that a user has, the more opportunities he or she will have to enable the compromise of the system (and be a victim of such a compromise). For example, a configuration profile can disallow use of the Safari web browser and the camera. Disabling the camera prevents a malicious or careless user from photographing sensitive areas, while disabling Safari helps ensure the user is protected from any web-based attacks (albeit at significant loss of capability). Similarly, it is possible to restrict the installation of third party apps, and this may be the right balance between security and functionality for some environments.

1.3 Risks, Mitigations, and Consequences

Understanding the risks – and available mitigations – involved in the deployment of smartphone platforms such as iOS provides a background for certain risk decisions. An attacker who has compromised *any* mobile device, and can remotely maintain control of that device, can use this access to gather a great deal of information about the user of the device and his or her environment. As described by NIST Special Publication 800-124 [7], the consequences of such attacks include:

- collecting audio ("hot-microphone" eavesdropping)
- using the cameras
- geolocation of the device (and presumably the user)
- collecting all data, including credentials stored on the device or accessed by it
- acting as the user on any network to which the device later connects

The following table describes risks (with attack vector) along with applicable mitigations that are either built into the iOS platform or can be employed by administrators or users. The following table is provided as a summary for risk decision makers – and to motivate administrators to apply mitigations whenever practical. *It should not be used to draw comparisons between iOS and other platforms.*

Risk	Mitigation
Data Compromise due to Lost Device (still reachable over any network interface — cellular or WiFi)	Enabling a Passcode provides protection for Apps that leverage the Data Protection API, such as Apple's Mail app and 3rd party apps, as well as for credential storage in Keychains. Using the latest hardware currently prevents usage of public Jailbreak tools to access other data on a lost device. Activating a remote wipe can be performed via ActiveSync, MDM, or iCloud. Find My iPhone or other geolocation could permit the lost device to be located.
Data Compromise due to Lost Device (not reachable over any network interface)	Enabling a Passcode encrypts some data on the device. Using the latest hardware currently prevents usage of public Jailbreak tools to access other data on a lost device.
Data Compromise due to Casual Access Attempt	Enabling a Passcode prevents a casual snoop from accessing the device. Provide user training to stress importance of physical control at all times.
Data Compromise via Host Computer Backup/Sync	Ensure proper hygiene and configuration of systems used for backup/sync. This may entail enterprise rollout of iTunes, to ensure protection of backup data. Train users not to connect their device to any untrusted computers/devices and provide additional AC outlet chargers. Encrypting iOS device backups in iTunes can mitigate data loss if the host computer is later compromised or lost.
Exploitation of Device via Malicious app	The Sandboxing feature prevents apps from carrying out certain malicious activities. The App Store's app vetting process provides accountability for developers, which discourages creation of malicious apps. Disabling the App Store, or permitting only installation of Enterprise-created Apps, further mitigates any threat from 3rd party app developers (at significant cost to capability).
Exploitation of Device via Malicious WiFi Network	Apply software updates. Provide user training on connecting only to trusted networks. Provide user training to encourage use of the VPN.
Exploitation of Device via Bluetooth Communications	Apply software updates. Monitor compliance with MDM software. iOS only implements a small subset of the available BlueTooth profiles, which decreases its likelihood to contain vulnerabilities that would give rise to exploitation.
Exploitation of Device via Cellular Network (e.g. SMS/MMS, baseband communications)	Apply software updates. Monitor compliance with MDM software. Provide user training to ensure awareness during travel.
Exploitation of Device via Malicious Email or Web Page	Apply software updates, with particular vigilance after public release of jailbreak tools. Monitor compliance with MDM software.

2. Configuration Deployment

This chapter presents information about creating and deploying settings to iOS devices, which are generally contained in *configuration profiles*. Configuration profiles are simply XML files that conform to Apple's XML DTD and the plist format. Additional information is available at http://www.apple.com/iphone/business/resources/. The settings contained in configuration policies are discussed in Chapter 3.

2.1 Nature of Configuration Profiles

> Understand that a user who controls an iOS device can opt to erase the device, which erases all data from the device including any configuration profiles. Understand also that users can typically append further restrictive settings, as well as services, onto the device, even in the presence of a configuration profile. Configuration Profile settings enforcement on the iOS devices are cumulative indicating that they can further restrict existing settings when applied.

iOS configuration profiles specify a collection of settings that can control some security-relevant behavior of an iOS device, but are not designed to provide an enterprise with total, arbitrary control over the user's device.

A "carrot and stick" approach can be employed to avoid tempting users to remove a configuration profile (either directly or via device reset). Bind "carrots" (such as credentials needed for enterprise access) to "sticks" (such as a passcode policy) in a single configuration profile. Removing a configuration profile implies that credentials necessary for accessing enterprise services (such as VPN certificates or email accounts) would also be lost, and thus deny the user such services. Also in this case, MDM software would become unable to query the device and the enterprise would be alerted as to the device's unmanaged status.

2.2 Mobile Device Management Software

Third-party MDM products, as well as Apple's own OS X Lion Server, can automate the deployment of configuration profiles and carry out the operational management of devices. Configuration profiles can also be provided via secure web-based services. Configuration profiles can also be created using Apple's iPhone Configuration Utility ("iPCU") as described in Section 2.3.2, but it does not provide mechanisms for automated deployment or reporting. iPCU provides a convenient means of surveying the settings which can be deployed to devices, although there is no guarantee that a particular MDM product will support all settings.

2.2.1 Select Mobile Device Management (MDM) Software

> Select an MDM product which uses Apple's MDM API, unless enterprise management of the devices is not needed.

Apple's MDM API provides the supported mechanism for enterprise device management, and various 3rd party vendors have built products upon it. For more information, see http://www.apple.com/iphone/business/integration/mdm/.

2.2.2 Understand Capabilities of MDM Software

Mobile device management software may also include features that are not part of the supported Apple MDM API:

- "Jailbreak detection" can determine if a user has chosen to jailbreak his or her device, which is a useful feature for enterprises who monitor compliance. However, it does not provide high assurance that a device has not been maliciously jailbroken by a sophisticated attacker. The situation is analogous to "root detection" on another mobile platform. It is also analogous to the historical and difficult problem of rootkit detection on desktop or server operating systems. In all these cases, the operating system itself becomes compromised. Since it alone operates at the most privileged levels, there are limits to the extent to which any add-on software can "ask a liar if he is lying."

  The system's cryptographically-verified boot process, runtime enforcement of code signatures, app sandboxing mechanism, controlled software distribution model via "app stores", and rapid software update capability very strongly address the problem of jailbreak-based attacks by themselves. Using add-on software to query for signs of jailbreak may provide an additional layer of defense.

