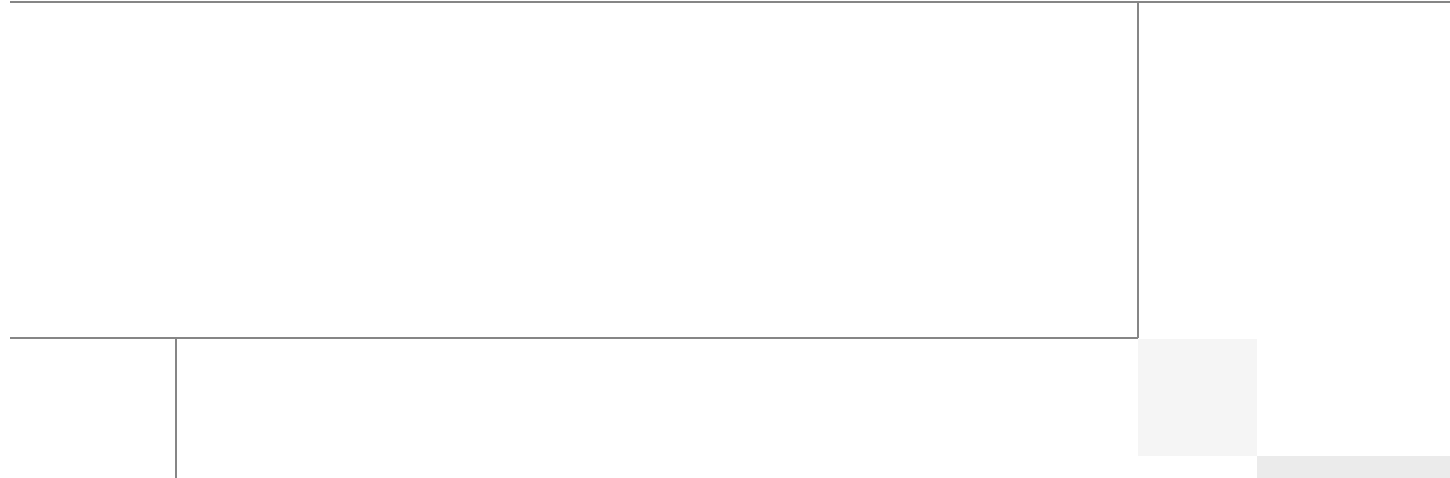




Integrating MobileIron with Cisco Identity Services Engine

Revised: August 6, 2013



ALL DESIGNS, SPECIFICATIONS, STATEMENTS, INFORMATION, AND RECOMMENDATIONS (COLLECTIVELY, "DESIGNS") IN THIS MANUAL ARE PRESENTED "AS IS," WITH ALL FAULTS. CISCO AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE. IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THE DESIGNS, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

THE DESIGNS ARE SUBJECT TO CHANGE WITHOUT NOTICE. USERS ARE SOLELY RESPONSIBLE FOR THEIR APPLICATION OF THE DESIGNS. THE DESIGNS DO NOT CONSTITUTE THE TECHNICAL OR OTHER PROFESSIONAL ADVICE OF CISCO, ITS SUPPLIERS OR PARTNERS. USERS SHOULD CONSULT THEIR OWN TECHNICAL ADVISORS BEFORE IMPLEMENTING THE DESIGNS. RESULTS MAY VARY DEPENDING ON FACTORS NOT TESTED BY CISCO.

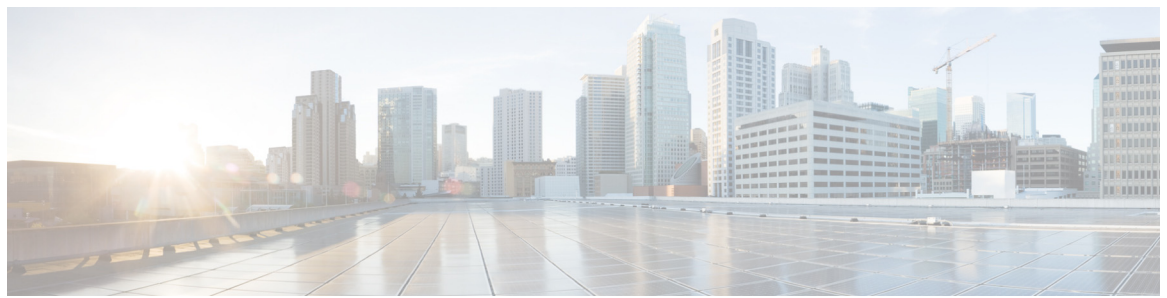
The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at <http://www.cisco.com/go/trademarks>. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Integrating MobileIron with Cisco Identity Services Engine

© 2013 Cisco Systems, Inc. All rights reserved.



