Australian Government

Department of Defence
Intelligence and Security

PROTECT

iOS Hardening Configuration Guide

FOR iPOD TOUCH, iPHONE AND iPAD RUNNING iOS 5.1 OR HIGHER

March 2012

iOS Hardening Configuration Guide

For iPod Touch, iPhone and iPad devices running iOS 5.1 or higher.

March 2012 (minor update)

About this Guide

This guide provides instructions and techniques for Australian government agencies to harden the security of iOS 5 devices.

Implementing the techniques and settings found in this document can affect system functionality, and may not be appropriate for every user or environment.

However agencies wishing to differ from the mandatory controls specified in this guide must note that the product will no longer fall under the evaluated configuration. In these cases, agencies should seek approval for non-compliance from their agency head and/or accreditation authority to allow for the formal acceptance of the risks involved.

iOS Evaluation

As per the Evaluated Product List, the Defence Signals Directorate (DSD) has found Apple iOS data protection classes A and B to be suitable for downgrading the handling of PROTECTED information to that of Unclassified. This document provides guidance on policy that either must be enforced or is at the agency's discretion.

iOS and the Australian Government Information Security Manual

This guide reflects policy specified in the ISM. Not all ISM requirements can currently be implemented on iOS 5 devices. In these cases, risk mitigation measures are provided (see Appendix E).

Chapter Six provides recommended passcode settings for iOS devices. This advice has been developed based on an assessment of security risks related specifically to iOS 5, and takes precedence over the non-platform specific advice in the ISM.

About the Defence Signals Directorate

As the Commonwealth authority on the security of information, the Defence Signals Directorate provides guidance and other assistance to Australian federal and state agencies on matters relating to the security and integrity of information.

For more information, go to www.dsd.gov.au.

Audience

This guide is for users and administrators of iOS 5 or later devices. These devices include the iPod Touch, iPhone and iPad.

To use this guide, readers should be:

- familiar with basic networking concepts
- an experienced Mac OS X or Windows administrator
- familiar with the Mac OS X or Windows interface.

Parts of this guide refer to features that require the engagement of the technical resources of agency telecommunications carriers, firewall vendors, or Mobile Device Management (MDM) vendors. While every effort has been made to ensure content involving these third party products is correct at the time of writing, agencies should always check with these vendors when planning an implementation.

Additionally, mention of third party products is not a specific endorsement of that vendor over another; they are mentioned as illustrative examples only.

Some instructions in this guide are complex, and if implemented incorrectly could cause serious effects to the device, the network and the agency's security posture. These instructions should only be used by experienced administrators, and should be used in conjunction with thorough testing.

Finally, for further clarification or assistance, Australian Government IT Security Advisors can consult the Defence Signals Directorate by emailing dsd.assist@defence.gov.au or calling the DSD Cyber Hotline on 1300 CYBER1 (1300 292 371).

What is in this Guide

This guide aims to assist in securing iOS 5 devices. It does not attempt to provide comprehensive information about securing computers and servers.

This guide includes the following chapters:

Note: Because Apple periodically releases new versions and updates to its software, images shown in this document may vary from what appears on the screen.

Using this Guide

The following list contains suggestions for using this guide:

- Read the guide in its entirety. Subsequent sections build on information and recommendations discussed in prior sections.
- The instructions in this guide should always be tested in a non-operational environment before deployment. This non-operational environment should simulate, as far as possible, the environment where the device will be deployed.
- This information is intended for mobile devices running iOS 5. Before securely configuring a device, determine what functions that device needs to perform, and apply security configurations to the device or supporting infrastructure where applicable.
- A security checklist is provided in the Appendix to track and record the chosen settings for each security task and note which settings are changed to secure the iOS device. This information can be helpful when developing an agency security standard.

Important: Any deviation from this guide should be evaluated to determine security risks and take measures to monitor or mitigate those risks.

Note: Documentation and advice is periodically updated by both DSD and relevant vendors. DSD recommends that agencies review revised help pages and new editions of guides.

Getting Additional Information

For security-specific information, consult the following:

- *Australian Government Information Security Manual* http://www.dsd.gov.au/infosec/ISM.htm —DSD provides information on securely configuring proprietary and open source software to Australian Government standards. Additional information for Australian government agencies, contractors and IRAP assessors, is available from DSD's "OnSecure" portal https://members.onsecure.gov.au

- *Apple iOS Security* (http://images.apple.com/ipad/business/docs/iOS_Security_May12.pdf) — Apple have released a good high level overview of iOS Security features.

- *NSA security configuration recommendations* (http://www.nsa.gov/ia/_files/os/applemac/Apple_iOS_5_Guide.pdfl) —The US National Security Agency (NSA) has also published a list of security recommendations for iOS 5.

- *NIST Security Configuration Checklists Repository* (http://web.nvd.nist.gov/view/ncp/repository) — is the US National Institute of Standards and Technology (NIST) repository for security configuration checklists.

- *DISA Security Technical Implementation Guide* (http://www.disa.mil/Services/Network-Services/Voice/SBU-Voice/Policy-Guidance-Publications)— is the US Defense Information Systems Agency (DISA) guide for implementing secure government networks. A US Department of Defense (DoD) PKI Certificate is required to access this information.

- *CIS Benchmark and Scoring Tool* (https://benchmarks.cisecurity.org/en-us/?route=downloads.multiform)—The Center for Internet Security benchmark and scoring tool is used to establish CIS benchmarks.

For further information consult the following resources:

- *Apple Product Security website* (www.apple.com/support/security/)—access to security information and resources, including security updates and notifications.

- *Developer documentation* (http://developer.apple.com) Registered developers get access to WWDC session videos and PDF documents. Free registration allowing access to documentation and developer SDK is available.

- *Apple Product Security Mailing Lists website* (http://lists.apple.com/mailman/listinfo/security-announce)—mailing lists for communicating by email with other administrators about security notifications and announcements.

- *iPhone, iPad and iPod Touch manuals* (http://support.apple.com/manuals) —PDF versions of all product documentations.

- *iPhone, iPad and iPod Touch user guides* (http://help.apple.com/iphone, http://help.apple.com/ipad, http://help.apple.com/ipodtouch) — available as HTML5 web applications that work offline on the devices

- *iPhone in Business website* (http://www.apple.com/iphone/business/integration/)—reference point for all enterprise related documentation for iOS integration.

- *Apple Developer Website* (http://developer.apple.com) registration required, contains extensive information on enterprise deployment of iOS devices, developer documentation on APIs and programming techniques for both web based and native iOS applications.

- *iOS Enterprise Deployment Articles* (http://developer.apple.com/library/ios/) – provides a detailed reference on a variety of enterprise deployment themes. These can be found in the iOS Developer Library under the “Networking & Internet” – “Enterprise Deployment” topic.

- *Apple Discussions website* (http://discussions.apple.com)—a way to share questions, knowledge, and advice with other administrators.

- *Apple Mailing Lists website* (http://www.lists.apple.com) — subscribe to mailing lists so agencies can communicate with other administrators using email.

- *Open Source website* (http://developer.apple.com/opensource/)—access to Darwin open source code, developer information and FAQs.

Chapter One
Introduction to Mobile Device Security Architecture

Mobile devices face the same security challenges as traditional desktop computers, but their mobility means they are also exposed to a set of risks quite different to those of a computer in a fixed location.

This chapter provides the planning steps and architecture considerations necessary to set up a secure environment for mobile devices. Much of the content in this chapter is platform agnostic, but some detail is written to specific features available in iOS 5. Not all of these options discussed will be applicable to all environments. Agencies need to take into account their own environment and consider their acceptable level of residual risk.

Assumptions

This chapter makes some basic assumptions regarding the pervasive threat environment:

- at some point, there will be no network connection present
- all radiated communication from the device has the potential to be monitored
- all conventional location, voice and SMS/MMS communications are on an insecure channel[1]

1 Although GSM for example is encrypted on some carrier networks, it is not encrypted on all, and some of the GSM encryption algorithms such as A5/1 on 2G networks are vulnerable to attack with rainbow tables. With moderate resources, it is also feasible to execute a MITM attack against GSM voice and have the MITM tell client devices to drop any GSM encryption.

- certain infrastructure supporting mobile devices can be trusted
- carrier infrastructure cannot always be trusted as secure in all countries.

Device Security off the Network

Once a device is off the data network, then protection of data on the device is determined by how the device implements data protection locally. There can be no referral to a server for policy, or any remote wipe command, if there is no network present.

When off the network, the security of the device is determined by:

- what policy has been cached locally from Exchange ActiveSync (EAS) or Configuration Profiles
- what the security settings set locally on the device are
- the device's cryptographic capabilities
- the correct use of file protection classes and keychain by Apps
- the strength of the device passcode.

Device Security on the Network

The general principle that applies for all data when the device is on a network is that wherever possible, all network traffic should be encrypted, noting that all classified network traffic must be encrypted as per the *Cryptographic Fundamentals* section of the ISM. This is not merely achieved by turning on a Virtual Private Network (VPN) for all traffic. Typically this involves using a mixture of:

- SSL to encrypt connections to specific hosts such as mail servers or management servers that need to be highly reachable
- SSL for any traffic that has sensitive data on it
- a VPN for more general intranet access
- WPA2 with EAP-TLS as a minimum for Wi-Fi security
- 802.1X authentication on Wi-Fi networks combined with Network Access Controls to compartmentalise Wi-Fi access to defined security domains
- a custom, authenticated APN[2] in conjunction with carriers to compartmentalise mobile data traffic to defined security domains
- data at rest encryption on mobile devices and transport security.

Apple Push Notification Service

Many Apps and services associated with iOS devices take advantage of the Apple Push Notification Service (APNS). APNS allows Apps to be sent small notifications, such as updating the badge on an icon, playing an alert tone, or displaying a short text message.

Examples of Apps that may use APNS include push email notification, Mobile Device Management (MDM) servers, and iOS client/server applications that are able to execute in the background (e.g. VoIP Apps, streaming audio Apps, or Apps that need

2 Access Point Name (APN) See the agency telecommunications carrier for more detail.

to be location aware). MDM servers send a request to the MDM agent on the device to “phone home” using APNS. The device and MDM server then exchange XML queries and responses inside an SSL tunnel.

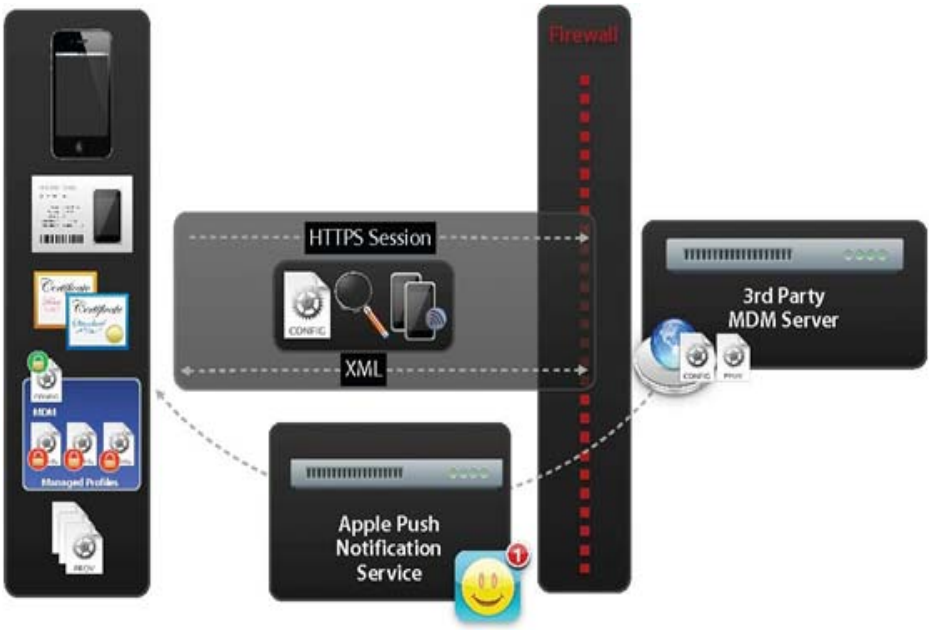


Figure 1: Apple Push Notification Service

It will be necessary to set appropriate firewall rules to enable APNS. Refer to Appendix F for information on ports and services.

Data Roaming

Data roaming generally refers to a process by which a device from a specific carrier’s network can take advantage of the data service on a different carrier. For example a device with a SIM from an Australian carrier, being used in the US on a US carrier’s network takes advantage of the carrier’s data infrastructure. Note that roaming need not be international; in some countries carriers with different coverage areas may allow some data roaming to avoid infrastructure duplication.

There are two main risks associated with data roaming.

- When roaming internationally, there are both implied and actual lower levels of trust with the level of eavesdropping and traffic analysis occurring on the foreign network. As soon as traffic goes international, it is no longer subject to the privacy and consumer protection requirements that apply to purely domestic communications in the host country. It is incorrect to assume that the rights protecting individual’s privacy are uniform internationally.
- If data roaming is switched off for cost management, then the device is “off the grid” for management and monitoring consoles such as EAS, MDM, or iCloud’s “Find My iPhone”. In some cases, private APN data can be preserved across international boundaries because of commercial arrangements between carriers. Note that data costs can still be high.

Apps

One of the major attractions of the iOS platform is the availability of a wide range of Apps and ease of App development. As outlined in DSD's *Strategies to Mitigate Targeted Cyber Intrusions*, DSD recommends that only applications that are required should be installed. There are a number of ways to procure and load applications onto an iOS device.

App Store

The App Store is hosted and curated by Apple, and is focused on mass-market distribution of paid and free applications. These Apps are loaded to a device either over-the-air (OTA) from the App Store itself, or via the iTunes application on the host computer for the iOS device.

Apple maintains discretionary control of curating App Store content, and can remove applications for a variety of reasons. DSD recommends that Apps are tested and approved prior to use within an agency.

Although App Store applications come from a curated environment and the runtime environment the Apps execute in is a relatively hardened one, agencies should assess the risks associated with allowing unrestricted user-initiated installation of Apps. Some risks that need to be considered are:

- the inappropriate use of data protection
- the inappropriate use of transport security
- the inappropriate access of contact list, photos and location information
- registration of URL handlers.