- "Secure containers" can provide data-at-rest protection and data-in-transit protection. These are typically software libraries included by 3rd party apps, which then make use of their functionality instead of that provided by the system's software libraries. These "containers" can be useful if the system's capable, built-in mechanisms which already provide these features do not meet particular requirements or certifications. Note, however, that they cannot protect their contents against privileged code running on the device, such as would result from a sophisticated, malicious jailbreak attack during system operation.

  They should also *not be confused* with the Sandboxing feature of the iOS kernel as described in Blazakis[4]. Rather, the Sandboxing feature strongly addresses the problem of malicious or co-opted apps trying to perform undesirable activities on the system (such as elevating their privileges) in the first place. Sandboxing constitutes a significant obstacle to attackers, but it does not allow apps to (rather inconceivably) protect themselves if the underlying operating system is compromised. App sandboxing may serve as a means of jailbreak detection as discussed above, in that an app which can access beyond its Sandbox may infer that it is running on a compromised device.

2.3 Deploying Configuration Profiles

After a configuration profile is created — typically in an MDM console — it must be deployed to devices. This section discusses methods available for installing a configuration profile onto an iOS device, along with their security implications.

Customizing profiles to individual users implies embedding sensitive authentication information within transmitted profiles. This introduces a need for confidentiality during transmission of such files.

2.3.1 Deploy Over-the-Air with Encryption and Authentication

If configuration profiles will be deployed over-the-air, ensure use of authentication and encryption.

If the iPhone can authenticate a configuration profile during its installation, the **Settings** ▷ **General** ▷ **Profile** screen will display **Verified**.

Over-the-air deployment that is authenticated and encrypted requires the support of enterprise infrastructure, such as directory services, an enterprise certificate trusted by iPhone, an SCEP server, and a web server. The server component of MDM products may provide some of this infrastructure.

Deploying configuration profiles to a device over-the-air consists of three major steps:

- *Authentication* of the user, typically leveraging existing directory services.
- *Enrollment*, which involves the device transmitting device-specific information to the enterprise, and receiving a device certificate in return.
- *Installation* of an encrypted, authenticated configuration profile onto the device.

Some MDM products include a server component that provides a web-based service for users to initiate this process, while others initiate the process by requiring users to download a particular MDM client app from the App Store which can facilitate the process.

Transmission and data formats used by the MDM protocol are thoroughly standards-based. Detailed, authoritative description of the transactions between the device and the enterprise are available to Apple-registered developers at http://developer.apple.com.

Additional description and security analysis is available in https://github.com/intrepidusgroup/imdmtools/blob/master/Presentations/InsideAppleMDM_ShmooCon_2012.pdf, linked from http://intrepidusgroup.com/insight/2012/01/changes-to-apple-mdm-for-ios-5-x/.

2.3.2 Manual Deployment with iPhone Configuration Utility

Manually using the iPhone Configuration Utility (iPCU) is the safest means of deploying configuration profiles to devices, but does not scale well as it depends on administrators' manual intervention. It also implies that an MDM server will not be used to remotely monitor device status. Nevertheless, transferring the profile to a device which is directly connected via USB cable avoids threats to confidentiality and integrity inherent in network transfers.

iPCU is available at http://www.apple.com/support/iphone/enterprise/ (cryptographic checksum unavailable). Documentation is available at http://help.apple.com/iosdeployment-ipcu/.

2.3.3 Avoid Unauthenticated, Unencrypted Deployment Methods

Avoid deployment of configuration profiles through methods that do not provide encryption and authentication. Email is especially discouraged.

It is possible to distribute configuration profiles to individual devices by emailing the profile to the user of the device or providing a link via SMS. Once the profile is accepted by the end user, the iOS device facilitates its installation. These methods are not recommended because they do not generally provide authentication of the sender of the configuration profile, or encryption of the profile itself during transmission. Users should generally be taught not to have confidence about the origin of email attachments or SMS messages.

Emailing configuration profiles also presupposes that the user has configured an email account on the iPhone.

Furthermore, once the configuration profile is in the receiver's mailbox, it will remain there until it is manually deleted. If the mobile profile contains sensitive information, its prolonged existence in an unmanaged mailbox poses additional risk.

3. Device Configuration

This chapter makes recommendations for security-relevant settings that reside on the iOS device itself. Section 3.1 presents settings that can be administratively deployed to an iOS device via a configuration profile. Deploying configuration profiles in a layered approach — with each profile containing payloads targeted for specific services — e.g. Exchange Access/requirements/access identity, is highly recommended so that an individual profile can be removed along with all the data for that one service without negatively affecting the rest of the devices and profiles. This approach involves creating one configuration profile for baseline services/devices, and then adding profiles for specific services. Many of the recommended settings can also be set manually on individual devices and provide value even if the device is not managed by an enterprise. Section 3.2 presents recommended settings that can only be applied manually.

3.1 Deployable Device Settings

The following subsections correspond to the types of configuration payloads, which can be surveyed via the iPhone Configuration Utility.

3.1.1 General

Apply the following General settings to identify the profile being deployed and to prevent users from removing the profile:

- Enter **Name** , **Identifier**, and **Description** as appropriate.
- Set **Security** to **Never** if practical, as it controls when and how the profile can be removed from the device. Letting end-users remove configuration profiles allows them to easily remove security settings contained in the profile. At the same time, this may not be practical for organizations implementing bring-your-own-device (BYOD) schemes which may allow for users to opt in or out at any time.

  Note that **Never** ties a configuration profile to the device, and cannot be removed unless the device is wiped. However, configuration profiles containing MDM Payloads cannot be set to **Never**. The user always has the ability to opt-out of MDM, but all configuration profiles, accounts, and data associated with those configuration profiles (delivered under MDM) would also be wiped from the device.

3.1.2 Passcode

The remarkable attention paid to passcode quality requirements represents misplaced priorities in the current network environment, as passcodes do not protect against many contemporary threats. However, setting a passcode enables cryptographic features that can protect data on the device if it is lost, stolen or temporarily out of possession. Hardware and software cryptographic features of iOS devices – not present on typical desktop or server systems – provide significant protections against the password-guessing threat when the passcode feature is enabled. Furthermore, iOS devices are likely to store only a single user's credentials, while complex passcode policies are designed to protect against the compromise of large numbers of credentials when they are stored on a single system that becomes compromised (such as a directory server). Onerous passcode policies may also drive users to attempt to subvert the settings. For these reasons, IT decision

makers should understand that onerous passcode policies on iOS devices provide little value (in the best case), and may end up being counterproductive. See Section 3.1.2.2 for discussion of which data is protected by enabling the passcode.

