Learning iOS Security

Enhance the security of your iOS platform and applications using iOS-centric security techniques

Allister Banks  Charles S. Edge
In this package, you will find:

- The authors biography
- A preview chapter from the book, Chapter 4 ‘Organizational Controls’
- A synopsis of the book’s content
- More information on Learning iOS Security

About the Authors

**Allister Banks** is an enthusiast. He's very excited to be in the exceedingly limited, exclusive club of coauthors of Charles S. Edge. After working for a decade with IT consulting companies on both the coasts of the U.S., he now works for a medical-focused institution with education and data center aspects. He has given speeches at LOPSA-East, MacTech Conference, and MacAdmins Conference at Penn State. He lives in New York. He contributes to various open source projects and speaks enough Japanese to order food.

**Charles S. Edge** has been working with Apple products since he was a child. Professionally, Charles started with the Mac OS and Apple server offerings in 1999 after working of years with various flavors of Unix. Charles began his consulting career with Support Technologies and Andersen Consulting. As the chief technology officer of 318, Inc., a consulting firm in Santa Monica, California, Charles built and nurtured a team of over 50 engineers, which was the largest Mac team in the world at that time. Charles is now a product manager at JAMF Software, with a focus on Bushel (http://www.bushel.com).

Charles has spoken at a variety of conferences including DefCon, BlackHat, LinuxWorld, MacWorld, MacSysAdmin, and Apple Worldwide Developers Conference. Charles has also written 12 books, over 3,000 blog posts, and a number of printed articles on Apple products.
Learning iOS Security

Nowadays, iOS is becoming more and more prevalent in companies and larger organizations. Whether this is a trend that is driven by Bring Your Own Device (BYOD) or something that is coming from within the IT department, our knowledge of platforms is being stretched more and more all the time. It's getting harder and harder to be an expert on every platform that is in use in our organizations!

You need to secure your iOS devices. Learning iOS security gives you the knowledge to build security into large-scale iOS deployments. This book takes you through good security practices; these include configuring privacy options to keep personal data away from prying eyes, learning about encryption options to keep data safe at rest, securing apps to reduce the risks introduced by third-party apps, and then laying down practical steps and procedures for carrying out these steps, both on-screen on devices and at scale using Apple Configurator, profiles, and Mobile Device Management (MDM) solutions.

This book also includes a section on debugging and viewing data so that you can check out how to further secure items not covered in detail in the book. We teach you how to provide enterprise-class security to your iPhone, iPad, and iPod Touch deployments. This includes a quick run-down of basic security steps and mass deployment of these steps to aid in your large-scale deployment of iOS devices.

This book is meant to be an easy-to-digest guide that follows real-world examples to implement best security practices. Each topic is covered in a theoretical context and further resources are provided where they are needed/applicable.

What This Book Covers

Chapter 1, iOS Security Overview, is a quick-and-dirty overview of the many steps to take to initially secure an iPad, iPhone, and iPod Touch. The purpose of this chapter isn't to go into too much depth with any given technology, but to provide a cheat sheet of sorts to get you started with iOS security.

Chapter 2, Introducing App Security, is a more thorough review of how to choose apps and secure them during an iOS deployment. Here, we look at an overview of sandboxing techniques and how to use Single App Mode and keybags. We also look at in-house Apps.

Chapter 3, Encrypting Devices, explains the encryption types and techniques that are used in iOS. Here, we look at Touch ID, Apple Pay, network encryption, and privacy concerns.
Chapter 4, *Organizational Controls*, introduces Apple Configurator and profile management. Here, we also look at the Find My iPhone app as it pertains to Activation Lock, ActiveSync policies (EAS Policies), and device supervision.

Chapter 5, *Mobile Device Management*, looks at Apple's Profile Manager and a simple third-party MDM called Bushel. Here, we look at Over the Air (OTA) profile management.