Agencies can manage these risks through discussions with the App developer or through conducting professional penetration testing.

Ad-hoc

Through the use of an Ad-hoc provisioning profile, up to 100 instances of a signed application binary can be installed via iTunes, iPhone Configuration Utility or Apple Configurator.

Ad-hoc applications are locked to a specific set of devices by the provisioning profile. These are most commonly used for beta testing of applications, or where very restricted distribution of a small number of instances of a bespoke application is appropriate.

Enterprise In-House Apps

Agencies with a Dun and Bradstreet Data Universal Numbering System (DUNS)[3] number can apply to become Enterprise developers. This allows the creation and distribution of custom applications and provisioning profiles within an agency for its

[3] DUNS is a unique nine digit number assigned to business by Dun and Bradstreet. See www.dnb.com for more information.

own use, of which the distribution is limited to their employees and contractors (i.e. not to the general public).

Enterprise In-House Apps can be installed using:

- iTunes
- iPhone Configuration Utility
- Apple Configurator
- OTA via a web site
- managed Apps using a MDM server.

Volume Purchase Program (VPP)

The VPP allows businesses to buy app store apps in bulk using a corporate purchasing card. VPP is not available in Australia. For more information on VPP go to http://www.apple.com/business/vpp/

Managed Apps

App Store and Enterprise In-house Applications installations can be triggered via an MDM server; these Apps are called "Managed Apps". Managed Apps can be uninstalled by the MDM server along with any associated data or can be set to uninstall when the MDM profile is removed. Paid App Store Apps can be installed but require VPP which is currently unavailable in Australia.

Web Apps

Safari Mobile has extensive support for HTML5, CSS3 and JavaScript features for Web Apps, including Apps that run full screen and offline. The Product Guides for iPod Touch, iPhone and iPad are all examples of these.

Web Apps are often a useful mechanism to deploy informational applications quickly from a central intranet point; however Mobile Safari on iOS is still subject to the same threats as other browsers.

GSM Voice and SMS/MMS Communication

As noted previously, GSM voice and SMS networks have a number of security weaknesses, where the confidentiality or authenticity of a voice or SMS communication cannot always be ensured, due to both 'Man-in-the-Middle' attacks and the variation in the security features implemented by carriers. As such, voice and SMS communication should generally be considered less secure than methods that implement a chain of trust back into a user's own agency such as SSL tunnelled email.

iMessage in iOS 5 has not currently been evaluated by DSD and should be treated in the same way as MMS.

iTunes

Since the release of iOS 5, iTunes is no longer a requirement for device management. If agencies decide to use iTunes as part of their device management

workflow it can be locked down for use on agency Standard Operating Environments (SOE) via registry keys or XML property lists as detailed here:

http://developer.apple.com/library/ios/#featuredarticles/FA_Deploying_iTunes/Introduction/Introduction.html

Apple IDs

One of the organisational risks that some users express concern about is a perceived need to associate a credit card with every Apple ID. This is actually a misconception, and no association with a credit card is required. The following approaches are recommended at the policy and procedural level.

- For a Bring Your Own Device (BYOD) model, there is generally implied trust that users can continue to install Apps on their own device. Therefore, users may register their existing Apple ID as part of the process of submitting to the agency Acceptable Use Policy (AUP). If users then purchase approved Apps, using their own credit card, they can be reimbursed. This provides one method to control expenditure of agency funds. An MDM console can be used to monitor what applications have been installed.
- For an agency device model, where users are not allowed to install their own Apps, per device Apple IDs are created that are not linked to a credit card. The process for doing this is described here:
  http://support.apple.com/kb/HT2534
  Individual App redemption codes, or store credit can then be gifted to those accounts and installed on the devices from an agency owned computer using iTunes. Note: The end user requires the Apple ID password in order to enable application updates.
- Apple IDs can be optionally used to create free iCloud accounts to facilitate user initiated device location and remote wipe.
- The most restrictive approach is to not reveal the Apple ID password to the end users, and install App Store Apps prior to issue of the device to the end user. However, to update these devices, there is an additional support load, as updates must be completed by IT staff. This approach is recommended for small controlled deployments only.
- Enterprise In-House Apps can be deployed either by iTunes, or OTA to devices, using a secure web site. In all the above cases, an MDM console allows monitoring of which App versions are installed on a device, allowing a management decision as to when updates are required. An MDM console can push a webclip to allow downloading of Enterprise In-House Apps to a fleet of devices.

Siri

Siri provides voice to text services using servers around the world rather than the device. Any dictation performed using Siri must be considered Unclassified. By default Siri can be used from a locked screen to perform actions such as opening emails and reading calendar entries. This behaviour can be disabled via configuration profile while still allowing Siri when unlocked.

Planning Questions

The following questions offer a guide for considerations in implementing policy on the device.

Question	Comments/Selection
How sensitive is the data I am intending to view or store on a mobile device?	In all cases a strong passcode should be set on the device in order to enable data protection. If the data is coming over a network, then it should be secured by some combination of encryption, typically SSL or VPN. If the data is classified refer to the ISM Controls Manual *Cryptographic Fundamentals* section.
Is it appropriate that data gets to the device over a mobile data or wireless network?	Using Apple Configurator on a trusted computer may be an acceptable alternative to transport security.
Do I want users to collaborate using that data in a networked fashion?	If users need to share data over a network, then a secure connection should be in place between the users collaborating.

Question	Comments/Selection
Does my agency want to allow employee owned devices to access some agency data?	Allowing employee owned devices usually has a significant reduction in costs in both procurement and management of mobile device fleets, but introduces a different set of expectations about the level of control an agency can exert over the devices. The balance point between control and flexibility is usually different, and is more consultative in process, than for agency owned devices. An important point to remember is that agencies will need to consider their legislated privacy obligations when determining policy for accessing/wiping employee owned devices.
Does my agency want to allow a mixture of employee and agency owned devices?	If mixed device ownership is allowed, then consideration needs to be given to which, if any, differences in access to information and services are appropriate. In some cases this could involve the use of managed container applications to separate agency data from personal data.
Does my agency need different policies applied to a device depending on whether it is employee or agency owned?	This is a complex issue that requires a mixture of user initiated opt-in Configuration Profiles, MDM managed profiles and pre-installed profiles on a per device basis, appropriate to its context. In some cases this could involve use of managed container applications to risk manage the separation of agency data from personal data.

Question	Comments/Selection
What balance does my agency need to set between the advantages of users being able to install App Store Apps themselves, versus the overhead of managing this centrally?	The more sensitive the data being accessed by a device is, the greater the risks are. Typically a combination of an approved whitelist and monitoring via MDM will mitigate the risks. At high levels of sensitivity, applications may need to be pre-screened, and pre-loaded by IT staff before device issue, or developed in-house and deployed to devices.
Do my agency's acceptable usage policies require explicit education and enforcement?	AUP compliance prior to devices being deployed is critical. AUP education content can be provided as a Web App and Web Clip on the devices for user reference. Other policy controls via EAS, MDM and Configuration profile may be required.
Are all of my devices with one carrier, and agency owned?	If agencies have a single billing arrangement with a carrier, then use of a custom secured APN with a proxy, can assist in enforcing tighter policy controls for devices on the mobile data network. In many cases, a custom APN with EAS and an authenticated, SSL encrypted reverse proxy may be sufficient security for low-level sensitivity data.
Do I need to support devices from multiple carriers and a mix of personal and agency ownership?	A VPN solution may be more appropriate than a custom APN.

Question	Comments/Selection
How can an agency remote wipe devices or secure containers whenever they are reachable on the network?	Remote wipe is usually best managed by a combination of EAS or an MDM console. If agencies do not have a 24/7 service desk capability, then use of Outlook Web Access (OWA) or iCloud can allow user-initiated remote wipes.
To what level does the agency care about its data being monitored and recorded by a third party?	Use of SSL, Wi-Fi encryption, and VPN needs to be considered as per ISM guidelines.
How does an agency develop applications that are customised to its environment?	In-house application development needs to be done in either HTML5/CSS3/Javascript, or native applications code signed with an Enterprise Development Certificate. Native Apps and Web Clips to web applications can be pushed OTA to devices that are under the control of an MDM server.
Does access to my agency information need to be pervasive?	If access to agency data is primarily appropriate on a site or campus, then potentially, focus on Wi-Fi security and limit agency data access, such as EAS PIM, or limited web site access via a reverse SSL proxy.
Do I need to be able to locate devices remotely?	Use of iCloud or an MDM can provide this functionality.
Do I need to digitally sign email (e.g. S/MIME or PGP)?	iOS 5 supports S/MIME, it does not natively support the use of PGP.

Chapter Two
Encryption in iOS

This chapter is provided to help agencies understand the underlying encryption architecture employed in iOS 5 to help make an informed assessment of the risks to Australian government information.

Data Protection

Apple has explained that one of the goals of iOS is to 'keep data safe even when the device is compromised'. However, as will be explained in this chapter, the onus remains largely on the developer as to how much or how little data protection is applied.

For this reason, it is important that an agency wishing to use a particular application understands security features of iOS 5 in order to make a more informed decision as to whether the application meets the security needs of the agency.

This is particularly important, as at the time of this publication, the only native application making full use of data protection within iOS 5 is Mail. It is important for administrators to note that users can still move attachments out of Mail to other Apps that use lower data protection classes. This can happen if installed Apps have registered URL handlers for file types. For PROTECTED devices, agencies should not allow user installation of Apps.

Secrets and Data

Within iOS 5, information stored by Apps can be broadly categorised as either a secret or as data. The term secret can be understood to mean information by which one may get access to data; this can include system credentials, keys and passwords. Data on the other hand, refers to user/application data such as text, pictures, documents and alike.

Accordingly there are two data stores where a developer may choose to store information: the File System and the Key Chain. Developers are encouraged to store secrets within the Key Chain and place more general application data within the File System.

Information stored within either of these stores can be customised to different levels of security, accessibility and portability. Note that it is entirely up to the developer to determine the level of protection applied. This choice is made by the App developer through API calls and the choice of availability as detailed in Table 1.

It is important to note that the default file system protection class on files is ‘None’ (accessible always), while key chain items are set to '...WhenUnlocked' (accessible only when unlocked).

Note: Agencies developing or making use of applications handling sensitive data should take care to investigate how data is handled within their application. They must ensure the appropriate data stores and availability flags (outlined in Table 1) are used to achieve the secure handling of Australian government information.

Classes of Protection

In version 4 of iOS, Apple introduced the concept of protection classes for stored data. This has continued in iOS 5 with some important enhancements.

An application developer has the option of setting the following availability flags with any File System Element or Key Chain entry they create.

Availability	File System Element	Key Chain Entry
When unlocked	…Complete	...WhenUnlocked
While locked	...CompleteUnlessOpen	N/A
After first unlock	...CompleteUntilFirstUserAuthentication	...AfterFirstUnlock
Always	…None	...Always

Table 1: Assignable File System and Key Chain Properties

From Table 1, it is possible to abstract these settings into four standard classes of containers with the following behaviour:

- Class A: Files and credentials within this class can only be read or written when the device is unlocked.
- Class B: Through the use of public key cryptography, files within this class can be written after the device is initially unlocked, and can be read only when unlocked.
- Class C: Files and credentials within this class can be read or written only after the device is initially unlocked.
- Class D: The lowest protection class, files and credentials within this class can be read or written to in all conditions.

iOS 5 Encryption Architecture

Figure 2 illustrates an example where four files exist, each assigned a different class:

- File 1 is of type *Class A*: accessible only when unlocked.
- File 2 is of type *Class B*: can be written to after first unlock, but can only be read when unlocked.
- File 3 is of type *Class C*: accessible after first unlock.
- File 4 is of type *Class D*: accessible always.

Note that while files were used for the purposes of this example, with the exception of Class B, Key Chain entries could just as easily be used in their place.

Similar to the File System, an application's credentials stored within the Key Chain are encrypted using the appropriate Class Key found within the System Keybag (please see the Keybag Section for more information).

However, as illustrated in Table 1, the protection offered by Class B is only available to File System Elements.

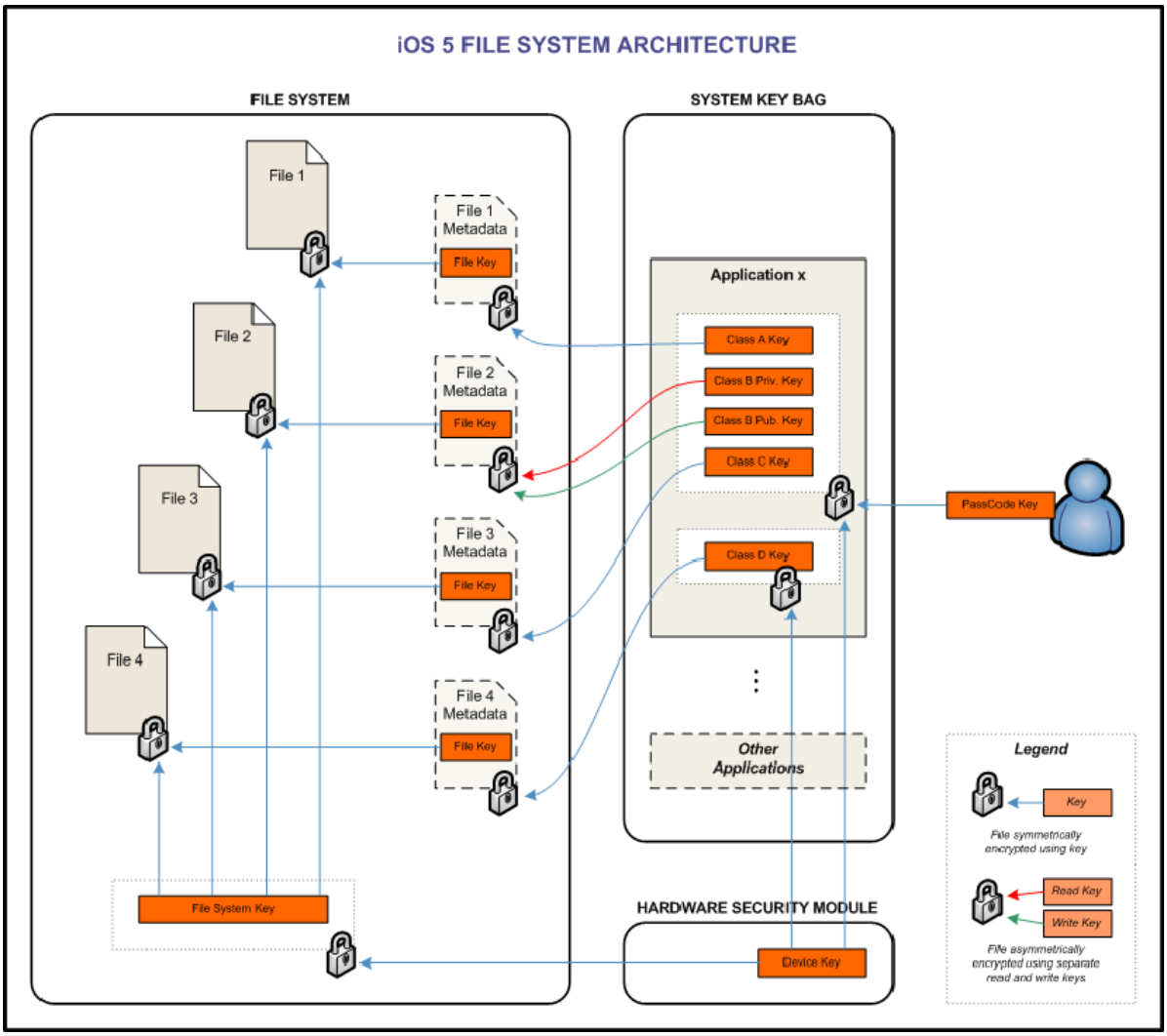


Figure 2: File System Architecture

As can be seen in Figure 2, irrespective of class, each file is encrypted with both a unique File Key and a File System Key.