The following publicly-available research provides rationale for these recommendations:

- *Apple iOS 4 Security Evaluation* by Dino Dai Zovi at http://trailofbits.com/2011/08/10/ios-4-security-evaluation/. The slides' section "Attacking Passcode" provides highly relevant platform-specific discussion for iOS devices.
- NIST Special Publication 800-118 *DRAFT Guide to Enterprise Password Management* at http://csrc.nist.gov/publications/PubsDrafts.html#SP-800-118 provides discussion about factors that should be considered in order to create effective password strength recommendations.

3.1.2.1 Enable Passcode

The following passcode settings are recommended, and can be deployed via a configuration profile.

3.1.2.1.1 Disable Simple Value for Passcode

Set **Allow simple value** to Unchecked.

This will enable display of the entire keyboard for passcode entry, instead of only a numeric keypad (and setting other requirements will also enable this).

3.1.2.1.2 Require Alphanumeric Value for Passcode

Set **Require alphanumeric value** to Checked.

Requiring alphanumeric values should increase the search space in a password-guessing attack.

3.1.2.1.3 Set Minimum Passcode Length

Set **Minimum passcode length** to 6.

Setting a minimum length should increase the search space in a password-guessing attack. A passcode length of 6 may be a reasonable balance between user experience and security for many deployment scenarios.

3.1.2.1.4 Set Minimum Number of Complex Characters

Set **Minimum number of complex characters** to 1.

Requiring complex characters should increase the search space in a password-guessing attack.

3.1.2.1.5 Set Maximum Passcode Age

Set **Maximum passcode age** to 120 days, if there is a need for such rotation.

Changing passcodes regularly prevents an attacker who has compromised the password from re-using it to regain access. This is an unlikely scenario, but is addressed by setting a password expiration.

3.1.2.1.6 Set Auto-Lock Time

Set **Auto-Lock (in minutes)** to 5 minutes, or less.

This ensures that the device will require passcode entry if lost or left unattended.

3.1.2.1.7 Set Passcode History

Set **Passcode History** to 3.

This ensures that users cannot trivially alternate between passcodes.

3.1.2.1.8 Set Grace Period for Device Lock

Set **Grace period for device lock** to 5 minutes, or less.

This permits unlock of the device after a certain period of time has passed since the last device lock. Allowing a Grace Period enhances usability and makes users more likely to tolerate passcode requirements.

3.1.2.1.9 Set Maximum Number of Failed Attempts

Set **Maximum number of failed attempts** to 10 attempts, or fewer.

Setting the device to automatically erase after a number of failed attempts can protect against witless password guessing attacks conducted through the unlock screen. (However, it does not protect against those conducted by processes running on the device, see Section 3.1.2.2 for more discussion).

3.1.2.2 Understand Which Files are Protected by Encryption

Enabling a passcode activates the Data Protection feature of iOS. The Data Protection feature encrypts items with a key whose availability depends on entry of the passcode. Currently, the following items are protected:

- **Email messages** stored by the built-in Mail app
- **Inactive Apps' Screenshots** displayed at app re-launch to create impression of "instant resume"
- **Some Keychain Items** such as email passwords and iTunes backup password
- **Data stored by third-party apps** which use the Data Protection API

In fact, the rest of the files on the device are encrypted, but they are still available to an attacker who can get privileged code to execute on the device. This is because the encryption key for these files is available even without the passcode (unlike the files above).

For older hardware models such as iPhone 3GS, iPad, and iPhone 4, this remains possible using publicly-available tools which provide the ability to execute privileged code on any device in physical possession. Examples of such tools include The iPhone Dev Team's *redsn0w*, which itself leverages a collection of exploits including *limera1n* by George Hotz (geohot). No tools have been released by the hacker community which allow for exploiting the boot ROM in this manner for iPhone 4S, and iPad 2, however.

Note also that even if privileged code can be run by an attacker on a lost or stolen iPhone, a password-guessing attack against the protected files must be executed on the device itself. This is because the key which encrypts the items listed above is derived from the passcode as well as a key that is bound to the hardware of the device (and not considered extricable from it).

The following references provide detailed explanation:

- *iPhone data protection in depth* by Jean-Baptiste Bédrune and Jean Sigwald (Sogeti ESEC) available at http://esec-lab.sogeti.com/dotclear/public/publications/11-hitbamsterdam-iphonedataprotection.pdf linked from http://esec-lab.sogeti.com/post/iOS-5-data-protection-updates.
- *Apple iOS 4 Security Evaluation* by Dino Dai Zovi (Trail of Bits) available at http://trailofbits.files.wordpress.com/2011/08/apple-ios-4-security-evaluation-whitepaper.pdf linked from http://trailofbits.com/2011/08/10/ios-4-security-evaluation/.

3.1.3 Restrictions

Some security-relevant restrictions can be placed upon the user of the iOS device.

3.1.3.1 Disable Installation of Third-Party Apps

Unless necessary, disable **Allow installing apps**. *This is unusual for a general-purpose device.*

While iOS includes features such as Sandboxing that are designed to prevent third-party apps from influencing the integrity of the operating system, they do have the ability to access data stored on the device such as Address Book (until recently), Location Data, or the Photo Library and have the ability to transmit this information.

3.1.3.2 Disable Camera

Disable **Allow use of camera**, if concerns exist about capturing sensitive images.

3.1.3.3 Disable Screen Capture

Disable **Allow screen capture**, if concerns exist about storing screen contents in the Photo Library.

While unlikely, this feature could accidentally (or intentionally) be triggered (by simultaneously pressing the Home and Sleep buttons) and lead to storage of sensitive information outside the intended storage area.

3.1.3.4 Disable or Configure Safari

If Safari can be disabled, uncheck **Allow use of Safari**. *This is very unusual for a general-purpose device.*

If Safari is needed, security-relevant Safari settings can be configured as follows.

3.1.3.4.1 Disable Safari Autofill

Set **Enable autofill** to Unchecked. This prevents storage of some potentially sensitive information by Safari.

3.1.3.4.2 Enable Safari Fraud Warning

Set **Force fraud warning** to Checked. This ensures users are warned when visiting known-fraudulent web sites.

3.1.3.4.3 Enable Safari Pop-up Blocking

Set **Block pop-ups** to Checked.

3.1.3.4.4 Accept Cookies from Visited Sites Only

Set **Accept cookies** to From visited sites.

3.1.3.5 iCloud configuration

Policies regarding the usage of "cloud" storage services continue to evolve, as do the assurances of safety by cloud providers. In general, if there is a need to store potentially sensitive information on the iOS device, then the following restrictions are recommended.