Chapter 6, *Debugging and Conclusion*, covers ways to troubleshoot and debug devices in larger deployments. In this chapter, we'll look at how to find logs and interpret them, how to get more data than you can use from devices, and then we will wrap up the book.
Now, we'll move on to explore the concepts involved in managing iOS devices from a central location on-premises. This includes device supervision, Activation Lock, Single App Mode, and more basic options presented by the old stalwart, ActiveSync. For most of the time, we will be looking at a tool called Apple Configurator that is developed by Apple. We consider it to be one of the easiest tools to recommend for environments that need more hands-on control when officially supporting iOS, either when migrating to a BYOD (short form for bring your own device) environment or in conjunction with an MDM. It fits a couple of specific workflows very well and has some features that are vital for hardening devices.

Besides Apple Configurator, which at the very least can provide a good reference for showing Apple's acknowledged use cases for starting with device management, we will also introduce Apple's Device Enrollment Program or DEP. Activation Lock is a thornier topic now, so we'll touch on this as well. Just to transition from Guided Access, which was covered in Chapter 2, Introducing App Security, we'll also discuss App Lock when we explain the difference between it interacting with Guided Access and Single App Mode. And, before we get into full-blown MDM in the following chapter, we will discuss ActiveSync as one of the original over-the-air management frameworks.

In brief, this chapter's topics are as follows:

- Apple Configurator
- Preparation, supervision, and assignment of iOS devices
- The distribution of apps with Apple Configurator and the Volume Purchase Program
- Activation Lock and Find My iPhone
- The Device Enrollment Program versus Apple Configurator
- App Lock and Single App Mode in contrast to Guided Access
- Refresher on what ActiveSync provides on iOS
Apple Configurator

Before the release of Apple Configurator on the Mac App Store, there were three other sanctioned applications for interaction with iOS devices: iTunes, Xcode, and iPhone Configuration Utility (iPCU). Xcode had the capability to connect multiple devices simultaneously, but even that functionality was limited for running tests on devices or for restoring a version of iOS. Still, we were without any concept of efficient, directly connected management tools, nor even the hint of integration with a directory service.

When the iPad was released, it did not come with a manual like a lawn mower, which shows you what its intended usage is and how to sharpen the blades. Apple just about said the same thing to its customers that it says to its developers, something to the effect of "we can't wait to see what YOU do with it", as if it was still an open question as to what its most popular use would be. Apple products have, however, historically been used extensively in education and the price was commonly a half to a third of the least expensive laptop Mac. This led to an influx of iPads in environments that might not have been particularly prepared to have so many computing devices on Wi-Fi. This leads us back to the lack of applications that allow tethered preparation and maintenance of many devices at once.

Perhaps, if customers that used Apple products for educational purposes in particular were asked what they wanted, as the paraphrased saying attributed to Henry Ford goes, they would have said a faster horse; instead they got Apple Configurator. We do not want to be repetitive, but we must recall that Apple's priorities are its customers first and foremost, and they sell an astounding amount of products to regular consumers. One may be inclined to cut them, and companies like Amazon who are selling to the general public with success, some slack, which is hard. Amazon's not trying to be CDW and Apple can't be everything to everyone; (although it has never stopped the sprawl of iTunes, which the Apple TV Assistant built into Apple Configurator which has a faint whiff of.)

Back in Chapter 2, Introducing App Security, we mentioned about the Volume Purchase Program (VPP) that Apple offers. This was an integral part of what was considered going into designing Apple Configurator, along with the Supervision concept that we've been hinting at throughout the book so far. However, before we get into that, let's discuss workflows.
Intended workflows

Of all the iOS form factors, at 9.6", the original and canonical iPad screen is comparably sized to 8.5" x 11" or an A4 sheet of paper, if you lose the margins and enjoyed staring at a light bulb all the time. (What? you don't prefer emissive screens?) If a telecom field worker has visited your home or business recently, you might have noticed that they now almost exclusively use tablets. Similarly, airlines have been giving their staff handheld devices for some time. When taking this rapid adoption of mobile devices into account, and recalling who Apple usually cares about when designing solutions, it may make more sense as to how Apple Configurator came into being.