The File System Key is used to encrypt all data within the device. As it is stored openly its use does not add to the cryptographic security of data, but is instead used to facilitate a remote wipe. Please see the Remote Wipe section for more information regarding this function.

The File Key is stored within the file's metadata, which is itself encrypted by the file's corresponding Class Key. The System Keybag stores all Class Keys within the

device. Please refer to the Keybag section for more information on different types of Keybags used throughout the system.

Upon turning on the device, the Class A, Class B (public and private) and Class C keys are initially inaccessible as they rely on the Passcode Key to be unencrypted.

When the device is first unlocked by the user, through the use of their Passcode, these keys are unencrypted, stored for use and the derived Passcode Key promptly forgotten.

The Device Key is stored within, and never divulged from, the Hardware Security Module (HSM). This acts to encrypt and decrypt files at will using the Device Key. Please refer to the Hardware Security Module Section for more information on this component.

As can be observed, the Class D Key is encrypted using the Device Key. As this decryption process is always available, irrespective of the state of the device, files protected by this Class Key are accessible always.

Finally, when the device re-enters a locked state, the Class A Key and Class B Private Key are forgotten, protecting that data, leaving the Class C Key and Class B Public Key accessible.

Remote Wipe

Remote Wipe is the ability for a network connected iOS device to have the data within the device made inaccessible (enacted by received system command). This is achieved in iOS by erasing the File System Key, which is used by the device to encrypt all user data (as shown in Figure 2). For this reason, once this key is removed, no user data within the device is retrievable.

Hardware Security Module (HSM)

Internal to the device, the HSM is the only means by which iOS can make use of the Device Key. This Device Key is unique to the device and is not exportable using any non-invasive technique.

For this reason (as files encrypted with the Device Key can only be decrypted on the device), the iOS architecture makes itself resistant to off-line attacks. The most significant being a brute-force to exhaust and thus discover the user's Passcode Key.

Keybags

There are three types of Keybags used in iOS: System, Backup and Escrow.

All Keybags are responsible for storing the systems Class Keys, which are in turn used to gain access to individual files or Key Chain entries (as shown in Figure 2).

The System Keybag, shown in Figure 2, is used internally within the device to facilitate the user’s access to the File System and Key Chain.

The Backup Keybag is designed to facilitate backups in a secure manner. This is done by transferring the encrypted contents of the File System, and Key Chain to a remote system along with the Backup Keybag.

The user then has the option to password protect this Keybag; this decision has implications concerning the portability of the Keybag. If the user specifies a password, the Backup Keybag is then encrypted with this password.

Given the password, this data can then be restored to an iOS device (note however that if a developer has specified data 'ThisDeviceOnly', such data will not be made portable). If, however, the user does not set a password, then the Backup Keybag is protected using the Device Key which never leaves the device. Consequently, the Backup Keybag can only be restored to the original device.

The Escrow Keybag is designed to enable a paired device (normally a computer) to gain full access to the device's file system when the device is in a locked state. Pairing in this context refers to connecting the iOS device in an unlocked state (or within 10 seconds of being in an unlocked state) to the other device in question.

An exchange then occurs, where the paired device receives a copy of the iOS device's Escrow Keybag. This Keybag is encrypted using the iOS device's Device Key, thus restricting access when disconnected from the iOS device.

Questions to ask App Developers

1. What is the flow of data throughout the application; source, storage, processing and transmission?
2. Of the data stored on the device, what class of container is it stored in? Note: Application data should always use Class A for storage where possible. If the application needs the ability to write to data in the background, Class B should be considered.
3. Of the data transmitted or received, is it done through a secure means? Agencies can refer developers to the Apple WWDC 2011 presentation, Session 208 on Securing iOS Applications.
4. What system or user credentials are being stored? Are they stored using the Key Chain Class A? If not why not?

References and Further Reading

For more information on the encryption used in iOS, please refer to the following:

- "iPhone data protection in depth" by Jean-Baptiste Bedrune and Jean Sigwald from SOGETI
- "Apple iOS 4 Security Evaluation" by Dino A. Dai Zovi
- "Session 208 Securing iOS Applications", Apple Developer WWDC 2011 Presentation.

Additionally Sessions 204 and 209 from the Developer WWDC 2010 Presentation provide relevant background information on iOS 4.

Verifying Data Protection is Enabled

There are two main methods of verifying that the file system of a device has been configured to support data protection. An MDM console can query the data protection status and report centrally. The user of a device can also validate if data protection is enabled by navigating to Settings → General → Passcode Lock and scrolling to the bottom of the screen. If data protection is enabled, “Data protection is enabled” will be displayed at the bottom of the screen.

Figure 3: iOS device with data protection enabled

Setting a Passcode

The last step in activating data protection is to set a passcode. In most environments enabling a passcode will form part of agency policy, and this will be enforced either over EAS, or via a configuration profile installed on the device. For password policies see chapter six.

Chapter Three
Security Features and Capabilities

This chapter covers mobile device security features, and the enabling technologies for implementing those features under iOS and related infrastructure.

Mobile Device Security Toolbox

When setting up a secure system that uses mobile devices, the security tools and solutions are not on a linear scale, where a solution to a higher security environment is provided by one product alone. Rather, the security posture of the devices can be progressively improved by combinations of capabilities shown below.

Figure 4: Security Features and Capabilities

Security features in iOS

iOS provides a number of features that include:

- management of credentials and passwords with Key Chain
- encryption of data in transit (using DACA[4] and DACP[5])
- encryption of data at rest and in transit (using DACA and DACP)
- digital signatures, certificates and trust services
- randomisation services
- code signed applications.

Enterprise In-House Applications developed for an agency should generally take advantage of these services, rather than re-inventing the same capabilities. More information is available in detail from the Apple Developer web site:

http://developer.apple.com/library/ios/#documentation/Security/Conceptual/Security_Overview/Introduction/Introduction.html

Figure 5: Security Services in iOS

iOS 4.2.1 introduced no-cost "Find My iPhone" functionality for iOS devices using a MobileMe account. In iOS 5 this same functionality is accessible with an iCloud account. The link below contains user level information on how to use the service. Typically this would only be for employee owned devices.

http://www.apple.com/iphone/find-my-iphone-setup/

[4] DSD Approved Cryptographic Algorithm
[5] DSD Approved Cryptographic Protocol

Find My iPhone user interface

Generally, when agency devices are used, the iCloud account would be the same as the Apple ID used to install agency owned Apps, and set up prior to issuing the device. Note that this requires a network connection, location services to be active, and the device to have opted in to the "Find My iPhone" service.

Virtualisation

Agencies may opt to present some agency applications to iOS devices over a network via a Virtual Desktop Infrastructure (VDI).

This works particularly well for users who are "micromobile" i.e. they move about a building or a campus during their work day, and able to take advantage of the relatively high bandwidth of a secure Wi-Fi network, but are not strictly away from the office location. Solutions in this space provide an ability to tune the application UI for a small screen suitable for presenting to mobile devices, rather than merely presenting a remote session to the standard agency desktop resolution. Due to dependency on network performance and differences in screen sizes and input device sizes, VDI based solutions should be thoroughly tested from a usability perspective. This approach also has the advantage that minimal agency data is stored on the device.

Note: Most major authentication token vendors have a soft token available for iOS.

Note: In some cases use of VDI is a classic usability/productivity trade off against security, as the absence of locally cached data means users are not able to be productive when the device is off the network and there is no integration with native applications running locally on the end point device.

Sandboxing

Sandboxing ensures that applications that run on iOS devices are restricted in terms of the resources that they can access. This is enforced by the kernel. For detailed information on the iOS/OS X sandbox see Dion Blazikis's paper "The Apple Sandbox"[6] or Vincenzo Iozzo's presentation "A Sandbox Odyssey"[7].

[6] http://securityevaluators.com/files/papers/apple-sandbox.pdf
[7] http://prezi.com/lxljhvzem6js/a-sandbox-odyssey-infiltrate-2012/

Managed Container Applications

For employee owned devices, or in cases where the agency requires greater protection of their contact and calendar information a third party solution may be required. A number of solutions can be used to provide additional levels of separation and policy enforcement for email, calendar and contact data managed by dedicated servers.

There is usually a usability/security trade off in the configuration, with custom solutions having a lower level of integration with other Apps on the device (e.g. it may not be possible to take a photo with the device's camera, and then send via email using the third party email client).

Note: Currently no third party managed container applications have been evaluated by DSD.

Content Filtering

Access to intranet sites and some mail, contact or calendar data can be achieved via reverse proxies and content filters. There are multiple solutions in this space.

EAS filtering products can be used to ensure email sent to Exchange ActiveSync devices have appropriate privacy markings for the classification the device is approved to by an agency. This approach can allow for an asymmetric strategy - mobile devices only receive email content at a classification appropriate to the device, as well as having policy and controls applied to the email content.

In this scenario, the agency's Wide Area Network (WAN) security domain is not extended out to the mobile device, and there is no need to lower the classification of the agency WAN. Such solutions can be used to redact specific content patterns from emails sent via EAS, for example, to scrub credit card numbers from all emails synced to mobile devices. This class of tools can also facilitate correct protective marking of email coming from mobile devices without direct on-device support for Australian government marking standards. For further information see the ISM section on *Content Filtering*.

Capability	Enablers	Comment
Remote Wipe	MDM, EAS, Apple Push Notification Service (APNS), Find My iPhone	
Proxy	Custom APN, VPN	iOS 5 does not implement a global proxy setting. A proxy can be set on a custom APN and a VPN session.
Firewall	Firewall on custom APN, Firewall on Wireless network	iOS 5 does not implement a local firewall. This is significantly mitigated by the runtime environment.
Force Device Settings	iPCU, Apple Configurator and MDM	Enterprise Deployment Guide lists an XML schema, this can be used to generate and sign profiles from custom scripts. iPCU is an easy to use GUI tool to generate the XML, but CA integration requires signing with OpenSSL tools.
Multi-factor Authentication	SSL CA infrastructure, DNS, RSA or CryptoCard (VPN Only), Smartcard (Requires third party software)	Depending on the agency's security posture, device certificates or soft tokens may be considered as a second factor of authentication.
OTA Configuration Profile (pull)	SSL CA infrastructure, DNS, Web Service, Directory Service	Externally sign and encrypt profiles, do not sign with iPCU.

Capability	Enablers	Comment
OTA Configuration and Provisioning Profiles (push)	Enterprise Developer Agreement, 3rd Party MDM appliance, Apple Configurator, SSL CA infrastructure, DNS, Directory Services, APNS	MDM should be tied into CA and Directory Services.
Mobile Device Monitoring	Enterprise Developer Agreement, 3rd Party MDM appliance, CA infrastructure, DNS, Directory Services, APNS	MDM should be tied into CA and Directory Services.
Mobile Device Management	Enterprise Developer Agreement, 3rd Party MDM appliance, CA infrastructure, DNS, Directory Services, APNS	MDM should be tied into CA and Directory Services.
Remote Application Deployment	Enterprise Developer Agreement, Web Server, 3rd Party MDM appliance (optional), APNS (optional)	Only Enterprise In-House Apps can be deployed OTA.
Home screen		Set Home screen to “If found return to PO BOX XXXX”. This could also be done with a Picture Frame Album.

Chapter Four
Deploying iOS Devices

There are a number of options open to administrators when deploying iOS devices.

iOS has a number of features that are aimed at helping administrators deploy iOS devices in agencies. Apple distributes two important tools to help administrators manage enterprise deployment: iPhone Configuration Utility (iPCU) and Apple Configurator. These tools each have important specialised uses and it is important for administrators to understand their function when planning an enterprise deployment.

	iPCU + iTunes	Apple Configurator
Platforms supported	Mac / Windows	Mac only
Profile creation	Can create configuration profiles	Can create configuration profiles
Install profiles	One device at a time	Many devices at a time
Activate	One device at a time	Many devices at a time
Name device	One device at a time	Many devices at a time
Update iOS	One device at a time	Many devices at a time
Install Apps	One device at a time	Many devices at a time
Backup/Restore	One device at a time	Many devices at a time

Table 2: Deployment Options

There are also many circumstances that require administrators to sanitise devices for deployment or when returning an employee owned device.