3.1.3.5.1 Disable iCloud Backups

Set **Allow backup** to Unchecked.

3.1.3.5.2 Disable iCloud Document Sync

Set **Allow document sync** to Unchecked.

3.1.3.5.3 Disable iCloud Photo Stream

Set **Allow Photo Stream** to Unchecked.

3.1.3.6 Security and Privacy

The **Security and Privacy** restrictions can control whether the device will send diagnostic data to Apple, whether the device will require the user to encrypt backups, and whether the user can decide to accept

untrusted TLS certificates.

3.1.3.6.1 Disable Sending Diagnostic Data to Apple

Set **Allow diagnositic data to be sent to Apple** to Unchecked, if this presents concerns about inadvertent transmission of sensitive data.

3.1.3.6.2 Disable User's Acceptance of Untrusted TLS Certificates

Set **Allow user to accept untrusted TLS certificates** to Unchecked.

Root CAs trusted by iOS are available at http://support.apple.com/kb/HT5012.

3.1.3.6.3 Force Encrypted Backups

Set **Force encrypted backups** to Checked, to protect device backups if the host later becomes compromised.

3.1.4 Wi-Fi

iOS devices support 802.1X authentication for WPA2 Enterprise networks, and this is strongly recommended. A RADIUS server is required for 802.1X authentication and typically involves the use of public key infrastructure. User education and training is also important, since the user has control over the device's network settings. Section 4.2.4 contains information for users.

DoD Instruction 8420.01, available at http://www.dtic.mil/whs/directives/corres/pdf/842001p.pdf, provides information for DoD entities regarding the configuration and deployment of WiFi networks.

3.1.4.1 Use WPA / WPA2 Enterprise for Wi-Fi Encryption

Using **WPA / WPA2 Enterprise** with **TLS** for authentication is recommended. If TLS support is not available, **PEAP** is the next best choice for authentication. All other authentication protocols are not recommended.

Use **WPA / WPA2** if authentication support is not available. Proxy servers can be configured with WiFi as another layer for providing control of the connection.

3.1.4.2 Disable Auto-Join for Wi-Fi

Ensure that **Auto Join** is disabled for WiFi networks.

Disabling auto join ensures that users are aware of when connections to WiFi networks are being made.

3.1.5 VPN

VPN connectivity obviously depends on an enterprise's available infrastructure, but VPNs which use IPsec are preferred.

Several SSL VPNs are also supported by iOS. Actual VPNs are preferred over SSL VPNs as they are designed to protect systemwide network communications. Note, however, that at this time iOS VPNs cannot be configured to route all traffic through a VPN, and operate in split tunnel mode. This behavior occurs even if **Send All Traffic** is selected as part of any VPN's configuration.

3.1.5.1 Select IPsec (Cisco) or L2TP for Use as VPN

Select **IPsec (Cisco)** or **L2TP** (which also uses IPsec) for use as the Connection Type if possible. Use of hardware tokens is generally preferred over passwords for user authentication.

Apple provides documentation regarding iOS VPN capabilities in the following documents:

- *VPN Server Configuration for iOS Devices*, available at http://help.apple.com/iosdeployment-vpn/
- *iOS: Supported protocols for VPN*, available at http://support.apple.com/kb/HT1288

The following documents provide recommendations for configuring VPNs in an enterprise infrastructure:

- *Guide to IPSec VPNs* (NIST SP 800-77), available at http://csrc.nist.gov/publications/nistpubs/800-77/sp800-77.pdf.
- *Guide to SSL VPNs* (NIST SP 800-113), available at http://csrc.nist.gov/publications/nistpubs/800-113/SP800-113.pdf.

3.1.6 Email

Permitting users to access only enterprise-supported email accounts decreases the risk posed by email-based attacks. It ensures that enterprise-provided countermeasures against email attacks (such as content filters or anti-virus software) can scan email transmitted to the device.

3.1.6.1 Prevent Moving Messages between Mail Accounts

Disable **Allow Move** for all email accounts.

3.1.6.2 Enable SSL for Mail Connections

Ensure **Use SSL** is enabled for all incoming and outgoing email accounts.

3.1.6.3 Enable S/MIME Support for Mail if Needed

Set **Enable S/MIME** to Checked, if encrypted and authenticated email support is needed.
Ensure that transmission of configuration profiles to devices is encrypted and authenticated if S/MIME certificates containing private keys are embedded. The iOS device can also be configured to use an SCEP server to retrieve S/MIME certificates for use with Mail.

3.1.7 Exchange ActiveSync

If your organization employs Microsoft Exchange to manage user accounts and maintain device policies, configuring Exchange ActiveSync will bind the device to the user's Microsoft Exchange account, syncing email, calendars and contacts with the device.

3.1.8 Prevent moving messages between ActiveSync accounts

Disable **Allow Move** for all Exchange ActiveSync accounts.

3.1.9 Allow Mail from this Account Only from the Mail App

Enable **Use Only in Mail** for all Exchange ActiveSync accounts.

3.1.10 Enable SSL for ActiveSync Communications

Ensure **Use SSL** is Checked for all Exchange ActiveSync accounts.

3.1.10.1 Enable S/MIME Support for ActiveSync if Needed

Set **Enable S/MIME** to Checked, if encrypted and authenticated email support is needed.
Ensure that transmission of configuration profiles to devices is encrypted and authenticated if S/MIME certificates containing private keys are embedded. The iOS device can also be configured to use an SCEP server to retrieve S/MIME certificates for use with Mail.

3.1.11 LDAP

3.1.11.1 Enable SSL for LDAP Connections

Ensure **Use SSL** is enabled if using an LDAP service.

3.1.12 CalDav

3.1.12.1 Enable SSL for CalDav Connections

Ensure **Use SSL** is enabled if using a CalDav service.

3.1.13 Subscribed Calendars

3.1.13.1 Enable SSL for Subscribed Calendar Connections

Ensure **Use SSL** is enabled if connecting to calendar subscriptions.

3.1.14 Credentials

If your organization employs any self-signed certificates, embed them in the configuration profile or use SCEP (See Section 3.1.15) to distribute. Note that embeddeding any credentials into the configuration profile introduces the need for encryption during profile deployment as discussed in Section 2.3.1.

3.1.15 SCEP

If your organization will use the Simple Certificate Enrollment Protocol to distribute certificates and configuration profiles, include its settings with the configuration profile. These settings may be handled by MDM products in a manner that is automated, and not tied to individual users.

3.1.15.1 Set a Challenge Password

In the **Challenge** field, enter a strong passphrase to be used as a pre-shared secret for automatic enrollment.