An iPad can conceivably replace a utility worker's clipboard or a student's three-ring binders and streamline processes along the way. Airline pilots began demanding iPads to replace their ungainly and heavy binders of airport and route maps, which actually saved fuel due to the drop in weight. We can start to see that devices will be used in a multitude of ways, but a particularly apt case is high-service and quick-turnaround environments, loaded with the apps and data people need to get their work done.

Apple Configurator's release was groundbreaking in that it was a series of firsts:

- Applications could be handed out in bulk without MDM, and these apps could then be reclaimed
- Backups could be created and restored without iTunes and restored or refreshed en masse
- New, more locked-down restrictions could be enabled

Educational institutions segment time into classes and they often gather devices in labs or carts. Hospitals and utility workers have shifts and can make a station around a time clock or a gathering place for devices, from where they can be checked in and out from. It is widely reported that Apple does not have a colossal R&D footprint, so when they make a tool they have to please as many end users as possible. They don't have the resources to quality assure and develop features that can serve every market. Please keep all of this in mind as we discuss what Apple Configurator can do, with at least an understanding of why it doesn't make French fries four different ways.
The following screenshot shows the splash screen on starting Apple Configurator for the first time, which graphically introduces its three modes:

![Splash Screen](image)

The splash screen on starting Apple Configurator for the first time graphically introduces its three modes:

**The interaction modes – Prepare, Supervise, and Assign**

After acquiring Apple Configurator from the Mac App Store (it is free, but requires a Mac at this time), you’re greeted with an image that breaks down its three cumulative modes of operation. First, there are the capabilities of the Prepare mode, which are as follows:

- Naming the device (this includes the option of sequential, numeric naming if you are preparing multiple devices at once, as it can handle up to 30 devices concurrently)
- Creating a (unsupervised) backup
• Applying a software update (which caches that version) and optionally, wiping the device in the process
• Importing, creating, exporting and/or applying configuration profiles

Finally, flipping a switch to move the device to the next mode, Supervision

Flipping this switch to make the device become supervised changes the behavior of Apple Configurator’s options. Therefore, you must then wipe the device and apply the most recent iOS update.

One might say that these distinctions help to prove that the device is indeed owned and under the control of the institution managing these devices, as it is assumed that regular people wouldn't let IT seize their property and remove all personalization or customization. (If they are like our customers at least.) However, Apple Configurator can easily be used in Prepare mode to lightly run an OS update, install a configuration profile, or even perform a backup and restoration.

Our technical editor points out that the device must trust the computer running Apple Configurator first to even do these light tasks, as we'll exploit in Chapter 6, Debugging and Conclusion.

This helps us to clearly define the distinction between preparation and supervision, as the second layer's powerful functionality rests on top of the first. The last mode, Assign, has just two additions:

• First, you can leverage a local or network-based directory service
• Second, the data created by a user from the directory can be stored on the computer running Apple Configurator

This allows the user to check in or check out of data as well as sets of apps, and it can also aid in the distribution of documents to devices that have compatible apps installed on them. It may seem like we're jumping ahead to discuss the Assign mode, but that's really the only additional feature.
Organizational Controls

Other than that, as whiz-bang features go, if users from the directory service have images associated with their LDAP records, there is a preference to show these images on the lock screen when assigning devices. You will access it from the Apple Configurator menu in the top left-hand corner of the screen, under Preferences. However, the stars have never aligned to the point that we’ve seen that in use in the real world. The following screenshot shows, in Preferences, where an assigned device can be configured to use an image from LDAP:

![Preferences screenshot](image)

In Preferences, where an assigned device can be configured to use an image from LDAP

The importance of supervision

Once the device has been wiped and updated by being tethered to a computer running Apple Configurator, you can take advantage of several options. These include:

- Customizing the lock screen image, as shown in the preceding image, optionally with the device's name or some other static text
- Enabling various network-related features including Always-On VPN, Content filters, Global HTTP proxy (as discussed in the previous chapter), and cellular data modifications
• Restricting various features such as the manual installation of configuration profiles, AirDrop, account modifications including Find My Friends, enabling other on-device restrictions, education-specific concerns like Siri's profanity filter, and whitelisting destinations or presetting passcodes for AirPlay