This chapter provides an overview of the tools used to manage iOS deployments and advice on device sanitisation procedures.

Configuration Profiles

Both iPCU and Apple Configurator use configuration profiles for iOS deployment. Configuration profiles are XML formatted plist files that contain device settings, security policies and restrictions. An administrator may use a configuration profile to:

- set passcode policy on a device
- set restrictions (such as disabling use of YouTube or Siri)
- configure wireless networks
- configure VPN
- configure email
- install X.509 certificates
- set a Mobile Device Management (MDM) server

These are only a few examples of possible configuration options; please see the iOS Configuration Profile Reference (http://developer.apple.com/library/ios/featuredarticles/iPhoneConfigurationProfileRef/iPhoneConfigurationProfileRef.pdf) for more information.

Note: Configuration profiles are not encrypted. Credentials that are stored in configuration profiles are available to anyone who has access to the files. Passwords may be in clear text or base64 encoded. Some of the credentials that could be in a configuration profile include:

- Wi-Fi passwords
- VPN Shared secrets
- email usernames/passwords
- ActiveSync usernames/passwords
- LDAP usernames/passwords
- CalDAV usernames/passwords
- CardDav usernames/password
- Subscribed Calendars usernames/passwords
- SCEP pre-shared secrets.

Care must be taken to make sure that these files are stored appropriately and not improperly accessed.

Provisioning Profiles

Provisioning profiles allow custom applications to be run on iOS devices. They are often used in the following ways:

- to allow developers to test applications on their devices
- to allow organisations to distribute applications directly to their employees.

To obtain an enterprise distribution provisioning profile, an agency must join the Apple Developer Enterprise Program. More information about the iOS Developer Enterprise Program can be found at:

http://developer.apple.com/programs/ios/enterprise/

If the enterprise program login is compromised an adversary could install malicious applications on users' iOS devices.

iPhone Configuration Utility

iPhone Configuration Utility (iPCU) allows administrators to create and install configuration profiles, install provisioning profiles and examine device information. While iPCU can be used to install configuration profiles on devices, it may only do so on one device at a time. If administrators are planning a small deployment, they may find it simpler to deploy devices one at a time using iPCU.

Installing iOS prior to deployment

iPCU cannot install iOS on devices. There are three ways to install iOS on devices:

- using iTunes
- using iOS 5 OTA update
- using Xcode Organizer.

Activating devices

In a small deployment using iPCU, devices must first be manually activated. If an Internet connection is available to the device, then activation can take place on the device itself. For iPhones or iPads with a mobile data connection, activation can take place using the device's mobile carrier. For iPod and Wi-Fi only iPad, an internet connected wireless network must be available to perform activation from the device. If no internet connection is available to the device, activation must be performed by connecting the iOS device to an internet connected host running iTunes.

If changes must be made to firewall rules, refer to Appendix F for information concerning hostnames and ports of activation servers.

Installing Configuration and Provisioning Profiles

After an unlocked iOS device has been connected to a host computer running iPCU, it is possible to both install and inspect profiles on a device. Configuration profiles that were created using iPCU will be signed by iPhone Configuration Utility.

In the same way, provisioning profiles can be both viewed and installed using iPCU.

iOS Updates

Device users may perform an iOS update by either accepting an OTA update from Apple, or by performing an update using iTunes.

For an update to be delivered via a corporate Wi-Fi network, it may be necessary to adjust firewall rules. Please refer to Appendix F for details.

References and Further Reading

Please refer to the following publication for additional information on iPCU:

- iPad: http://www.apple.com/ipad/business/resources
- iPhone: http://www.apple.com/iphone/business/resources

Apple Configurator

Apple Configurator allows administrators to set up and deploy groups of similarly configured iOS devices. Administrators may use Apple Configurator to:

- activate and name groups of new devices
- update iOS on groups of devices
- install Apps and Configuration Profiles on groups of devices

- backup/restore devices
- retrieve documents

It is suggested that administrators use Apple Configurator with an MDM for large deployments of iOS devices.

A key feature of Apple Configurator is the ability to place devices into what is called **"Supervised"** mode, which changes the way devices are able to pair with hosts. A user with a **Supervised** device will not be able to pair their iOS device with iTunes on their PC. Some of the effects on users will include:

- being unable to sync music or media from their computer running iTunes to their iOS device
- being unable to install Apps on their device using iTunes
- being unable to backup their device using iTunes
- increased difficulty in jailbreaking their device
- not being notified when changes are made to their devices' configuration
- finding their administrator has placed a message on their device locked screen.

Though it may not be appropriate to use **Supervised** mode in a BYOD model, there are reasons why **Supervised** mode is desirable for agency owned devices.

- Sensitive data on each device is better protected. Users cannot sync or backup their device contents to their home computer. iOS forensic recovery utilities may not be able to recover data from the device without a jailbreak.
- Users cannot easily sidestep restrictions. The only viable way to bypass restrictions is to erase the device.

Devices not configured as **Supervised Devices** are referred to as **Unsupervised Devices**.

Note: PROTECTED devices must use **Supervised** mode.

Supervisory Host Identity Certificate

Normally, an unlocked iOS device is able to pair with any host running iTunes (or supporting the lockdown protocol). When an iOS device is set to **Supervised** mode, it authenticates with a host using the "Supervisory Host Identity Certificate". The supervised device will then only pair with a host running Apple Configurator with the correct Supervisory Host Identity Certificate. Ordinary pairing with iTunes is not possible with any other hosts. On a Mac host running Apple Configurator, the Supervisory Host Identity Certificate is stored in the login keychain.

While supervised devices are unable to establish new trust relationships with iTunes hosts, a trust relationship will be formed between devices and the Apple Configurator host. A record of this trust relationship is stored in Escrow Keybag files, which on Mac OS X are located at /var/db/lockdown.

Note: Escrow Keybag files in this directory should be protected in a similar manner to private keys.

Installing iOS

A key feature of Apple Configurator is its ability to install iOS on many devices concurrently. Additionally, varied device platforms (iPhone, iPad, iPod) can all be simultaneously connected. Apple Configurator will seamlessly download iOS for all supported device platforms when there is an internet connection available. A maximum of 30 devices can be connected concurrently for installation.

Activating devices

Apple Configurator will attempt to automatically activate all connected devices after operating system installation. It is important for administrators to note that iPhones and iPads require a SIM for activation. If the SIM has a passcode lock, automatic activation will be unsuccessful.

Installing Configuration Profiles

Apple Configurator may be used both to install configuration profiles and to create new configuration profiles. These profiles can be installed on devices in bulk when initially preparing devices for deployment. As an example, this may be used to initially roll out a trust profile for an agency MDM server.

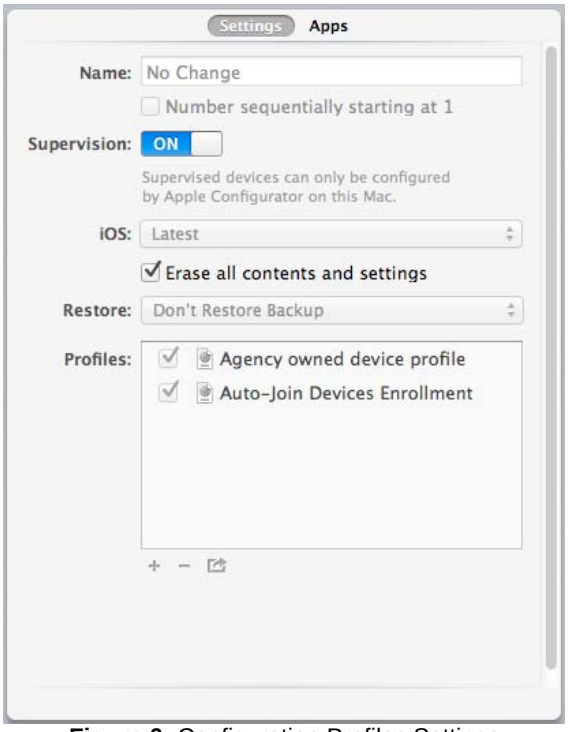


Figure 6: Configuration Profiles Settings

iOS Updates

There are two methods for Apple Configurator deployed devices to receive iOS updates. Devices with an internet connection will prompt users to install OTA

updates. Alternatively, users can return their devices to have them updated using Apple Configurator.

References and Further Reading

Please refer to the following publication for additional information on Apple Configurator:

- Apple Configurator User Guide: http://help.apple.com/configurator/mac/1.0/

Device Sanitisation

Administrators should clean and re-provision devices for the following reasons:

- to sanitise a returned iOS device for re-issue
- to sanitise an employee owned iOS device before provisioning
- to sanitise a deployed employee-owned iOS device prior to the employee leaving
- to break all device-to-host trust relationships and invalidate old Escrow Keybag files.

Breaking the device-to-host trust relationship

When an iOS device pairs with a host, a trust relationship is formed. In many cases an administrator may want to erase an iOS device and break all the established host trust relationships that a device has previously created. The most reliable method to break all established relationships is to restore the iOS device firmware using what is commonly known as “Device Firmware Upgrade” mode (DFU mode).

Please note that restoring a device in this way will also erase all data and settings on the device. The DFU mode restoration can be performed from a host that has no established trust relationship with the device, and the device passcode is not required.

DFU Mode Restoration

To perform an iOS firmware restoration follow this procedure:

1. Connect the iOS device to the host PC running iTunes
2. If iTunes is unable to pair with the iOS device, please clear any error dialog boxes
3. Press and hold both the Sleep/Wake and Home buttons on the iOS device for ten seconds
4. Release the Sleep/Wake button, and continue to hold the home button
5. Release the home button after iTunes generates the following dialog box:

6. After clicking OK, click the “Restore” button to begin installing iOS.

Sanitising an iOS device for re-issue

If an agency-owned device is returned for re-issue to another employee, it should be cleaned by performing a DFU mode restoration. One reason that this is important is so that the previous iOS device owner cannot take advantage of any old device-host trust relationships to retrieve data from the device. By performing the DFU mode restoration the old trust relationships are broken.

Before an iOS device is re-provisioned for enterprise use, it is recommended to perform a DFU mode restoration. This will ensure that the device is in a known state.

Sanitising employee owned iOS devices

Employee owned iOS devices are making their way into government agencies, creating a new set of challenges for administrators. Some of the challenges being faced include:

- jailbroken devices running untrusted code
- jailbroken devices being able to bypass all security protection (including 3rd party managed containers)
- unpatched iOS devices that are vulnerable to exploitation
- devices previously configured with conflicting settings and/or configuration profiles.

Older versions of iOS

Each revision of iOS includes many security related fixes. If left unpatched, iOS devices could be exploited remotely, risking both employee’s personal information and the security of the corporate network.

The agency acceptable use policy should require users to install iOS updates as they become available.

Jailbroken employee owned devices

Jailbroken devices allow users to install applications on their iOS devices from outside of Apple’s App Store. Jailbreaking carries with it a number of negative side effects that impact the security of the corporate network and the confidentiality of data stored on a device.

- Jailbreaking disables application code signing checks. The iOS code signing check helps to prevent malware executing on a device. Removing this check makes exploitation easier and more likely.
- Jailbreaking may disable or break important security features such as Address Space Layout Randomisation (ASLR) and application sandboxing. ASLR increases the difficulty of successful exploitation of vulnerability. Malware on a Jailbroken device would not be constrained by the application sandbox.

- Jailbreaking relies upon serious unpatched operating system vulnerabilities.
- Jailbroken devices should be assumed to be untrusted.

Administrators should not allow employee owned jailbroken iOS devices to be provisioned on the corporate network.

For these reasons it is important to ensure that devices are sanitised prior to deployment.

Sanitisation prior to deployment

When considering how to sanitise employee-owned iOS devices for enterprise deployment, it is important to take into account the data that employees already have on their devices. Employees may have expectations about how they will be able to use their devices and the effect of enterprise deployment on their device. As an example, an employee might expect their iPhone's contact list to be preserved after deployment. If the device is erased using DFU mode this will not be the case.

If an employee's personal data is to be preserved, the following procedure may be performed prior to enterprise deployment:

1. Take a backup of device
2. Perform DFU mode restore
3. Restore a backup to device
4. Delete backup from host
5. Provision and deploy the device as per MDM instructions

This will clear the existing host trust relationships on the device, but will preserve the employee's data. When following this procedure agencies must consider their legal responsibilities to protect the privacy of their users' data.

If there is no need to preserve an employee's personal data on a device, agencies should simply perform a DFU mode restore.

Sanitisation for departing Employees

When an employee departs an agency or no longer requires iOS device connection to the agency network, it is important to remove existing host trust relationships from the employee's iOS device. On return, agency owned devices should be sanitised by performing a DFU mode restore as described previously. Employees should be made aware that agency-owned devices will be sanitised upon return.

In a BYOD model, the following procedure is suggested for departing employees:

1. Remove MDM profile from iOS device, which will:
    - remove the corporate mail account installed by the MDM
    - remove any Apps which have been installed by the MDM which will also remove any associated data
2. Take backup of device
3. Perform DFU mode restore
4. Restore backup to device

5. Erase backup files from host

This will remove any trust relationships established between the iOS device and any agency computers. If the employee does not return their iOS device prior to departing, it may be necessary to use the MDM remote wipe function. Employees should be made aware of this fact in an agency acceptable use policy.

Chapter Five
Suggested Policies

This chapter lists suggested policies in graduated levels of response, applied to iOS devices at varying security classifications. The agency's Information Technology Security Advisor should be consulted for the specific usage scenarios for a deployment.

If iOS devices are being considered for use at classifications above PROTECTED, agencies must undertake a risk assessment following the guidance in the ISM as well as their own agency security policies and determine mitigation procedures and policy. Agencies must also obtain appropriate approval for any non-compliance in accordance with the ISM.