Integrating MobileIron with Cisco Identity Services Engine

This document supplements the Cisco Bring Your Own Device (BYOD) CVD (http://www.cisco.com/en/US/docs/solutions/Enterprise/Borderless_Networks/Unified_Access/BYOD_Design_Guide.html) and provides mobile device management (MDM) partner-specific information as needed to integrate with Cisco ISE. In an effort to maintain readability, some of the information presented in the CVD is repeated here. However this document is not intended to provide standalone BYOD guidance. Furthermore, only a subset of the MobileIron MDM functionality is discussed. Features not required to extend ISE's capabilities may be mentioned, but not in the detail required for a comprehensive understanding. The reader should be familiar with the MobileIron Administrator's guide.

This document is targeted at existing MobileIron customers. Information necessary to select an MDM partner is not offered in this document. The features discussed are considered to be core functionality present in all MDM software and are required to be compatible with the ISE API.

Overview

MobileIron is a leading provider of MDM software used to establish and enforce device policy on hand-held endpoints. This could include corporate- or employee-owned phones and tablets. Devices manufactured by all the major equipment providers are supported at some level. Apple iOS and Android devices are the primary focus, but MobileIron also supports Blackberry, Win8, and Apple's OS X Lion and Mountain Lion software (10.7 and 10.8, respectively).

Mobile Device management is a relatively new phenomenon and is in a constant state of expansion. Features can be grouped into several categories:

- **Device Restrictions**—There are two common types of restrictions. Either some feature of the device is disabled, such as the camera, or there are additional requirements for basic usage, such as a PIN lock or storage encryption. When a restriction is in place, the user is not offered the choice of non-compliance. Restrictions are used to reduce security risks to the enterprise.
- **Device Compliance**—This may also be referred to as posture enforcement. The MDM will check the attributes of the device against a list of acceptable operational conditions. Compliance checks can be enforced based on their severity. For example, an email could be sent to the user when they have exceeded 80% of their data plan or MobileIron can automatically issue a corporate wipe if the



Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Copyright © 2013 Cisco Systems, Inc. All rights reserved.

device has been compromised. A compliance check is different from a restriction because the user can take the device out of compliance. Compliance can be used to increase security or reduce operational costs.

- **Notifications**—Administrators can send a message to a large population of devices. This could be a push message to the device notification page. For example, “The fire drill is complete, you may return to the building” could be sent to all devices on a particular campus. Notifications are used to increase productivity.
- **Content Distribution**—Bookmarks, documents, and other content can be pushed to devices in the background, with or without the user intervention, or made available on demand. This data is then stored in a corporate container. Content distribution is used to increase productivity.
- **Application Distribution**—The MDM can offer a company catalog of available software or install required software. The software can come from public repositories or can be corporate developed applications. Application distribution has both a security and productivity gains. Security is enhanced because any software distributed by the MDM, including local storage associated to the software, is removed as part of a corporate wipe. This is not true if the user installs the same software from the Apple App Store.

MobileIron’s MDM solution has three main components:

- Policy server
- Device OS API
- Device client software

Beyond these, there are additional components for enterprise integration with systems such as email, Web, and secure corporate data. The majority of the base functionality is available through the MDM API built into the mobile device operating system. MobileIron requires the client software to detect some conditions such as jail-broken¹ or rooted devices. Because ISE tests for these conditions, the MobileIron server is configured to treat the client software as a required application and will install the software during the onboarding process.

MobileIron also offers a scripting engine (Assemble) that can provide additional flexibility beyond what is shown here. This is an advanced tool appropriate in specific scenarios. It is mentioned here to reinforce that this document is not a full exploration of all of MobileIron’s capabilities.

Deployment Models

MobileIron offers its Virtual Smartphone Platform (VSP) as both an on-premise model and a cloud service model. The two models are functionally equivalent. The CVD explores the advantages and disadvantages of each of the models. An obvious difference is the topology. An on-premise model is defined when the MDM server is located in the enterprise DMZ and managed directly by the enterprise. A cloud model places the MDM server in the cloud and is offered as a software subscription. Both models support integration with corporate services such as corporate directories, exchange, or a Blackberry Enterprise Server. The cloud model provides directory integration with MobileIron’s Connector Server. Currently Connected cloud does not support SCEP. Although the use cases in the CVD do not require SCEP from the MDM, other scenarios not covered here could benefit from this functionality. With VSP 5.6, an on-premises deployment would be required if SCEP via the MDM is a requirement.