• Hide (by which we mean disable, to bring about the effect that the app is not shown) built-in applications like Game Center, iTunes Store, iMessage, Podcasts, or store components like In-App Purchase or the iBooks Store

• Stop the removal of any other apps, including the ones that Apple Configurator may have installed, or prevent the addition of any so-called Internet accounts (such as Facebook, Twitter, and so on) or e-mail accounts

Restricting Safari does not require supervision, but it is a common error to believe that you'll allow all the web functionality you want by using a Web Clip payload in a configuration profile. For example, for accessing your intranet only. If you restrict Safari, the app will be removed and Web Clips will not even launch if present.

A bigger point than even these settings, which were advocated by so many of Apple's customers in large institutions, is the ability to install profiles with zero taps. If the device is still in Prepare mode, you'll need to respond to the prompts on the screen to accept certificate notifications, learn about what the profile will do to the device, and eventually, install, and then tap on done, per profile. Loading a profile onto a supervised device is silent. In fact, when restoring the backup to supervised devices, you don't even need to go through any setup or activation steps. (More recent versions of Apple Configurator can allow similar behavior without restoring a backup, by selecting which prompts to skip.)

If this wasn't a security book, we could probably stop here. However, by far the biggest point from a security perspective is the fact that, by default, a supervised device can be disabled from connecting to any other computer running Apple Configurator. An attacker cannot piggyback on iTunes to target another device too. This mitigates many of the pairing-based complications that we'll be discussing in Chapter 6, Debugging and Conclusion. In fact, if it was desirable to allow moving any content to the device from another computer, the device must be designated at time of supervision to **Allow devices to connect to other Macs** (by which they imply PCs as well).
Further, if a specific configuration profile with a restriction payload is applied, **Allow pairing with non-Configurator hosts** must also be selected. If you want to, this can allow you to optionally disable pairing later via MDM, in case it is not clear whether your end users will need it at the time of supervision, but if you are using Apple Configurator to supervise the device, then it must be connected to the computer again. You can see each of these settings in the following screenshot:

When discussing workflows, we said Apple Configurator is a good fit for high-service, fast-turnaround use cases, which leads to another big feature of supervision: the ability to refresh the device to a stored state upon reconnection. If this includes the restoration of a larger backup with many apps, this can be a more lengthy process, but in any case, all of the ingredients are cached locally in Apple Configurator’s support directories. (Apps such as iMovie and Keynote run in to hundreds of MBs and flash storage in general is optimized for reading and not writing, so it’s good to measure if the cycle time meets your expectations.) This can essentially reimage the iOS device if Apple Configurator is open on the computer to which the device is attached.

Optionally, in the event you are not restoring a backup, you can also have apps and profiles that may have been added to the deleted device, so user training regarding supervised devices is very important. If this behavior is not desired for any reason, you must at least temporarily turn off these settings in Apple Configurator’s Preferences, as shown in the following screenshot:
In Preferences where supervised devices are configured to automatically refresh when they are connected

**Apps, VPP, and Apple Configurator**

When the usage model is one customer for one device, an MDM can prompt an end user for their Apple ID. Apple Configurator doesn't require a user that receives a device prepared by it to plug anything in, allowing shared usage models that just weren't possible before.

If an Apple ID is authorized for use on the computer running Apple Configurator, even if it is not associated with VPP, you can go ahead and import and distribute free applications. The recommended way to go about obtaining the .ipa files (the archived bundles that are iOS applications, as discussed in *Chapter 2, Introducing App Security*) is to download them from the App Store section in iTunes. However, no matter what ID the app was downloaded with (for example, if an iOS device already synched with the computer and backed up its purchases with iTunes), the DRM can be removed from the app bundle and imported with whatever Apple ID Apple Configurator wants to use. However, if you forget to authorize the computer in iTunes, you'd see the following error:

When an app to be installed on a device is imported without the associated Apple ID authorized in iTunes
Keep in mind that the updates for any application installed with Apple Configurator are tied to the Apple ID it was imported with, which may have unintended consequences when it prompts for updates on every device.