		COMPLIANCE	
Feature	**Unclassified**	**Unclassified (DLM[8])**	**Protected**
Hardware Crypto iOS Devices	Agency's decision	Recommended	Must
BYOD (Bring Your Own Device)	Agency's decision	May be possible (MDM opt-in for AUP agreement and enforcement recommended). See ISM section on *Mobile Devices*	May be possible (MDM opt-in for AUP agreement and enforcement recommended) See ISM section on *Mobile Devices*
Passcode	Must	Must	Must
iTunes Account	Personal or Agency	Personal or Agency	Personal or Agency

[8] DLM: Dissemination Limiting Marker

Feature	Unclassified	COMPLIANCE Unclassified (DLM[8])	Protected
Sync to Content/Sync to iTunes Account.	Yes, if Personal iTunes	Generally no	Generally no
Home Computer backup enforcement	Stated in agency usage policy	Stated in agency usage policy	Stated in agency usage policy
iCloud	Agencies need to assess the risk in their own situation.	Agencies need to assess the risk in their own situation.	No syncing documents and data, No Backup. iTunes Purchases and iTunes Match at Agency discretion.
User ability to install applications	Agencies need to assess the risk in their own situation.	Agency approved applications only. Recommend agency Apple Id. Consider MDM enforced Agency Store Apps whitelist.	Agency approved applications only. Recommend agency Apple Id. MDM enforced Agency Store Apps whitelist.
EAS	Recommended if Exchange or Lotus is used for agency email.	Recommended if Exchange or Lotus is used for agency email. Second factor of authentication using a certificate is preferred.	Possible, with certificate authentication. For some agencies a dedicated mail container or VDI for email access may be preferable.

Feature	Unclassified	COMPLIANCE Unclassified (DLM[8])	Protected
EAS Filtering	Should be used if mobile device security domain is lower classification than intranet security domain.	Should be used if mobile device security domain is lower classification than intranet security domain.	Should be used if mobile device security domain is lower classification than intranet security domain.
Email secured independently of device passcode	Use a dedicated third party mail container.	Use a dedicated third party mail container.	Use a dedicated third party mail container.
MDM	Optional depending on role of device/scale of deployment.	Optional depending on role of device or scale of deployment. Recommended if BYOD model used.	Recommended
Custom APN for 3G data	Optional	Recommended	Recommended
VPN-on-Demand	Optional depending on role	Recommended	Recommended
SSL Reverse Proxy	Optional depending on role	Optional depending on role of device/ scale of deployment	VPN-On-Demand recommended

Feature	Unclassified	COMPLIANCE Unclassified (DLM[8])	Protected
CA Infrastructure	Optional depending on role	Recommended	Required

Chapter Six
Recommended Device Profile Settings

This chapter lists the profile settings that should typically be used when an iOS device is used on an Australian government network.

Note: If profiles are not being pushed by an MDM solution, the correct technique with Configuration Profiles is bundling the payloads in a way that:

- Profiles pulled to the device, bundle restrictions with authentication, so if the profile is removed, all access to agency resources is removed.
- If an MDM is used, the MDM master profile is always removable, but if it is removed all managed profiles are lost as well.

Pre-loaded Configuration Profiles and MDM managed profiles can be mixed on devices, but the MDM server cannot remove the profiles manually installed on the device.

The following settings are a baseline for use on PROTECTED networks. Agency discretion can be varied to be more restrictive if required by local requirements, or lowered at lower classifications in accordance with ISM policy. Where a profile setting is not discussed below, agencies should examine their own particular technical and policy needs. iPhone Configuration Utility and Apple Configurator can be used to view the full range of profile setting that can be deployed.

General (non-Managed Profiles only):

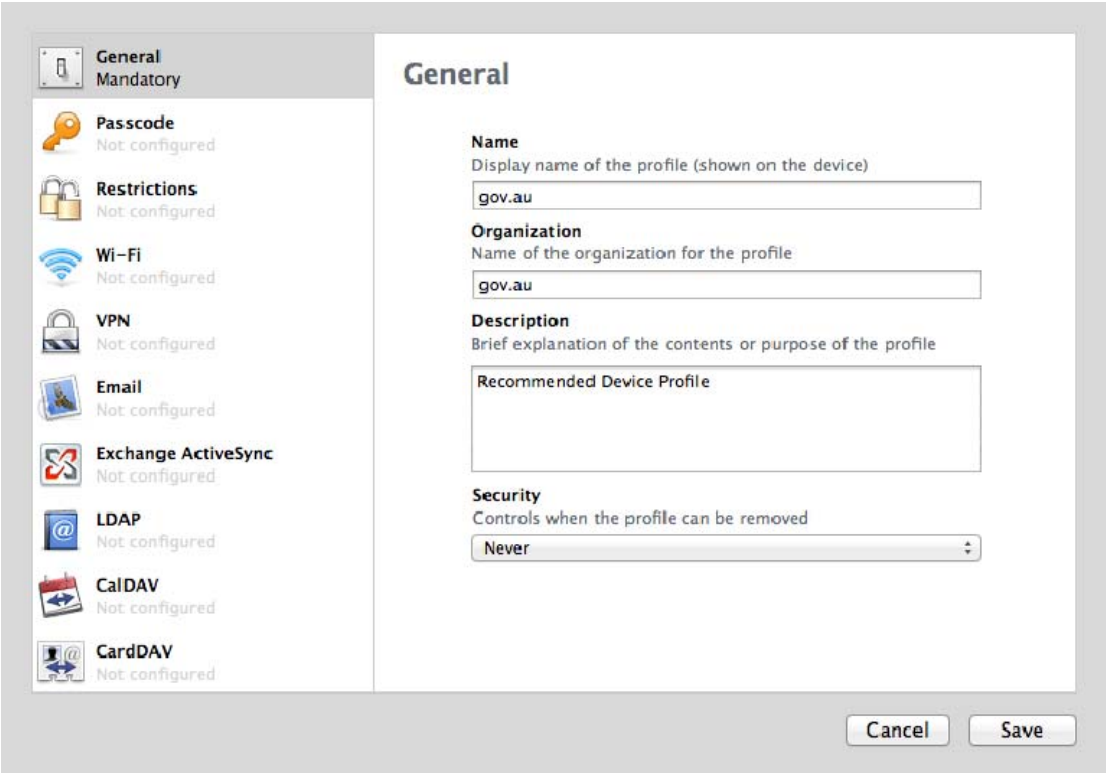


Figure 7: General Settings

- Profile Security should be “Remove Always” if setting is for convenience for users that does not contain any sensitive data (e.g. a subscribed calendar of Australian public holidays). Opt-In MDM profiles would usually fit into this category as well.
- Profile security would usually be “Remove with Passcode” for profiles that IT staff can remove temporarily. Generally users would not receive the passcode to such profiles.
- Most profiles that are not MDM managed would be set to “Never”. The Passcode policy profile, if used, should be set to “Never”.

Passcode (can be set via EAS depending on version, OR Configuration Profile):

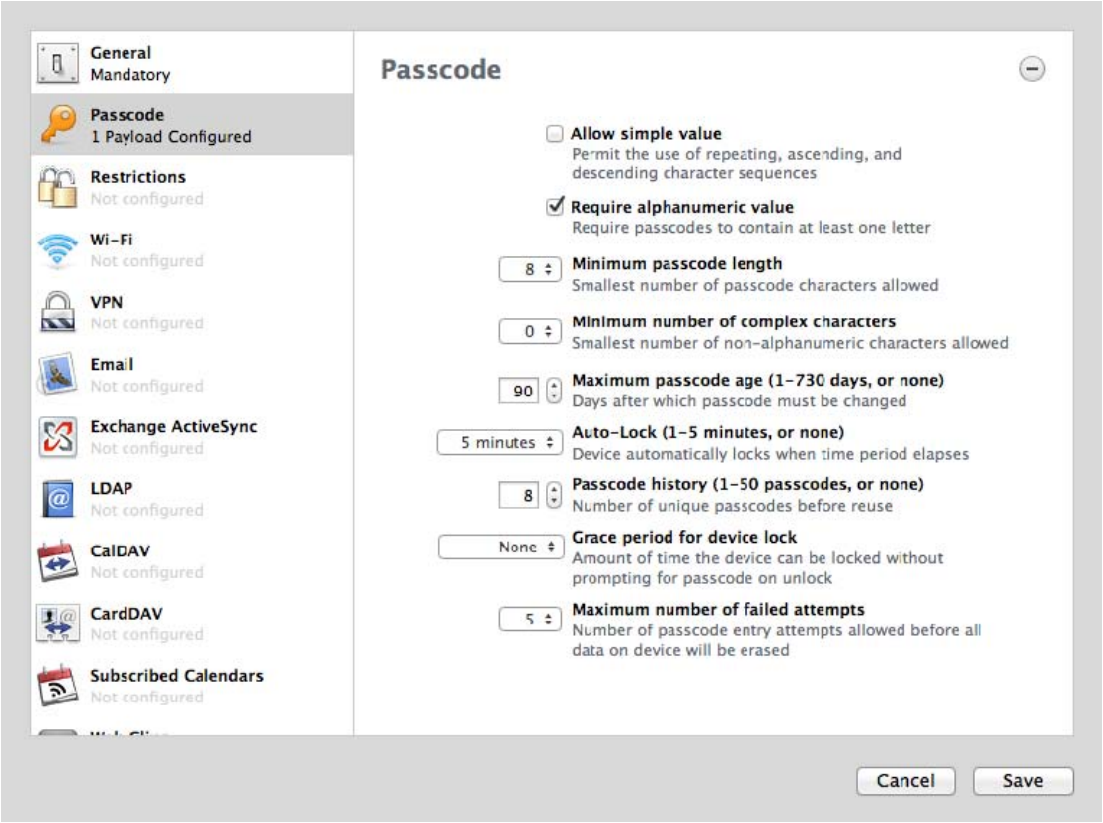


Figure 8: Passcode Payload

- a maximum passcode length of 90 days
- do not allow simple value
- require alphanumeric passcode
- minimum of eight characters
- auto-lock of five minutes (Note: Current maximum allowed time on iOS)
- history of eight passwords
- immediate device lock
- auto-wipe on five failed attempts.

Depending on the EAS version, only some of the above may be set by the EAS Server and a configuration profile would be required.

Restrictions:

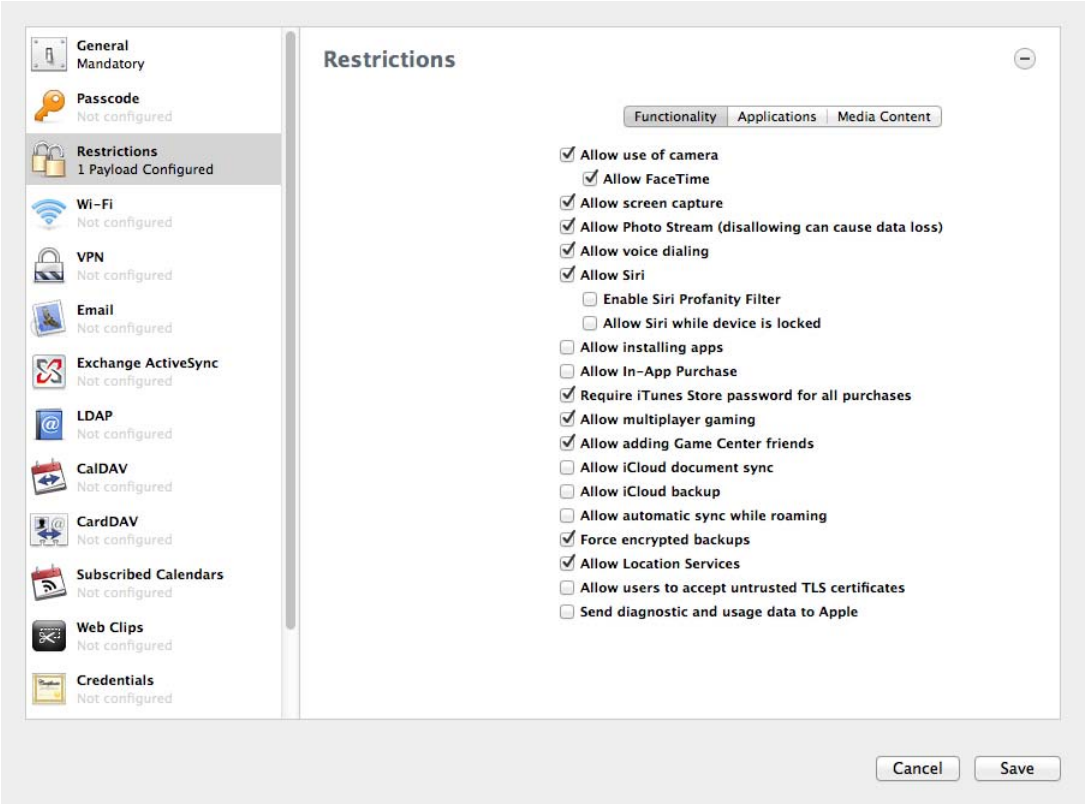


Figure 9: Restrictions Payload

- allow use of Camera: up to agency
- allow use of FaceTime: up to agency
- allow screen capture: up to agency
- allow voice dialling: on
- allow Siri (Siri utilises servers in various locations and all uses of Siri dictation must be treated as Unclassified.)
- allow Siri while device is locked: off
- require iTunes Store password for all purchases
- allow multiplayer gaming: up to agency
- allow adding Game Center friends: up to agency
- allow installing Apps: Recommend off at PROTECTED. Potentially on as an exception at lower levels, as per discussion and mitigation measure noted previously.
- allow in-App purchase: off if App installation off, potentially on if user-installed Apps allowed
- allow automatic sync while roaming: usually off
- force encrypted backups
- allow Location Services
- do not allow users to accept untrusted TLS certificates.