3.1.16 Mobile Device Management

Some behavior of Mobile Device Management (MDM) software can be configured inside a configuration profile. This includes how much information an MDM server can retrieve from a device, whether an MDM server can update profiles remotely, whether an MDM server can remotely wipe a device, and whether an MDM server can reset a device's password. These settings allow for more fine-grained adjustment between enterprise control versus user control of a device. Some MDM products may not permit administrators to disable some of their capability for querying devices.

3.1.16.1 Sign Messages

Set **Sign messages** to Checked.

This setting causes responses generated by the device (in response to commands from the MDM server) to be signed with the device's identity certificate.

3.1.16.2 Check Out When Removed

Set **Check out when removed** to Checked.

This causes the device to send a message to the MDM server whenever the configuration profile is removed.

3.1.16.3 Access Rights for Remote Administrators

The following settings control what an MDM server is permitted to query from an iOS device. For an enterprise-owned, enterprise-controlled device, permitting the enterprise administrator to query as much information as possible is appropriate. Some MDM products may simply include these access rights by default and offer options to retrieve less information from the device.

At the same time, querying all of these types of information may not be appropriate even for some enterprise users, and for enterprises that support BYOD scenarios. The terms of any individual organization's "BYOD Contract" with their users is beyond the scope of this document.

3.1.16.3.1 Allow Remote Query of General Settings

In the **Query device for** section, set **General settings** to Checked.

3.1.16.3.2 Allow Remote Query of Network Settings

In the **Query device for** section, set **Network settings** to Checked.

3.1.16.3.3 Allow Remote Query of Security Settings

In the **Query device for** section, set **Security settings** to Checked.

3.1.16.3.4 Allow Remote Query of Restrictions

In the **Query device for** section, set **Restriction** to Checked.

3.1.16.3.5 Allow Remote Query of Configuration Profiles

In the **Query device for** section, set **Configuration Profiles** to Checked.

3.1.16.3.6 Allow Remote Query of Provisioning Profiles

In the **Query device for** section, set **Provisioning Profiles** to Checked.

3.1.16.3.7 Allow Remote Query of Applications

In the **Query device for** section, set **Applications** to Checked.

3.1.16.3.8 Allow Remote Addition/Removal of Configuration Profiles

In the **Add / Remove** section, set **Configuration Profiles** to Checked.

3.1.16.3.9 Allow Remote Addition/Removal of Provisioning Profiles

In the **Add / Remove** section, set **Provisioning Profiles** to Checked.

This allows an MDM to be able to update profiles remotely.

3.1.16.3.10 Allow or Disallow Remote Change of Device Password

In the **Security** section, set **Change device password** to Unchecked or Checked. This entails a risk decision, though Checked is likely to be appropriate for most scenarios.

Most enterprises are likely well-served by permitting the MDM administrator to remotely send a change password command, in order to allow users with forgotten passcodes to maintain access to their devices. This would also permit an enterprise with appropriate authority, and which needed to overcome the Data Protection feature (such as for forensics purposes) the ability to do so.

At the same time, an attacker who has compromised communications between the device and its MDM server (or the MDM server itself), could maliciously send a password-change command to defeat the data-at-rest protection on the devices. This would depend upon an attacker's physical compromise of the device as well as compromise of TLS communications (or the MDM server itself). This unlikely scenario is described in Schuetz[9].

3.1.16.3.11 Allow Remote Wipe

Set **Remote wipe** to Checked.

This permits an MDM server administrator to remotely wipe an iOS device in the event that it is lost.

Note also that a layered configuration profile approach which involves specific configuration profiles permitting access to specific services (which can be removed by an MDM server), effectively permits selective removal of access to services (and their local data). This can provide a form of remote wipe that is more appropriate with BYOD scenarios that are incompatible with IT staff wiping entire devices.

3.2 Manually-Configured Device Settings

The following security-relevant settings can be manually applied. These settings depend on the user's control of the device, and thus training users can help them make appropriate choices.

3.2.1 Disable Loading of Remote Images, if Practical

To disable the automatic loading of images in email, set **Settings** ▷ **Mail, Contacts, Calendars**▷ **Load Remote Images** ▷ **Off** , if this is practical.

Automatically loading images in email messages can leak usage information to authors of fraudulent email. It can also provide an opportunity for malicious images to exploit any implementation flaws in complex graphics libraries. At the same time, this may also inhibit viewing of images that are useful.

3.2.2 Disable Bluetooth Manually, if Practical

To disable Bluetooth, set **Settings** ▷ **General** ▷ **Bluetooth** ▷ **Off** when practical.

Leaving Bluetooth enabled can expose the presence of an iOS device, although the device provides visual cues when it is in the Bluetooth "discoverable" mode which allows it to pair with other devices. The Bluetooth profiles supported by iOS are described at http://support.apple.com/kb/HT3647.

3.2.3 Disable Wi-Fi, if Practical

If the iOS device is not to be connected to a Wi-Fi network, disable Wi-Fi. Set **Settings** ▷ **General** ▷ **Network** ▷ **Wi-Fi** ▷ **Wi-Fi** ▷ **Off**.

Disabling **Ask to Join Networks** will prevent the phone from automatically associating with previously known (but potentially-spoofed) access points without user interaction, and should be disabled whenever possible. Users should be instructed to use only trusted WiFi networks, as discussed in Section 4.2.4.

3.2.4 Disable Ping Manually

If Ping could spread potentially sensitive information, disable it by setting **Settings** ▷ **General** ▷ **Restrictions** ▷ **Ping** ▷ **Off**.

Ping is Apple's social network for music.

3.2.5 Disable Location Services, if Practical

If the ability of apps and web pages to determine the location of the device poses an unacceptable risk, disable Location Services. Set **Settings** ▷ **General** ▷ **Location Services** ▷ **Off**. Note also that usage of Location Service can be controlled on a per-app basis, at the user's discretion. Given the utility of location information for some apps (such as Maps), user-determined settings may be most practical.

If an application (such as Maps) wishes to use Location Services while being disabled, the user will be prompted to return to **Settings** to enable it.

4. Device Usage and Handling

This chapter provides recommendations on device usage and handling, for both administrators and users.

Section 4.1 provides handling and usage guidance for administrators. These topics include issuing devices, managing and accounting for devices once in users' hands, and effectively educating users on secure device usage.

Section 4.2 provides handling and usage guidance for users, which must be effectively communicated by administrators. These topics include important recommendations such as maintaining physical control of the device, not jailbreaking the device, and preventing connections to untrusted networks. This section closes with suggested usage statements that could be provided to users.