This is especially true when the Apple ID has an e-mail address for the username that is not associated with your institution, because end users see it when prompted. We're not saying that this has happened to any of our customers.

If you have different groups that are sharing the same set of supervised devices, apps can go out and come back in if another setup is required where these apps shouldn't be present. Apple Configurator can group devices arbitrarily as you choose and apply settings as needed, and apps are one of the things that can come along for the ride.

These processes are just the same for paid apps that have been purchased under the VPP. It becomes very important, however, to follow Apple's guidance as to what version of VPP purchases should be chosen based on your use case. Also, you should be careful to not apply an app to a device if it has not been first put into the Supervise mode, as this will not allow you to reclaim the app code if you're relying on this method of app distribution.

While this is not necessarily pertinent for a security discussion, the online VPP portal from Apple provides an interface to download redemption codes for use with Apple Configurator, and it inquires internally how many of these have ever been applied to devices. The Apple Configurator interface helpfully provides feedback about how many have been redeemed per product and it provides a spreadsheet of codes as well. It may seem obvious, but do not use the same spreadsheet of codes with an MDM or other distribution methods.

**Mass restoring and naming of devices**

From a branding or support standpoint, having the icons consistently arranged with a standard home screen background is desirable. Although MDMs are supposedly gaining this functionality, the original way to do these customizations, whether in the Prepare or Supervise modes, is to create a backup. (Backups made from a device in one mode cannot be restored to another with Apple Configurator.) This often requires manual interaction and if you have an MDM, it would make sense to allow it to perform any applicable configurations. It's very straightforward in the interface where you would initiate the creation of a backup when you are in either mode, and you can even access the stored backups.

Apple Configurator also protects the throughput of the USB bus by limiting concurrent operations to somewhere in the range of three at a time.
Note that the application is limited to 30 concurrent USB connections over a powered hub, which is obviously not the maximum for the protocol.

Also, keep in mind that except with very recent, specialized hardware, USB hubs can practically be considered addressless except for physical identification. The most reliable way to be confident that devices on a large hub are being named or otherwise prepared in a particular order is to attach each cable to the device in the sequence that you like.

Note that if you supervised a device and it is lost, stolen, or broken to the point that it cannot reconnect to Apple Configurator, you will lose any applicable app codes if you are using VPP. (Which is to say the original "redemption codes" version in comparison to the licenses model referred to in the VPP portal as "managed distribution", for use with MDM.) To reclaim the previously supervised device's name to keep your inventory neat, you can select it from the list in Apple Configurator and under the Devices menu, hold down the Option key. Unsupervise will change to Remove and you can prepare a new device to take that slot in the sequence. The same goes when a device is repaired and replaced with a device that has a different serial number, if you were not able to unsupervise the previous device before it left your possession.

**Backup concerns**

When there is a supervision relationship between many of your devices and you realize that only small workgroups or sets of devices fit in the Apple Configurator usage model, backups become crucial, and alternatives to prevent over-reliance or an abundance of hacky workarounds become attractive. Taking backups as the first topic, Apple ships built-in backup software called Time Machine that can be used to protect the computer that runs Apple Configurator, but it is limited in its capabilities. You can either directly connect a hard drive (which can be encrypted), or send the backup over the local network to a machine running a compatible endpoint. It is not optimized for over-the-WAN offsite backup, among other shortcomings.

To separately understand the files in use, first we'll reprise our talk about sandboxing. In a rare reversal of the "do as I say, not as I do" maxim, Apple is following its own rules with Apple Configurator by using the container model for its data storage, which puts the files it operates with away from the view of the user. It is literally deep within a hidden folder. You can reach it by navigating to Users | CurrentUser (the current user's name) | Library | Containers | com.apple.configurator | Data | Library. Yes, the repetition is intentional.
Similar to Time Machine, Apple Configurator leverages links to refer to files outside of its sandbox for which it doesn’t need write access. (Time Machine uses hard links to stub unchanged files from previous backups, which lets it present a complete set when you browse the most current folder structure in its storage destination.)