1. Apple prefers the term “compromised OS” when referring to systems where the user has gained root access to the operating system.

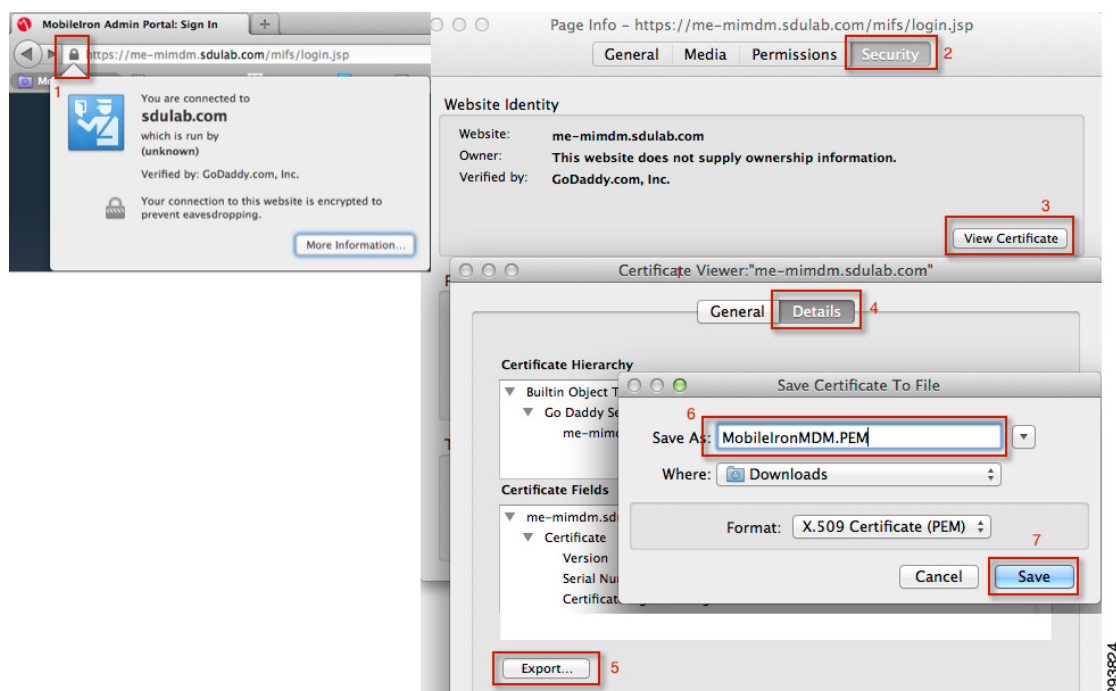
Getting MobileIron Ready for ISE

The first requirement is to establish basic connectivity between the Cisco ISE server and the MobileIron MDM server. In both the on-premise and the cloud model, a firewall is typically located between these two components. The firewall should be configured to allow an HTTPS session from ISE located in the data center to the MDM server located in either the corporate DMZ or public Internet. The session is established outbound from ISE towards the MDM where ISE takes the client role. This is a common direction for Web traffic through corporate firewalls.

Import VSP Certificate to ISE

The MobileIron MDM server incorporates an HTTPS portal to support the various users of the system. In the case of a cloud service, this website will be provided to the enterprise. ISE must establish trust with this website. Even though the cloud website is authenticated with a publicly signed certificate, ISE does not maintain a list of trusted root CAs. Therefore, the administrator must establish the trust relationship. The simplest approach is to export the MDM site certificate then import the certificate into local cert store in ISE. Most browsers allow this. Firefox is shown in [Figure 1](#). Note that MobileIron supports Firefox 14 and Internet Explorer 8.