4.1 Handling Guidance for Administrators

If the enterprise is planning to procure and distribute devices to users, administrators should establish procedures for this activity. Some items from this section may not apply to BYOD scenarios, however, such as inventory management and prompt retirement of unsupported devices.

4.1.1 Establish a User Training Program

> Create or make available training resources to educate users about device security issues and organization policies. Ensure that all device users are aware of risks and properly trained to mitigate them.

Security and policy awareness training reduces the risk of user-originated security compromise. This relates closely to any agreements between the user and enterprise regarding device handling, which should be verified for each user prior to their being issued a device, as described in Section 4.1.2.4.

4.1.2 Issuing Devices

This section provides recommendations for enterprises issuing iOS devices.

4.1.2.1 Issue Only Supported Devices

> Ensure that only supported hardware versions are issued. Supported hardware versions are defined as those that can run the latest version of iOS and receive all updates. To determine this, administrators will need to manually note which systems can be updated whenever security updates are provided.

Sometimes only the current version and the previous version of the iPhone or iPad hardware can run all updates. This suggests that IT planners should anticipate a 2 year (or 3 year, at most) refresh cycle for enterprise-purchased devices.

4.1.2.2 Erase and Reset Devices, if Re-issuing

If re-issuing devices, erase them before distributing them to users.

Use the command **Settings** ▷ **General** ▷ **Reset** ▷ **Erase All Content and Settings** to erase a device. Clearing content and settings returns the device to a stable state and prevents accidental exposure of the prior user's data.

4.1.2.3 Update Device-to-User Registration

Establish a system for attributing individual devices to users prior to issuance. This information must be updated every time a device is issued or transferred. Existing inventory tracking systems or MDM software can enable automation of this process.

The following pieces of information from each device can be useful:

- UDID (Unique Device Identifier)
- Serial Number
- IMEI (if equipped with a cellular connectivity)
- Model Number
- Wi-Fi MAC Address
- Bluetooth MAC Address

MDM products may also report this information. This information should be protected accordingly.

4.1.2.4 Verify User Training History

Ensure that users are familiar with the training before receiving a new device, and at regular intervals afterward.

4.1.2.5 Provide Recharging Hardware with Device

Distribute AC power adapters to users when issuing devices and warn users not to connect their devices to unauthorized systems. It may be prudent to distribute additional AC power adapters to remove the temptation to connect the devices to unknown PCs.

Connecting iOS devices to unauthorized systems, even if only intending to recharge the device, presents a security risk. Providing a power adapter, and easy access to replacements and additional adapters, will help combat temptation to connect to other systems. Users should never be left with connecting to a computer as their only option to recharge their device.

4.1.3 Dealing with a Lost or Stolen iOS Device

If an iOS device is reported as lost or stolen, the device should be immediately disabled to prevent unauthorized use or access. The system administrator can issue a remote "Wipe" command to erase all media, data and settings from the device, restoring it to factory settings. Be aware of the circumstances under which

issuing a wipe may not be possible, such as keeping a device in Airplane Mode or simply lacking network connectivity.

4.1.3.1 Establish Procedure for Lost or Stolen iOS devices

Establish and test a procedure to issue a wipe command to erase data from a lost or stolen iOS device. Ensure that users are also aware of their responsibilities to report lost or stolen devices, as documented in Section 4.2.1.2.

Wipe commands can be issued by an MDM server or by Exchange ActiveSync. Users can also initiate remote wipe using iCloud, if the device is enrolled.

4.1.4 Retire Devices Which Cannot Run Latest OS Version

Immediately retire any devices which cannot run the latest iOS version. This requires vigilance on the part of administrators, to monitor when an update is issued but is not supported on older devices.

iOS updates include both security patches as well as new functionality. Ensure that all iOS device hardware provided and managed by the enterprise can always run the latest iOS. For example, all iPhone 3G devices should be immediately retired, because they cannot run iOS 5.

4.1.5 Monitor Devices Using MDM, Especially for Updates

As discussed in Section 2.2, MDM products enable enterprise integration and reporting for iOS devices. Regularly monitor the status of devices using MDM software and respond accordingly. Particularly important is ensuring that the version of iOS is kept up to date, which implies that all available security updates are installed. Some MDM products include the ability to disable access to enterprise resources if devices are not kept up to date or are otherwise not compliant.

4.2 Handling Guidance for Users

User education is one of the strongest tools an organization can use to minimize the risk of security issues. Educating users helps raise awareness of their actions and helps them understand the reasoning behind policy decisions.

This section details physical handling guidance and security policy topics to be reinforced to users through an organization-developed user education program.

4.2.1 Physical Control

Maintain physical control of your iOS device at all times.

All guidance contained in this document depends upon uninterrupted physical control of your iOS device. It is your responsibility to maintain possession of the device.

Never leave your iOS device unattended in an insecure location. An unattended device is at high risk for loss, theft, and other forms of compromise that could violate the confidentiality, integrity, or availability of the device and the information contained therein.

4.2.1.1 Surrendering Physical Control

Learn the proper procedure for relinquishing control of the iOS device to another entity.

There are times when physical control of the iOS device must be surrendered, such as when passing through security or customs inspections. The following are possible methods of mitigating potential loss of personal, financial or company information.

- Before entering security or customs checkpoints, power down the iOS device, remove its SIM card using the SIM eject tool or an unwound paper clip, and place the SIM card in a physically separate location such as a bag or your coat pocket.
- Place the device in a clear, tamper evident bag.
- Ensure passcode is enabled.

Organizations may elect to require all of these steps based on their security policy.

4.2.1.2 Notify Security or Administrative Personnel Upon Loss of Physical Control

Obtain the contact information of your System Security Officer (SSO) for use in reporting the loss of physical control of your iOS device and learn which scenarios require SSO notification.

If there is any suspicion that a device has been accessed by an unauthorized user, report the incident immediately to the appropriate SSO or administrative personnel.

If a device is lost or stolen, the administrator or SSO should be contacted immediately in order to execute the remote wipe procedure through Microsoft Exchange as described in Section 4.1.3, and to create a detailed incident report describing the event.

Even if you lose control of your iOS device for a period of time but regain it later, it should be inspected for signs of physical compromise by system administration or security personnel. If a compromise is suspected, actions should be taken to sanitize or destroy the device, depending on the sensitivity of the data and severity of the situation in which it was compromised.

4.2.1.3 Be Aware of Your Surroundings

Be aware of the danger of "shoulder surfing," which refers to the ability of others to see your entry or viewing of sensitive information on the phone.

Because anyone nearby can potentially view any information displayed on the device, be wary of your environment when viewing any sensitive information, and particularly wary when entering passwords. Due to obvious physical and user interface constraints, password entry is susceptible to shoulder surfing, whether by observation of finger position or brief display of each character on-screen. Some third-party products may be available to mitigate this risk.