Another repeated pattern is the use of SQLite as the storage mechanism for the database of supervised devices and other inventory-related information. This is located in a subdirectory of the path listed earlier and you can go to it by navigating to Application Support | com.apple.configurator | AppleConfigurator.storedata. iOS software updates that are often full OS installations get cached within Firmware under Caches and apps imported into the program get stored in Resources, which you can reach by navigating to Application Support | com.apple.configurator.

**Configurator as chaperone**

It is a common troubleshooting tip to turn up the verbosity of a process, look through the logs, and check any settings or configuration files. Mac folks have long gathered commands that enable hidden settings in preference files that are Apple-flavored XML files, just as we said were the case for configuration profiles. If you run defaults write com.apple.configurator LogLevel ALL (with the preference domain mapping to the path of com.apple.configurator.plist at Preferences by navigating to Users | CurrentUser (the current user's name) | Library | Containers | com.apple.configurator | Data | Library), you will cause informational text built into the debug output of the application to be written to logs. You can then sift through this information by viewing system.log in the Console application inside the Utilities folder in Applications, if you're running as an admin user on Mac. (Otherwise, you can tail the system.log file by navigating to var | log if you can elevate yourself to an admin user from a shell.)

Sometimes, old codenames for apps, devices, or features stick around in the inner workings of applications, and if you run defaults read on the preceding file (or open it in a binary plist compatible text editor such as Xcode), you’ll notice the ChaperoneCertificateIssuer and ChaperoneCertificateSerial key/value pairs. Supervision may very well have used this Chaperone naming internally at Apple during development. Similarly, the name of the profile that Apple Configurator installs when supervising the device is referred to as com.apple.configurator.chaperoneprofile. The following screenshot shows the settings on a supervised device; this is an example of Apple Configurator's installed profile:
In Settings on a supervised device, this is an example of what Apple Configurator's installed profile looks like.

In past versions of Apple Configurator, you would see that the console output also mentions the Boolean (true/false) value for the "chaperoned" property of a device that is being interacted with. This concept of a host having a responsibility relationship with the device helps further stress the importance of guarding the computer that is running Apple Configurator. If this machine is ever compromised, (or perhaps even worse, experiences data loss) you would be in quite a pickle indeed.

**Activation Lock and Find My iPhone**

A boon for theft prevention (or a bust for the iOS device resale market), is the implementation of a new feature, as of iOS 7, by Apple called *Activation Lock*, which is an extension of iCloud's previous Find My iPhone feature. If you had an iCloud account configured with the setting on an iOS 7 device and it needed to be reactivated from scratch after a restore, the process would not have been able to proceed until that account's password was entered. This was felt to be a burden and a management headache for those who lent out devices regularly, but by some municipality's statistics, this alone reduced theft of iOS devices as they became practically useless.
A few links to note
The citation for the claim that thefts (and the iPhone resale market) are impacted by this feature can be found at http://arstechnica.com/apple/2014/06/ios-7-activation-lock-cutting-iphone-theft-damages-resale-market/.

Apple's Check Activation Lock Status page at https://www.icloud.com/activationlock/ for use before you buy or receive a phone.


Apple, as the central clearinghouse of devices that must come onto the network and check in before being allowed to be activated, can theoretically ensure that devices can only be activated by their rightful owners.

To address the problem of institutions that want control over whether customers can enable this feature and do not find it desirable when they'd like to reprovision the device to another user, two techniques exist. The first one is that an MDM can block Activation Lock until a bypass code can be generated for the device and sent to the service for a certain window of time after an enrollment that is akin to a full disk encryption key escrow, which provides a distinct, non-identifying "get out of jail free" card so that you can reactivate the device without the presence of the previous iCloud-identified user. You can find more details at http://support.apple.com/en-us/HT202804 in Apple's documentation about how they recommend folks mix tools such as an MDM or Apple Configurator into their support procedures around Activation Lock.

The reference implementation of MDM for Apple, the Profile Manager service in their OS X Server app, has specific documentation on the Activation Lock bypass code at http://help.apple.com/profilemanager/mac/4.0/#/apd94BD5B2E-6448-450D-B76F-605ABEBEC9D7.