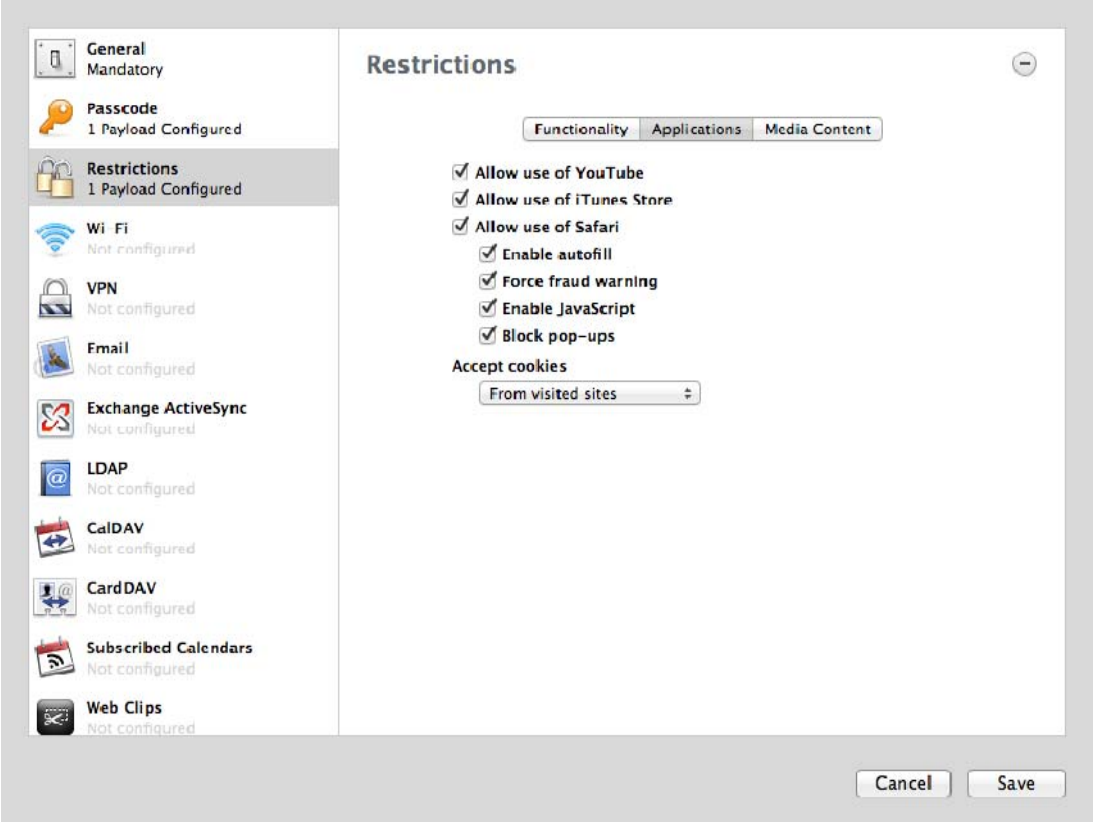


Figure 10: Restrictions Payload

- allow use of YouTube: as per agency policy
- allow use of iTunes Music Store: as per agency policy
- allow use of Safari: enable autofill, force fraud warning, enable JavaScript, block popups.

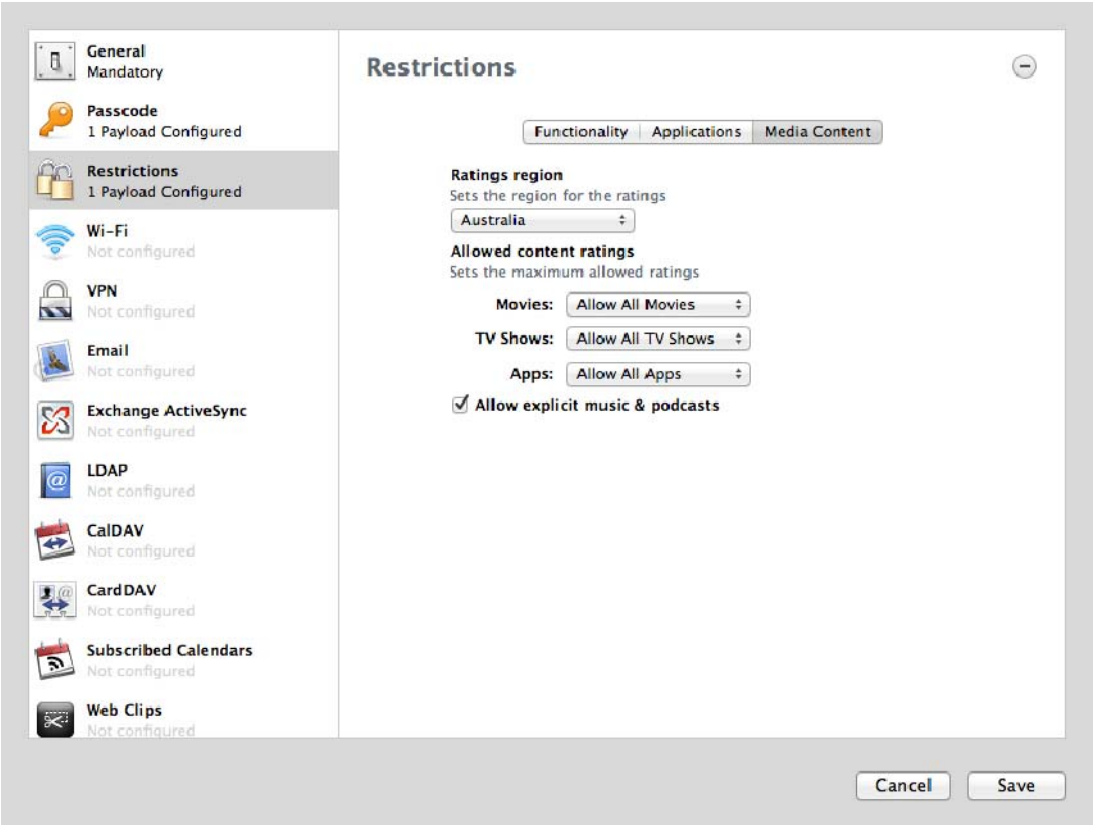


Figure 11: Restrictions Payload

- allow use of explicit music and podcasts: usually off, as per agency policy
- ratings Region: Australia
- allowed content ratings: up to agency policy.

Wi-Fi:

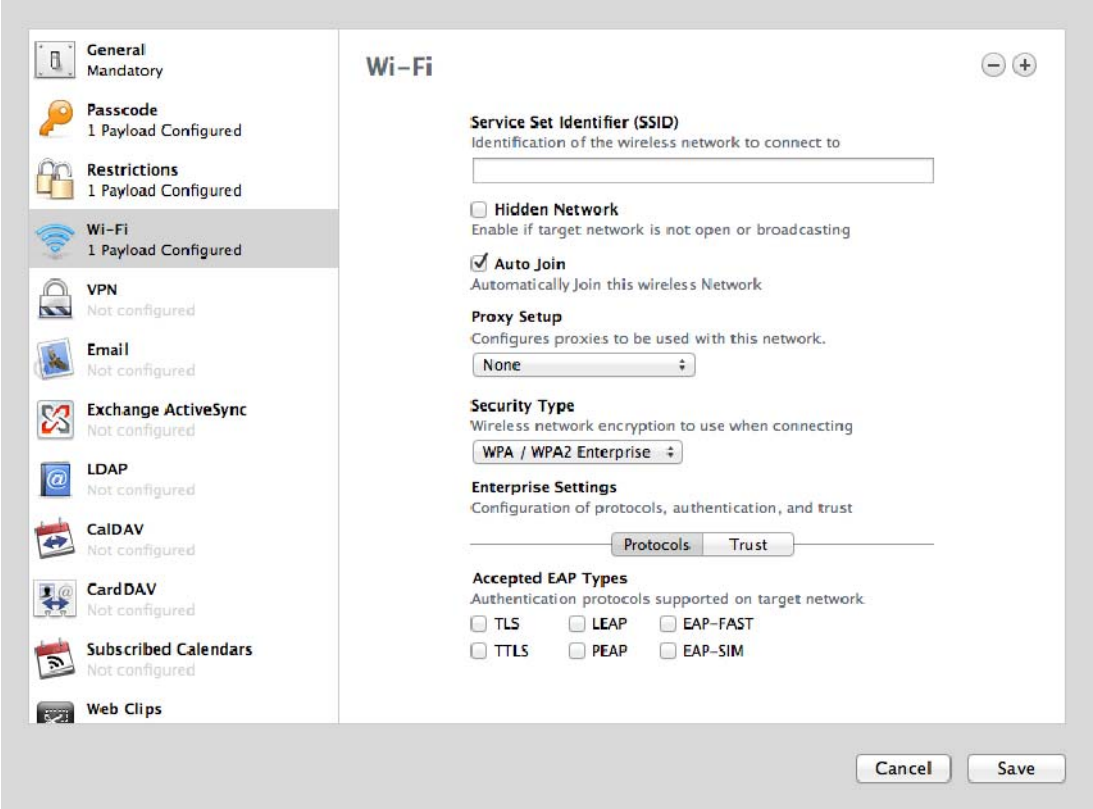


Figure 12: Wi-Fi Payload

- SSID of network as appropriate
- hidden SSID as per agency policy
- WPA2 Authentication with EAP-TLS and a pre-shared key as a minimum, but per user RADIUS or 802.1X is recommended
- protocols, authentication and trust to match network requirements. 802.1X with device identity certificate and username/password is the preferred authentication mechanism for Unclassified (DLM) and higher.

VPN:

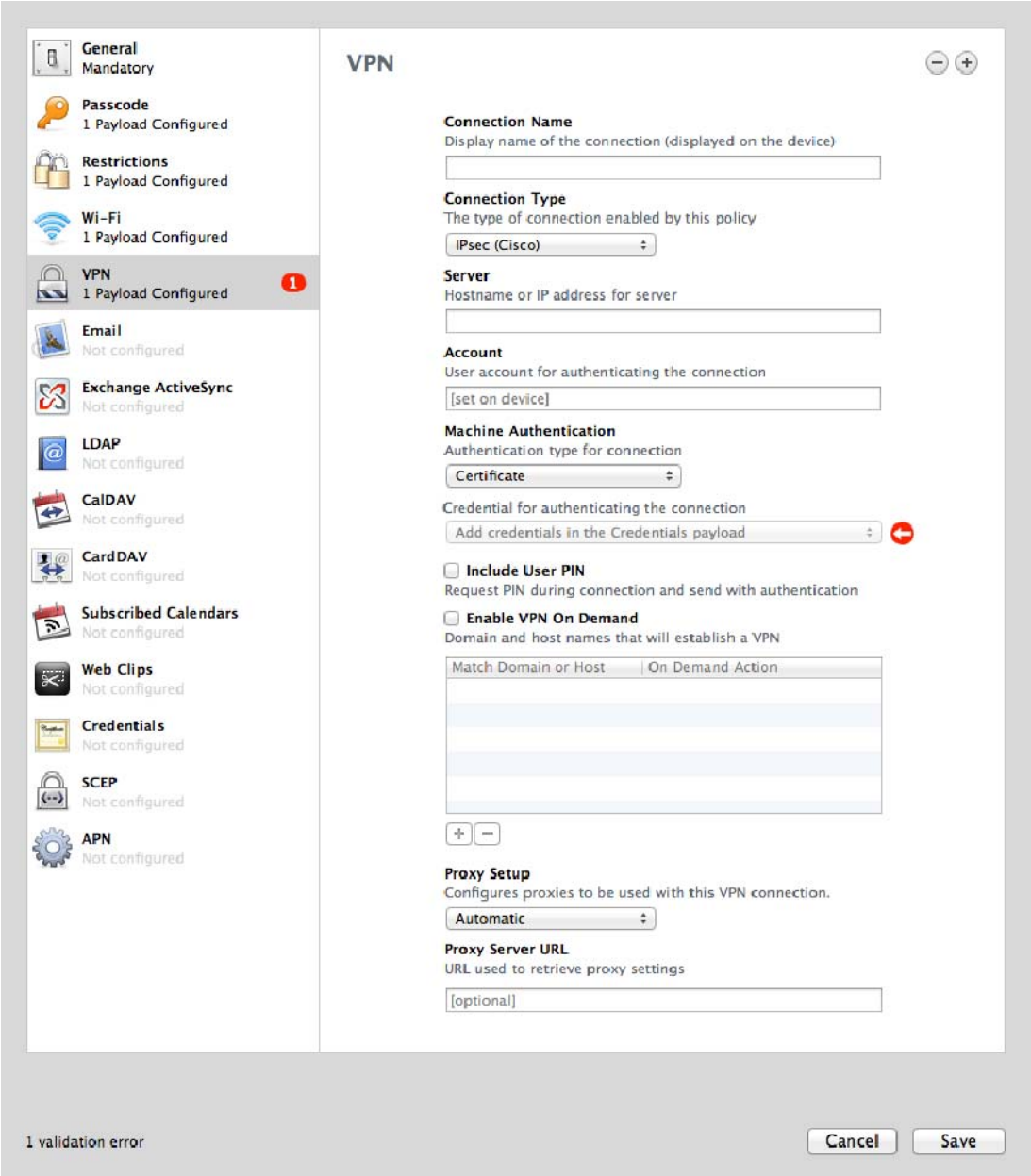


Figure 13: VPN Payload

- IPSec and SSL are DSD Approved Cryptographic Protocols, please refer to the Evaluated Products List (EPL) for more information – http://www.dsd.gov.au/infosec/epl/
- "VPN Server Configuration for iOS Devices" on http://help.apple.com/iosdeployment-vpn/ should be consulted for server side settings that iOS supports.
- certificate based Machine Authentication. Full trust chain needs to be included.
- split tunnel VPN should be off (set VPN concentrator side)
- VPN on Demand should be enabled with a whitelist of agency URLs or domains that device is allowed to access
- proxy should be configured, ideally a PAC file.

Email

- Not typically needed if EAS (e.g. Exchange ActiveSync Gateway, Lotus Notes Traveller) is in use. Otherwise appropriate to IMAP server, and can co-exist with Exchange.
- If set, SSL only, with authentication.

Exchange ActiveSync

- Settings as per EAS server details, SSL authentication credentials required to control both which device and which users have access to EAS.
- Note if a profile with an EAS payload is removed, all EAS synced email and attachments are deleted from the device.

LDAP

- As per agency requirements if desired. Not typically needed if Exchange GAL is used, but can co-exist.
- SSL recommended.

CalDAV

- As per agency requirements if required. May not be needed if Exchange used, but can co-exist.
- SSL recommended.

CardDAV

- As per agency requirements if required. May not be needed if Exchange is used, but can co-exist.
- SSL recommended.