4.2.1.4 Follow Procedures for Secure Areas

Learn the proper procedure for handling your iOS device in a secure area.

If your organization has a secure area for talking about confidential information, you should be educated about the risks of bringing your iOS device into those areas. The following policies may be implemented for device security in secure areas:

- Leave iOS devices outside conference rooms.
- Applications that record audio or video must be removed or their use restricted.
- Ensure the camera on the back of the iOS device is blocked (e.g. opaque tape) to prevent photo or video recording.
- Ensure that all iOS devices, if present, are in airplane mode with Wi-Fi turned off. Refer to Chapter 3 for more information.

4.2.2 Do Not Jailbreak or Unlock Your iOS Device

Jailbreaking is a term that describes the process of modifying the iOS device's operating system, often with the goal of running unsigned code or performing unsupported customizations to the operating system. *Unlocking* allows users to operate an iOS device on a cellular network it is not authorized to connect to. Unlocking an iOS device requires a jailbroken iOS device first.

Do not jailbreak or unlock your iOS device. *Doing so makes it much easier for attackers to introduce malicious code onto the device.*

Jailbreaking significantly increases the iOS device's susceptibility to compromise. It disables the enforcement of code signatures, a critical security feature. This enables access to a wide range of software with little accountability and minimal vetting. Jailbreaking tools also typically install and activates services that make the device easier to remotely access, such as SSH.

4.2.3 Install Software Updates When Available

Install software updates as quickly as possible. Updates can be applied over-the-air, and are indicated by a red circle on the "Settings" app. Apply by following **Settings** ▷ **General** ▷ **Software Update**.

Software updates for iOS devices can contain fixes for security vulnerabilities. As these vulnerabilities often become public knowledge when the updates are released, installing updates as soon as they are available is strongly recommended. Supported 3rd party software should not be broken by updates.

Prior to iOS 5, applying iOS updates always required the use of iTunes. Migrating from iOS 4 to iOS 5 also requires the use of iTunes.

4.2.4 Connect Only to Trusted Networks

Do not connect your iOS device to untrusted wireless networks.

Connections to untrusted WiFi networks introduce some risks. Attacks on the iOS device, or eavesdropping on the data it transmits, can occur due to use of such networks. Because the user controls the WiFi settings, he or she must understand the risks associated with untrusted wireless networks and behave responsibly. Some organizations have policies that forbid connection to non-enterprise controlled networks. Other organizations forbid or prevent the use of personal devices on enterprise networks.

4.2.5 Email Accounts

4.2.5.1 Consider Risks of Using Personal Email Accounts

Do not add personal email accounts to your iOS device, unless you are comfortable with (or approved for) the additional risk.

Adding personal email accounts implies that personal, non-company data will be transferred to and stored on the device. This likely violates organizational policy with regard to use of company resources for personal use, but it also increases risk. It increases the risk of your personal information being compromised as a result of an attack against the device, and also increases the risks of company information being compromised as a result of an attack carried out against your personal email account. See the next section for more information about phishing attacks and the motivation for segregating email accounts between different systems.

4.2.5.2 Be Aware of Phishing

Be aware of phishing attempts, including receiving mobile profiles from attackers.

Phishing is a term referring to a fraudulent communication (usually email) pretending to be from a reputable source asking for personal, financial or company information. Adding personal email accounts to your iOS device greatly increases your availability to receive phishing emails, which may accidentally release important information about yourself or your organization. By removing personal email accounts from the device, you are protecting your organization from divulging information through your device to these malicious actors.

4.2.6 Disable Bluetooth if Practical

Disable Bluetooth communication if not necessary.

Disabling Bluetooth reduces the possible attack surface for exploitation, although such vulnerabilities are rare and the iOS over-the-air update process enables rapid patching upon any public disclosure. Bluetooth also permits wireless device discovery and can be used to reveal a limited amount of information from the device. If practical, it is safest to keep Bluetooth disabled. The Bluetooth profiles available on iOS devices are documented at http://support.apple.com/kb/HT3647.

4.2.7 Recharge Device Only Through Approved Methods

Recharge your device by either connecting to an organization-approved system or by using the AC power adapter you received when you were issued your device.

Connecting your iOS device to unknown systems exposes the device to unnecessary risks, including the loss of personal or company information. Syncing only with trusted systems also helps maintain the integrity of your iOS device.

5. Supporting Infrastructure

This chapter contains recommendations for infrastructure elements which support iOS device deployment.

5.1 iTunes

With iOS 5, use of iTunes is no longer a prerequisite for deployment of iOS devices. However, iTunes supports the ability to back up data from iOS devices. If a backup capability is necessary, iTunes deployment in the enterprise may be necessary. Alternatively, data could be backed up in "cloud" services or users could be expected to back up iOS devices on personally-owned systems (as in a BYOD scenario). However, legal concerns may arise regarding the presense of enterprise data on personally-owned systems.

If enterprise deployment of iTunes is planned in order to support the ability to back up iOS devices, it can be configured to improve its security posture. A small number of items are also presented which are of negligible security concern, but may be of interest to administrators who have network bandwidth or deployment concerns.

Apple provides guidance for iTunes deployment in support of iOS at http://help.apple.com/iosdeployment-itunes/.

- Settings for Mac OS X systems, and mechanisms for deploying them, are described at http://support.apple.com/kb/HT2653 and http://support.apple.com/kb/HT3490.
- Settings for Windows systems are described at http://support.apple.com/kb/HT2102.

The following sections reference specific settings but do not provide implementation instructions, which vary by host platform and systems management mechanism. Most settings here also correspond to settings in the GUI, which become "grayed-out" if administratively disabled.

5.1.1 Disable Music Sharing

To prevent the system from sharing music (and potentially other files) over the local network when iTunes is running, set the `disableSharedMusic` key to `true`.

5.1.2 Disable Ping

To prevent users from using Ping to potentially share sensitive information, set the `disablePing` key to `true`.

5.1.3 Disable iTunes Store (if Bandwidth Constrained)

If using the iTunes store is not appropriate due to limited bandwidth, set the `disableMusicStore` key to `YES`.

5.1.4 Disable Radio (if Bandwidth Constrained)

If using streaming audio is not appropriate due to limited bandwidth, set the `disableRadio` key to `YES`.

5.1.5 Use Activation-Only Mode (if Direct Connectivity Unavailable)

iOS 5 permits device activation if network connectivity is available. However, if cellular or trusted WiFi network connectivity is not available to support device activation, iTunes can be put into a special mode to support rapid activation of multiple devices. This is described in http://support.apple.com/kb/HT4335.