The other technique to deal with Activation Lock is that by default supervision does not allow this feature to be enabled in the first place. Are you getting the idea that Apple really wants you to supervise your devices? Only if you then use an MDM that enables the feature (via escrowing a bypass code or otherwise) can devices use the feature. Even if the end user enables Activation Lock on a supervised device, putting the device into Recovery mode will allow you to wipe (or prepare or refresh) it as you see fit. If you're given a device that was not supervised before Activation Lock was enabled, you will get an error message that says that it is "Unable to check iOS".
Recovery mode is a state where the device has booted to its firmware and has been told that it needs a fresh OS installation. It previously showed a Connect to iTunes message with a USB connector, but now it shows an arrow from a lightning connector to the new red iTunes icon (http://support.apple.com/en-us/HT1212). You can also use a utility like RecBoot or others if you often find yourself recovering a forgotten password, but be sure to carefully evaluate and inspect applications that purport to do cool things to iOS devices, as they are not officially sanctioned by Apple and may be from compromised sources (http://jaxov.com/2010/05/recboot-iphone-recovery-mode/). The following screenshot shows a prompt that displays the error encountered when you try to prepare a device with Activation Lock enabled:

The error presented when you try to prepare a device with Activation Lock enabled

Addressing the rough spots

For years, Apple said you could try a stick-and-carrot approach, using HR policy and enticements to stop end users from removing MDM or supervision profiles, with the ultimate caveat being that end users could always wipe the device. iOS 8 finally delivered a more comprehensive way to ensure that the devices are managed after being given to end users. Now, there is a restriction on access to the setting that erases all data and settings if the device is supervised, but only DEP, which we’ll discuss later, truly keeps the device locked to your MDM. You can also restrict the removal of profiles by setting passwords as needed for removal in an ad hoc manner.

Between the small (intended) workgroup scale, inflexibility regarding interaction with things like backups, and the singular, fat client-based point of failure, many have hoped that there were other options. GroundControl is a new product that can provide some of the powerful features and functionality of Configurator without its limitations. (Disclaimer: one of our technical editors is the lead developer on this project.) This cloud-based solution aims to put tight control of the deployment process in the hands of the stakeholders. You can learn more about this at https://www.groundctl.com.
DEP versus Apple Configurator

The Device Enrollment Program (DEP) is provided by Apple to alter the setup assistant so that devices can be unboxed by end users, but they are then forced to enroll into the MDM. DEP can also enable supervision without Apple Configurator. In fact, Apple recommends that you are not supposed to use devices that have DEP with Apple Configurator, at least while they are assigned to an MDM. Just as Activation Lock would cause trouble with Apple Configurator; DEP would like to kick in when the device is being activated, and this is not currently engineered into the product. Apple's documentation regarding the example use cases where DEP can be used with Apple Configurator is found at http://support.apple.com/en-us/HT201092.

To get going with DEP, a significant amount of paperwork is required such as associating Apple IDs, tracking down purchases, getting a D-U-N-S number if you don't already have one for your Apple Enterprise Developer account, and then connecting the DEP portal to your MDM. And even before all that, it may not be available in your country. The complete list of countries that have DEP can be found at https://deploy.apple.com/qforms/open/register/country/aws.

The actual moving parts for setting up DEP with your MDM are mostly concerned with what you want to see as part of the setup assistant. There is also the option to lock the MDM profile and enable supervision.

Keep in mind that things such as supervision and locking down devices shouldn't be a concern when you're only supporting a BYOD program. However, there are certainly many important considerations to keep in mind when you transition from previously deployed and supervised devices to DEP. Just like supervision, you must wipe the device so that it always points to your MDM during setup. This brings us to a bit of a show-stopper for many, and that is the fact that you are not supposed to restore the backup taken from the same device that is now being associated with DEP.