Subscribed Calendars

- As per agency requirements.
- SSL should be used if there is any sensitivity to the calendar data.

Web Clips

- As per agency requirements. These are “aliases” or links to URLs with a custom icon on the home screen.
- Typical use would include links to pages for AUP, helpdesk contact details, telephone URLs, and SCEP re-enrolment pages. Note that these web pages could use preference manifest settings in their HTML to work when the site is offline or the device is off the network.
- Web clips can also be used to install Enterprise In-House Applications.

Credentials

- Include SSL chain of trust back to the root CA certificate, including intermediates.

SCEP

- Used when pre-configuring SCEP enrolment prior to device issue - rather than OTA opt-in. OTA opt-in is the normal method used.

MDM

- Used when pre-configuring MDM enrolment prior to device issue - rather than OTA opt-in. OTA opt-in is the normal method used.
- Usually, credentials should be added, all messages signed, and all access rights enabled for remote administrators.

- The Development APNS should generally not be used for production systems.

Advanced (Used when a custom APN for Mobile data is used)

- Authentication should be set.
- Proxy should be set appropriately.

All details here are worked out with the telecommunications carrier.

Other Settings not managed by Configuration Profile:

GSM Voice and SMS/MMS

- GSM Voice and SMS/MMS should only be used for Unclassified data at this time.
- Whilst a secure VOIP solution is technically possible, no Sectera compatible solutions are available on iOS at time of writing.

Mobile Data

- A SIM PIN should be set prior to issue.
- Data Roaming should generally be set to off.

Bluetooth

- Generally, Bluetooth should be set to off, unless there is a specific business reason for its use (e.g. Bluetooth headset with a phone, or Bluetooth Keyboard). See ISM section *Mobile Devices* for further information.

Picture Frame (iPad Only)

- This feature is a similar to a screen saver on the login screen.
- It should either be set to point to a specific Photo Album that contains data of no sensitivity (under Settings → Picture Frame), or Picture Frame can be turned off in Settings → General → Passcode.

Wi-Fi

- "Ask to join networks" should be set to off. This requires the user to explicitly choose to join a network. iOS auto-joins previously known networks only.

Dock Connector

Whilst unlocked, iOS could establish a trust relationship through the dock connector with devices or host computers. This behaviour can be managed in **Supervised** devices provisioned using Apple Configurator. It is recommended that users be instructed to only connect their iOS device to their agency-issued charger or computer.

Chapter Seven
Mobile Device Management

iOS 5 devices can use web and SCEP servers to establish trust relationships, and pull policy to devices. Devices establish initial trust via SCEP and then can be monitored and managed by servers, services or appliances using Apple's MDM Protocol, and APNS.

Management without MDM

Policy on iOS devices and information security can be managed by a combination of:

- Configuration Profiles loaded on a device
- Exchange ActiveSync policy
- Network security features (e.g. SCEP, 802.1X, firewalls, Proxies, custom APNs)
- Application specific behaviour (Configuration Profiles can be loaded via the iPhone Configuration Utility over USB, pulled OTA from a web site, or included in piggybacked on an SCEP enrolment transaction). In addition, they can be emailed to a device, but this can present a "chicken-and-egg" problem. Sending an SMS containing a website URL is possible, but as SMS can be easily spoofed, it is generally not recommended. For small scale or limited scope deployments, a full iOS 5 MDM solution may not be needed, but it usually has significant advantages with larger fleets, or more complex usage scenarios.

MDM Vendors

At the time of writing this guide there were a number of vendors shipping MDM solutions that have full support for Apple's MDM protocol and APNS integration. Some of these MDM solutions focus purely on device policy and monitoring. Others enhance this functionality, providing additional features via an App and event triggers for business rules that integrate with Exchange ActiveSync, Certificate Authorities and Directory Services. Many vendors can manage multi-platform clients. In this chapter the discussion will be restricted to iOS features.

MDM functions

Once an iOS 5 device is enrolled with an MDM Server, an Apple MDM agent is activated on the client device. It can then perform a number of tasks without user interaction, including querying status of the device, and installing or removing Managed Profiles. The interaction between an MDM server and a device can occur in two or three main ways:

- The MDM server can send an APNS notification to a device.
- A device, typically on receipt of a push notification, contacts the MDM server in an SSL encrypted session and exchanges information using XML. This

may be a simple query/response transaction or it may lead to the device pulling content down from a location the MDM server told it to, such as a configuration profile or provisioning profiles.

- The MDM vendor may also have a client App that can interact with the MDM server. Such Apps can interact in proprietary ways beyond the functionality that the MDM protocol allows for. Such Apps do not operate at any elevated level of privilege, and if available on the App store, are subject to normal App Store approval processes, but can enhance the functionality and the user experience.

Note that an MDM server cannot install native Apps remotely without some user intervention. Web apps can be deployed without user intervention by pushing a web clip to the device. Remote App installation occurs a number of ways:

- The MDM server can silently install or remove provisioning profiles to enable or disable an application from running on a device. The application binary still needs to be downloaded to the device by some means. Enterprise Apps can either have a provisioning profile external to the App so it can be installed/removed, typically via MDM, or have the provisioning profile embedded within the App itself, which means downloading the App bundle is sufficient for it to run (if present, the Provisioning Profile is copied from the App bundle, by the installer, and installed when the App is installed).
- The MDM server can silently install or remove a configuration profile that contains a web clip. If the web clip points to an appropriately constructed web site, touching on it will download an Enterprise iOS application to the device. The Web clip can also be the URL for a Web app, in which case it is usable immediately.
- The MDM solution may also, either via a native app or a web app, provide a list of approved or recommended App Store Apps, and Enterprise In-House Apps. When touched by a user, this will open the App Store or web server on the device for the user to download or purchase.
- The MDM can install “Managed Apps” remotely, the user is still required to accept the install but doesn’t need to authenticate. However, in Australia only free App Store Apps and enterprise in-house Apps can be installed as Managed Apps.

Appendix A
Security Checklist

The following checklist will assist an agency in ensuring that all key tasks in securely deploying iOS devices have been completed.

Task	Comments
Before Deploying iOS Devices	
Develop agency policy and procedures, including any restrictions, for the use of iOS devices that align with Australian government legislation, policies and standards, and that adhere to Australian government requirements.	Effective policies and procedures help to ensure that an agency considers relevant issues and operates in accordance with legislation and whole-of-government guidelines. Documenting and making these available to users will help ensure that users are aware of an agency's expectations of them when using mobile devices. On iOS devices, placing a policy Web Clip on the device makes it highly accessible to the user.
Implement processes to security classify, protectively mark, and control the flow of information that may be transmitted to/from the iOS device.	Filtering solutions at the EAS server can both filter and mark email based on header metadata and shorthand notation in the subject line. Agencies must security classify and protectively mark all email and controls must be implemented at email servers and gateways to restrict delivery of inappropriately classified information to and from an agency, including to mobile devices.

Task	Comments
Undertake an iOS device pre-implementation review.	Agencies deploying iOS devices may consider undertaking a pre-implementation review. This review would assess the planned deployment strategy, mitigation controls, policies and procedures against the requirements defined in the relevant policy and guidance documents. DSD can assist in ensuring the necessary steps have been followed.
Manage Use of iOS Devices	
Provide users with training on the use of iOS devices and security requirements.	In many areas of administration, failure to follow policies and procedures is not a result of deliberate actions, but a lack of awareness of requirements. Training in the appropriate use of devices can assist users to implement policies and procedures. The existence of training can also help distinguish deliberate misuse from incompetent usage. As part of this training agencies should also inform users that these devices are likely to be an attractive target for thieves, and that the implications of the information contained in them being accessed by others could be detrimental to the Australian government.
Ensure that users formally acknowledge their agreement to adhere to agency specific Acceptable Usage Policy and procedures.	Users using a mobile device are responsible for its use. Users must be aware of and agree in accordance with the agency's policy and procedures. The ramifications of failing to apply those policies and procedures must also be clear to users.

Task	Comments
Ensure that users classify and protectively mark all email with the highest classification of the content or attachment, in accordance with Australian government standards.	Users must be conscious of the security classification of information that they are sending to or from mobile devices. Agencies must ensure that users classify and protectively mark all agency-originated email or attachments in accordance with the highest classification of the content.
Infrastructure Issues	
Server infrastructure for EAS, MDM, CA and Web that supports an iOS deployment must be controlled, either directly or under contract, by the Australian government.	Use of EAS, MDM and CA infrastructure allows many risks to be mitigated. These servers should be situated in a controlled environment, and will permit the implementation of consistent policy and device settings. Software as a Service (SaaS) solutions may not be acceptable for production deployments.
Agencies must ensure that content is transferred between an iOS Device and an agency's ICT systems in accordance with Australian government policy.	Email protective marking filtering mechanisms must be implemented to provide a higher level of security by automatically preventing information of an inappropriate classification being sent to a mobile device. These mechanisms are described in the Implementation Guide for Email Protective Markings for Australian government Agencies.

Task	Comments
Ensure that email originating outside the agency is not sent to the iOS device unless it is classified and labelled appropriately.	Communications originating outside the agency may also include classified information. The policies and standards applied to external communications must also be applied to internally generated information. Emails that do not have protective markings should not be transmitted to mobile devices. Agency policy may define a subset, e.g. an agency may only permit Unclassified information to be forwarded to a mobile device. These mechanisms are described in the Implementation Guide for Email Protective Markings for Australian government Agencies.
Review and Audit	
Undertake an iOS post implementation review.	Agencies that deploy iOS devices must undertake a post implementation review. This may assist in identifying policy and implementation inconsistencies and assess the mitigation controls for completeness against the Risk Management Plan (RMP), The System Security Plan (SSP), Standard Operating Procedures (SOP) and the implementation of email protective marking controls. This review must be completed within twelve months of the live production deployment.
Audit compliance with policies and standards for the use of iOS devices.	Setting out policy without monitoring compliance is unsound practice. There should be appropriate internal and – from time to time – external checks of compliance with policies regarding the use of mobile devices. There should also be regular reviews of internal policies, to test their currency and adequacy.

Appendix B
Configuration Profiles Format

This Appendix provides the references for the format of mobileconfig files for agencies wishing to create their own tools or custom configurations without deploying a commercial MDM solution.

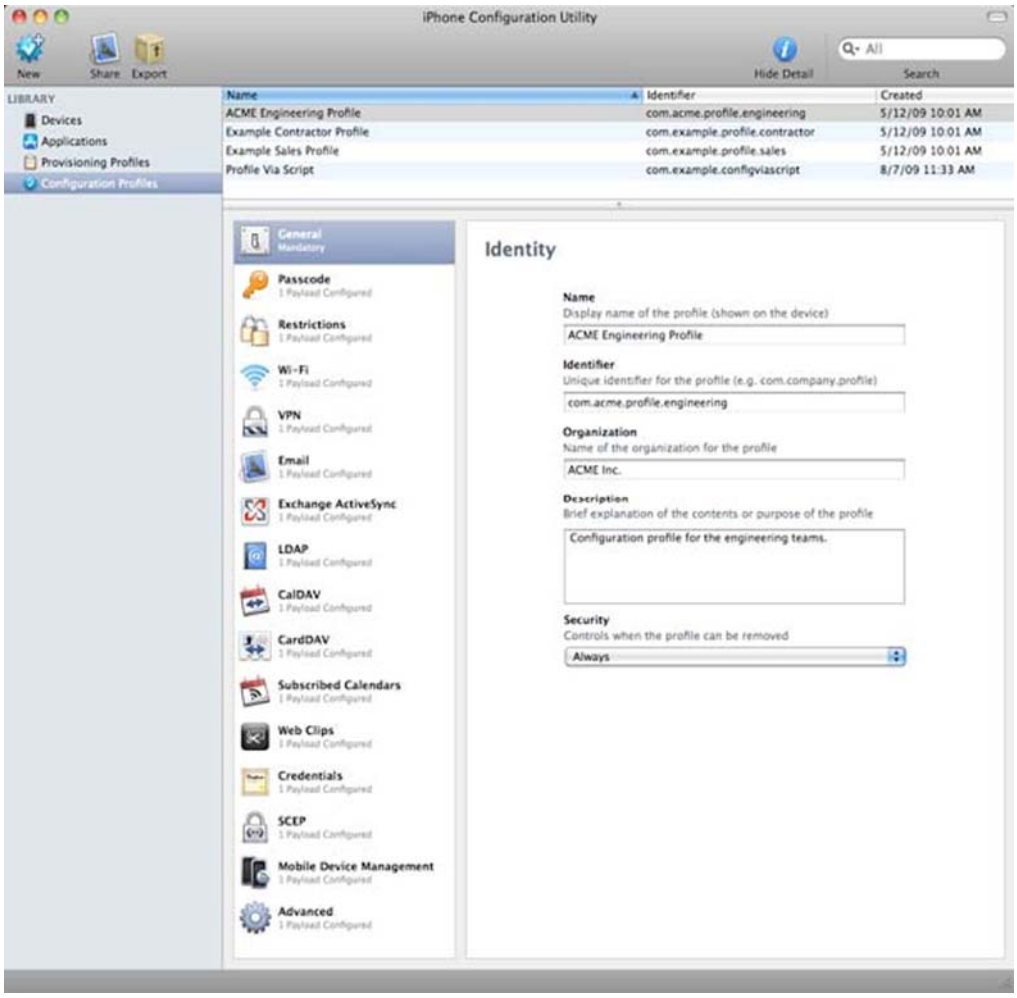
Configuration Profiles use the Apple XML DTD and the general property list (plist) format. A general description of the Apple plist format is available at www.apple.com/DTDs/PropertyList-1.0.dtd.