6. Deployment Checklist

Configuration Creation and Deployment Items

Section	Action	Result	Notes
2.1	Understand the enforceability of configuration profiles		
2.2.1	Select mobile device management (MDM) software		
2.2.2	Understand capabilities of MDM software		
2.3.1	Ensure configuration profiles are deployed with encryption and authentication, if deploying over-the-air		
2.3.2	Manually deploy configuration profiles using iPCU (unusual)		
2.3.3	Avoid deploying configuration profiles via unauthenticated, unencrypted methods such as email		

Configuration Profile Items

Section	Action	Result	Notes
3.1.1	Disallow removal of configuration profiles, if practical		
3.1.2	Enable passcode on device		
3.1.2	Disallow simple values for passcode		
3.1.2	Set passcode minimum length to 6		
3.1.2	Set complex character minimum length to 1		
3.1.2	Set maximum passcode age to 120 days or longer		
3.1.2	Set passcode auto-lock to 5 minutes or less		
3.1.2	Set passcode grace period to 5 minutes or less		
3.1.2	Set passcode number of failed attempts permitted to 10, or fewer		
3.1.2.2	Understand which files are protected by encryption		
3.1.3.1	Disable installation of third-party apps if possible		
3.1.3.2	Disable camera if appropriate		
3.1.3.3	Disable screen capture if appropriate		
3.1.3.4	Disable (unusual) or configure Safari		

3.1.3.4.1	Configure Safari restriction: disable autofill
3.1.3.4.2	Configure Safari restriction: enable fraud warning
3.1.3.4.3	Configure Safari restriction: block pop-ups
3.1.3.4.4	Configure Safari restrictions: accept cookies From visited sites only
3.1.3.5.1	Disable iCloud backups unless needed
3.1.3.5.2	Disable iCloud document sync unless needed
3.1.3.5.3	Disable iCloud photo stream unless needed
3.1.3.6.1	Disable sending diagnostic data to Apple
3.1.3.6.2	Disable user's acceptance of untrusted TLS certificates
3.1.3.6.3	Force encrypted backups of device data
3.1.4.1	Use WPA / WPA2 Enterprise with TLS for Wi-Fi encryption
3.1.4.2	Disable Auto-join for Wi-Fi networks
3.1.5.1	Select IPsec (Cisco) or L2TP for Use as VPN
3.1.6.1	Disable allow move for email accounts
3.1.6.2	Enable SSL for all incoming email accounts
3.1.6.2	Enable SSL for all outgoing email accounts
3.1.6.3	Enable S/MIME support and add certificates, if needed
3.1.8	Disable allow move for ActiveSync accounts
3.1.9	Prevent outgoing mail from being sent outside of Mail app for Exchange Accounts
3.1.10	Enable SSL for ActiveSync communications
3.1.10.1	Enable S/MIME support and add certificates, if needed
3.1.11.1	Enable SSL for LDAP connections
3.1.12.1	Enable SSL for any CalDav connections
3.1.13.1	Enable SSL for subscribed calendar connections
3.1.16.1	For MDM, enable message signing
3.1.16.2	For MDM, check out when profile is removed
3.1.16.3.1	For MDM, allow remote query of general settings
3.1.16.3.2	For MDM, allow remote query of network settings
3.1.16.3.3	For MDM, allow remote query of security settings
3.1.16.3.4	For MDM, allow remote query of restrictions

3.1.16.3.5	For MDM, allow remote query of configuration profiles
3.1.16.3.6	For MDM, allow remote query of provisioning profiles
3.1.16.3.7	For MDM, allow remote query of applications
3.1.16.3.8	For MDM, allow remote addition/removal of configuration profiles
3.1.16.3.9	For MDM, allow remote addition/removal of provisioning profiles
3.1.16.3.10	For MDM, allow or disallow remote activation of device password change
3.1.16.3.11	Allow remote wipe

Usage and Handling Items

Section	Action	Result	Notes
4.1.1	Establish a user training program		
4.1.2.1	Issue only supported devices		
4.1.2.2	Reset devices prior to any re-issuance		
4.1.2.3	Establish device-to-user registration prior to issuance		
4.1.2.4	Verify user compliance with training programs before issuance		
4.1.2.5	Provide recharging hardware with device and warn users not to connect to unauthorized systems		
4.1.3.1	Establish procedure for lost or stolen devices		
4.1.4	Retire devices which cannot run latest iOS version		
4.1.5	Monitor devices using MDM software, especially for updates		

Supporting Infrastructure Items

Section	Action	Result	Notes
5.1	Determine whether to support backup via iTunes deployment		
5.1.1	If deploying iTunes, disable music sharing		
5.1.2	If deploying iTunes, disable Ping		
5.1.3	If deploying iTunes and bandwidth-constrained, disable iTunes Store		
5.1.4	If deploying iTunes and bandwidth-constrained, disable Radio		

Bibliography

[1] Apple, Inc. *iOS Deployment Guides*, Dec. 2008. http://www.apple.com/iphone/business/resources/.

[2] Apple, Inc. *iPhone User Guide (for iOS 5.0 Software)*, Oct. 2011. http://manuals.info.apple.com/en_US/iphone_user_guide.pdf.

[3] Bédrune, J.-B., and Sigwuld, J. iPhone Data Protection in Depth, May 2011. http://esec-lab.sogeti.com/dotclear/public/publications/11-hitbamsterdam-iphonedataprotection.pdf.

[4] Blazakis, D. The Apple Sandbox, Jan. 2011. http://securityevaluators.com/files/papers/apple-sandbox.pdf.

[5] Dai Zovi, D. iOS 4 Security Evaluation, Aug. 2011. http://trailofbits.files.wordpress.com/2011/08/apple-ios-4-security-evaluation-whitepaper.pdf.

[6] Defence Signals Directorate. *iOS Hardening Configuration Guide*, Jun. 2011. http://www.dsd.gov.au/publications/iOS_Hardening_Guide.pdf.

[7] National Institute of Standards and Technology. *Guidelines on Cell Phone and PDA Security*, Oct. 2008. http://csrc.nist.gov/publications/nistpubs/800-124/SP800-124.pdf.

[8] National Security Agency. *Information Assurance Mitigations Guidance*, Jan. 2012. http://www.nsa.gov/ia/mitigation_guidance/index.shtml.

[9] Schuetz, D. Inside Apple's MDM Black Box, Jan. 2012. https://github.com/intrepidusgroup/imdmtools/blob/master/Presentations/InsideAppleMDM_ShmooCon_2012.pdf.