This makes it sound like there isn't a real migration path for pre-existing managed devices. We are not making this up. For more information, you can refer to http://support.apple.com/en-us/HT202977. You are even expected to MDM-wipe or Apple Configurator-unsupervise devices before they can be considered active within DEP. For moving data, the following choice quote is included under Apple Configurator: Transitioning to Apple Deployment Programs:
When an iCloud backup is restored to the same device, all supervision and profiles come from the backup regardless of how it was configured in the Device Enrollment Program. For this reason, when restoring backups each user should transition to a new or different device to ensure Device Enrollment Program supervision and MDM enrollment are enforced.

When we filed a radar (bug report) on this behavior, the response received "works as intended".

**Guided Access versus App Lock versus Single App Mode**

The previous section on **Guided Access** in Chapter 2, *Introducing App Security*, introduced us to the concept of putting the device into a mode where very little can go wrong with it, but this also limits it to a single purpose—locking the device to run only one app. Note that this would only be applicable for supervised devices. Apple Configurator can be told which app to run and the device will bypass the home screen after the device is woken from sleep. The previous guidance applies for making sure that you can get access to the Apple Configurator station in case it needs maintenance, or to make sure that the network access is reliable if using Single App Mode with MDM. In addition, ensure that the power settings are applied, as end users would need to put the screen to sleep manually since they don't have access to settings.

As Single App Mode allows ad hoc, over-the-air application of the profile to make the device enter this locked-to-app mode, you can first allow end users to set a passcode on the device before the home screen becomes inaccessible. While this allows it to remain locked when unattended, make sure you consider apps that prompt for authentication and allow you to log out if sensitive data or systems are to be used.

**ActiveSync**

You may get along very well without any of these tools that we've discussed so far. In addition, MDM is not particularly necessary if the ActiveSync protocol delivers the restrictions and security features that you need. The protocol was also adopted by paid versions of the Google Apps product and it is natively supported when you configure an Exchange e-mail account on iOS.
Organizational Controls

Many aspects of the server and Outlook Web Access interface work in exactly the same manner with iOS as they would with Blackberry, Symbian, Windows Mobile, Windows Phone, or an Android device. However, while the 14.0 version of the specification should be supported, the actual applicable settings have remained somewhat unchanged for years. Recently, Microsoft has been promoting various new products to manage mobile devices, which support the native management frameworks of each of the popular platforms.

As a refresher, management settings enforceable via the ActiveSync protocol are as follows:

- Wiping the device (if the device is lost or stolen)
- Enforcing a device passcode, with complexity, expiration, history, timeout before prompt, and failed attempt thresholds
- Allowing use of the camera (which was originally focused around courts or government-related buildings and contractors)
- Disabling sync while the device is roaming to help with data usage while you are outside normal cellular coverage

Further, via a configuration profile, you can limit how far in the past your mail is synced, along with other account-specific settings like certificates.

Summary

Over the course of this chapter, we spent a lot of time investigating Apple Configurator. We discussed the Prepare mode, which can make lightweight, one-off changes as per your need. Supervision and user check out or assignment sets up long-term management "chaperone" relationships with iOS devices. We went over how Apple Configurator distributes the older version of VPP app codes and how it can lock the device into an app. As Activation Lock helped to make a device's theft become less effective, supervision also provided a safety net for institutions by allowing them to reclaim devices via the Recovery mode. We also reminded you that before evaluating an MDM, many restriction-related features are actually available to ActiveSync as an alternative.
For security professionals, it may seem like Apple is clueless about the needs of large enterprises, and Apple Configurator may not help with that impression. But by providing best practices we’re left with the most supportable management, which works with the platform instead of against it. Apple has pushed the idea of "tier zero" or "the new IT" as a hands-off, infinitely scalable solution where IT lets end users perform maintenance tasks and it doesn't need to build walls between work and personal data in everyone's devices. We can do our best work when we are protecting devices by concentrating on how little of the device needs to be managed, even if they are owned by institutions. Even when it seems that the controls that are available aren't of industrial strength, practical concerns are going to trump a tightly locked-down experience. Apple, its customers, and its developers still need room to experiment and bring real innovation and productivity to mobile devices.
Where to buy this book
You can buy Learning iOS Security from the Packt Publishing website.
Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.
Click here for ordering and shipping details.