To get started with Configuration Profiles, iPhone Configuration Utility (iPCU) can be used to create a skeleton file that can be modified using the information in this appendix, or use the examples at http://developer.apple.com.

iPhone Configuration Utility is documented in detail here:

http://developer.apple.com/library/ios/#featuredarticles/FA_iPhone_Configuration_Utility/ Introduction/Introduction.html
A screen shot of the iPhone Configuration Utility is on the following page, showing the range of different profile payloads. This document uses the terms payload and profile. A profile is the whole file that configures certain (single or multiple) settings on iPhone, iPod touch, or iPad. A payload is an individual component of the profile file.

iPhone Configuration Utility



- For further information on configuration profile format, full documentation is available from:
- http://developer.apple.com/library/ios/ -featuredarticles/iPhoneConfigurationProfileRef/Introduction/Introduction.html
- For further information on configuration profiles, including scripting of iPCU and sample Ruby code for building a SCEP server that generates profiles on demand, see:

http://developer.apple.com/library/ios/ -documentation/NetworkingInternet/Conceptual/iPhoneOTAConfiguration/Introduction/Introduction.html

Appendix C
Sample Scripts

This Appendix provides sample scripts for iOS deployment tasks. The scripts in this section should be modified to fit agency needs and configurations.

http://developer.apple.com/library/ios/ -
featuredarticles/FA_iPhone_Configuration_Utility/Introduction/Introduction.html

Sample C# Script for iPhone Configuration Utility

This sample script demonstrates creating configuration files using iPhone. Configuration Utility for Windows.

```
using System;
using Com.Apple.iPCUScripting;
public class TestScript : IScript
  {
 private IApplication _host;
 public TestScript()
          {
          }
public void main (IApplication inHost)
{ _host = inHost;

string msg = string.Format("# of config profiles : {0}",
_host.ConfigurationProfiles.Count);
Console.WriteLine(msg);

IConfigurationProfile profile = _host.AddConfigurationProfile();
 profile.Name = "Profile Via Script";
profile.Identifier = "com.example.configviascript";
profile.Organization = "Example Org";
profile.Description = "This is a configuration profile created via the new scripting
feature in iPCU";

// passcode
IPasscodePayload passcodePayload = profile.AddPasscodePayload();
passcodePayload.PasscodeRequired = true;
passcodePayload.AllowSimple = true;

// restrictions
IRestrictionsPayload restrictionsPayload = profile.AddRestrictionsPayload();
restrictionsPayload.AllowYouTube = false;

// wi-fi IWiFiPayload
wifiPayload = profile.AddWiFiPayload();
wifiPayload.ServiceSetIdentifier = "Example Wi-Fi";
wifiPayload.EncryptionType = WirelessEncryptionType.WPA;
wifiPayload.Password = "password";
```

```
wifiPayload = profile.AddWiFiPayload();
profile.RemoveWiFiPayload(wifiPayload);
// vpn
IVPNPayload vpnPayload = profile.AddVPNPayload();
vpnPayload.ConnectionName = "Example VPN Connection";
vpnPayload = profile.AddVPNPayload();
profile.RemoveVPNPayload(vpnPayload);

// email
IEmailPayload emailPayload = profile.AddEmailPayload();
emailPayload.AccountDescription = "Email Account 1 Via Scripting";
emailPayload = profile.AddEmailPayload();
emailPayload.AccountDescription = "Email Account 2 Via Scripting";

// exchange
IExchangePayload exchangePayload = profile.AddExchangePayload();
exchangePayload.AccountName = "ExchangePayloadAccount";

// ldap
ILDAPPayload ldapPayload = profile.AddLDAPPayload();
ldapPayload.Description = "LDAP Account 1 Via Scripting";
ldapPayload = profile.AddLDAPPayload();
ldapPayload.Description = "LDAP Account 2 Via Scripting";

// webclip
IWebClipPayload wcPayload = profile.AddWebClipPayload();
wcPayload.Label = "Web Clip 1 Via Scripting";
wcPayload = profile.AddWebClipPayload();
wcPayload.Label = "Web Clip 2 Via Scripting";

        }
 }
```

Sample AppleScript for iPhone Configuration Utility

This sample script demonstrates creating configuration files using iPhone Configuration Utility for Mac OS X.

```
tell application "iPhone Configuration Utility"
        log (count of every configuration profile)

        set the Profile to make new configuration profile with properties{displayed
name: “Profile Via Script", profile identifier:"com.example.configviascript",
organization:"Example Org.", account description:"This is a configuration profile
created via AppleScript"} with properties {label:"Web Clip Account 1 with properties
{label:"Web Clip Account 2”}

        tell theProfile
                make new passcode payload with properties {passcode required:true,
        simple value allowed:true}
                make new restrictions payload with properties {YouTube
        allowed:false}
                make new WiFi payload with properties {service set
        identifier:"Example Wi-Fi", security type:WPA, password:"password"}
```

```
        set theWiFiPayload to make new WiFi payload
        delete theWiFiPayload
        make new VPN payload with properties {connection name:"Example
VPN Connection"}
        set theVPNPayload to make new VPN payload
        delete theVPNPayload
        make new email payload with properties {account description:"Email
Account 1 Via Scripting"}
        make new email payload with properties {account description:"Email
Account 2 Via Scripting"}
        make new Exchange ActiveSync payload with properties {account
name:"ExchangePayloadAccount"}
        make new LDAP payload with properties {account description:"LDAP
Account 1 Via Scripting"}
        make new LDAP payload with properties {account description:"LDAP
Account 2 Via Scripting"}
        make new web clip payload Via Scripting"}
        make new web clip payload Via Scripting"}
                end tell
        end tell
```

Appendix D
Example Scenarios

This Appendix describes hypothetical scenarios showing how the various techniques can be combined.

Unclassified Example

An art gallery wishes to use iPod touches as an interactive tour guide for Unclassified information at a specific site. The tour guide information is largely contained within a single App.

The Gallery purchased an Enterprise Developer Agreement, and uses this to code-sign the App they have had developed by a contractor.

They set up a Wi-Fi network for the site, and use a Kiosk with a locked down instance of iTunes, and OTA App and profile provisioning from a secured web server to deploy, manage and reset devices during use with minimal effort.

Unclassified (DLM) Example

An agency wants to use iPads as a field-based information gathering tool for its users. Information will come from a mix of existing web sites and with some data entry fed into an existing system with an XML interface using an Enterprise In-House App the agency has developed. The devices will also allow users to send and receive email in the field. The agency's primary WAN is classified "PROTECTED".

In this case the agency uses a combination of an MDM server, Exchange ActiveSync, and a 3rd party gateway filter. These enable the agency to control policy on the devices, control email that is sent to the devices and implement protective markings on email sent from the devices. Access to the limited intranet sites with Unclassified (DLM) data is controlled by a reverse proxy. The custom App and its supporting server infrastructure undergo a separate TRA (Threat Risk Assessment). A Wi-Fi network configured according to relevant ISM government system (G) controls is provided at selected locations to support OTA provisioning and updating of devices.

PROTECTED Example

An agency has decided to issue iPhones for their mobile fleet. Their users require access to their PROTECTED email and attachments, as well as access to a PROTECTED intranet.

The IT team will use Apple Configurator to configure the iPhones before they are issued to users. The iPhones will be configured as **Supervised** devices and will be pre-enrolled with the agency's MDM server.

The iPhones will connect to the agency's exchange server using a client certificate for authentication. The VPN will be configured as "On-Demand" with certificate authentication and the whitelist will be a regular expression that matches all top level domains – this forces all traffic over the VPN. The agency has a custom APN for their mobile data traffic and the VPN proxy is set to the agency's proxy so that all internet traffic flows through the agencies certified gateway.

The agency will require all users to sign an acceptable use policy that requires them to install OTA updates when they are available, compliance will be monitored by the IT team using the MDM.

Appendix E
Risk Management Guide

This Appendix provides a guide to typical risks associated with mobile devices and recommended mitigation measures.

Australian Government Information Security Manual (ISM)

This appendix should be read in conjunction with the ISM, available from the DSD website: http://www.dsd.gov.au/infosec/ism.htm
iOS devices do not completely comply with all requirements described within the ISM.

Mobile Device Risks

Typical risks, the recommended mitigation measures and the pre-conditions for those mitigation measures are covered in the table below. There are several residual risks in ISM policy that cannot be completely mitigated by technical controls. Agencies will need to assess, accept and manage any residual risks and develop appropriate policy guidance.

- iOS 5 does not have a local firewall. This is partially mitigated by firewalling at the network layer, and significantly mitigated by the sandboxed runtime environment in iOS.
- iOS 5 allows the user to deliberately connect to an untrusted Wi-Fi network. Note that iOS devices will not autoconnect to any unknown Wi-Fi network. The only mitigations available at this time are pre-configured settings, user education and AUP.
- iOS 5 allows the user to deliberately enable or disable the radios in the device - there is no method for a configuration profile to force a radio off. The only mitigations available at this time are user education, AUP or hardware modification (the latter being permanent and will void the warranty).
- iOS 5 has no “always-on” setting for VPN. It is either manually initiated, or on-demand based on a whitelist. Options to mitigate this for PIM data (if EAS and/or VPN on demand are assessed as insufficient mitigations) include using a 3rd party PIM solution, filtering at the EAS, or using approved VDI solution to access sensitive data. For web site access, an SSL reverse proxy may be more suitable than VPN in some scenarios.

Risk	Mitigations	Implied Preconditions
Device lost, still on network	Strong passcode, data protection enabled, remote wipe, Find My iPhone/iPad.	Configuration Profiles, EAS or MDM Server in a network reachable location or iCloud account.
Device lost, off network	Strong passcode, local wipe, data protection enabled.	Configuration Profiles, Device restored to iOS 5 prior to use in field.
Device lost, casual access attempt	Strong passcode, local Wipe, data protection enabled.	Configuration Profiles, Device restored to iOS 5 prior to use in field.
Device lost, forensic access attempt without passcode knowledge	Strong passcode, local wipe, use of Supervised mode, data protection enabled, App usage of appropriate data protection class[9].	Configuration Profiles, Device restored to iOS 5 prior to use in field. Device in Supervised Mode.
Jailbreaking	Strong passcode, data protection enabled, use of devices with hardware cryptographic module, use of MDM console, use of VDI infrastructure. Use of Supervised mode, MDM App or enterprise Apps with "canary" code to detect and report jailbreaking, AUP should prohibit jailbreaking.	Jailbreaking from host computer when device passcode is known is still likely to be feasible, unless supervised mode is used.

[9] Information for developers implementing data protection classes is available from: http://developer.apple.com/videos/wwdc/2010/?id=209

Risk	Mitigations	Implied Preconditions
Malicious runtime code	Code signing, memory and filesystem sandboxing, use of VDI infrastructure, no-execute heap, disable user-added applications, do not jailbreak operational devices.	In-house application development capability, CA infrastructure. May mitigate on lower security levels by “approved” lists and MDM monitoring as mitigation.
Users cut and paste agency data into a public email account (e.g. Yahoo or Gmail) and sent it from the device.	On iOS 5 disable the creation of separate email accounts, and restrict access to webmail via custom APN and agency proxy, disable screen shots on device via Configuration Profile, filter sensitive mail or attachments at the EAS gateway, use of VDI for sensitive email, containing agency email to a third party email App container.	Configuration Profiles, use of agency proxy. Note that any data that is displayed on the screen of any device can be photographed or video recorded by a camera, and sent via other means. This kind of leakage by deliberate action generally cannot be mitigated against for a mobile device.
Network trust	Use of 802.1X NAC, IPSEC or SSL VPN, encrypted VDI.	Use of 802.1X with CA & NAC on Wireless, VPN on Demand with client certificates for agency network access, use of SSL reverse proxy for low security data.

Risk	Mitigations	Implied Preconditions
Firewall	Use of Custom APN on 3G, 802.1X, SSL VPN.	A custom APN is an arrangement with the agency telephone carrier. This allows devices on 3G data to have a deterministic IP range that can be more easily firewalled or proxied.
Data compromise via host computer backup	Force encrypted profile onto device, user education, physical security of backup host, iTunes in host SOE.	SSL CA infrastructure to sign and encrypt profiles into agency chain of trust. Potentially allow use of locked down iTunes configuration on agency computers so backup resides on agency assets.
Data compromise via Bluetooth	OS 5 only includes four or six of the 26 Bluetooth profiles, depending on device, and specifically does not include file transfer related Bluetooth profiles. Included profiles are for microphone, speakers, and human input devices, as well as Apps that use a Bluetooth PAN. See http://support.apple.com/kb/HT3647	Apps that share information via Bluetooth PAN not approved for use on devices where this vector is a concern.

Appendix F
Firewall Rules

Several firewall rules may need to be implemented to allow correct functionality. Depending on what functionality is required from iOS devices, MDM servers and iTunes several firewall rules may need to be implemented.

Firewall ports

iTunes and iOS devices may need firewall rules adjusted, depending on the functionality required, or allowed, on an intranet. The main knowledge base articles describing ports required by Apple devices are given below, with a summary around iOS and iTunes in the following table below:

- http://support.apple.com/kb/TS1379
- http://support.apple.com/kb/TS1629

DNS name	Port(s)	Reason
ocsp.apple.com	443	Online Certificate Status for code signing certificates, checked periodically while online and after device reboot.
crl.apple.com	443	Certificate Revocation List for codesigning certificates, checked periodically while online and after device reboot.
gateway.push.apple.com	2195 (outbound push e.g. MDM) 2196 (for devices to receive)	Apple Push Notification Service (for a development environment only, gateway.sandbox.push.apple.com is used instead).

feedback.push.apple.com	2195 (outbound push - e.g. MDM) 2196 (for devices to receive)	Apple Push Notification Service (for a development environment only, feedback.sandbox.push.apple.com is used instead).
phobos.apple.com	80, 443	iTunes Store, Device Activation.
itunes.apple.com	80, 443	iTunes Store, Device Activation.
deimos.apple.com	80, 443	iTunes U.
deimos3.apple.com	80, 443	iTunes Music Store and album cover media servers.
ax.itunes.apple.com	80, 443	iTunes Store, Device Activation.
gs.apple.com	80, 443	iTunes Store, Device Activation.
albert.apple.com	80, 443	iTunes Store, Device Activation.
ax.init.itunes.apple.com	80, 443	Device Activation.
evintl-ocsp.verisign.com	80, 443	Verification of digital signatures of iTunes purchased content.
evsecure-ocsp.verisign.com	80, 443	Verification of digital signatures of iTunes purchased content.

a1535.phobos.apple.com	80, 443	iTunes Music Store and album cover media servers.