Figure 1 Export MobileIron Web Certificate

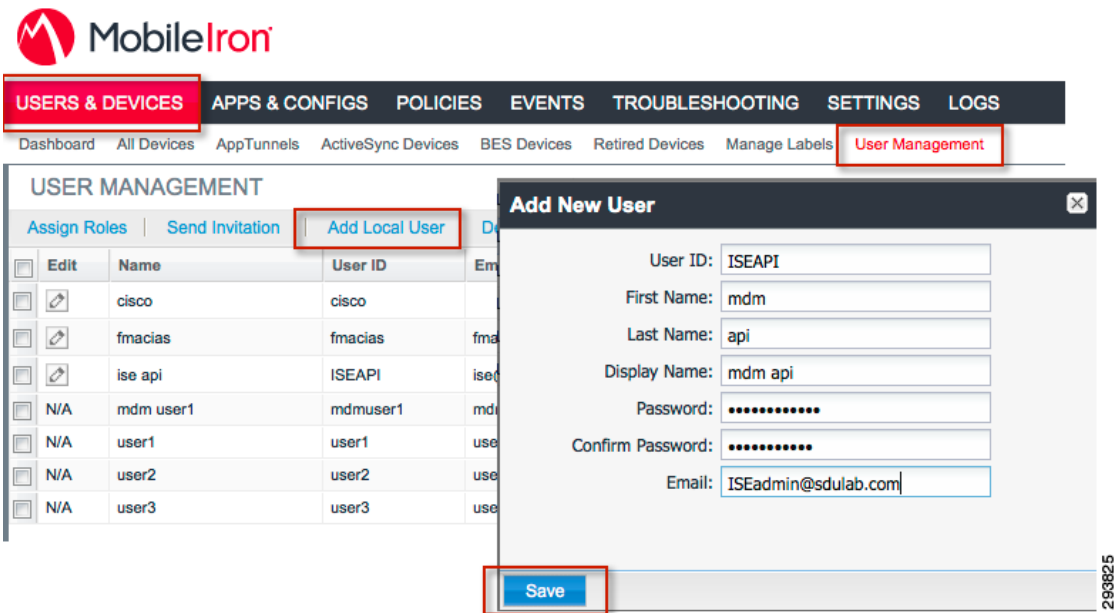


In case of an on-premise deployment, the site certificate will need to be signed by a trusted CA and installed on the MDM server prior to importing the certificate to ISE. This task is completed from the System Manager portal and is well documented by MobileIron.

Grant ISE Access to the MobileIron API

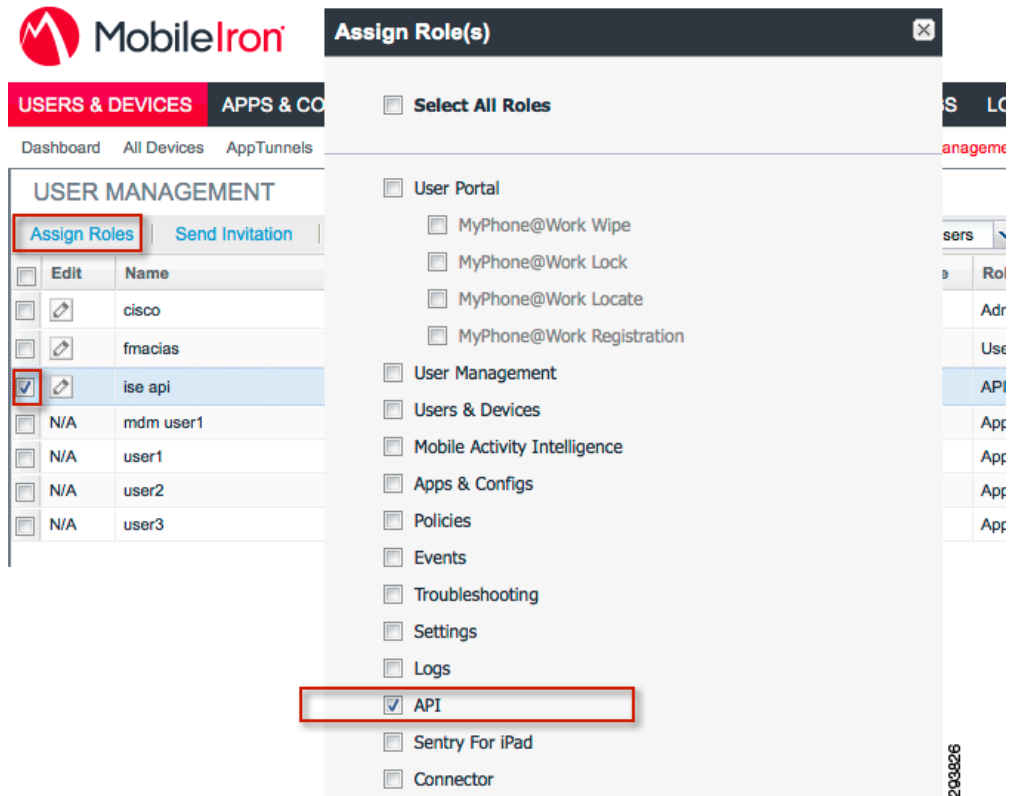
The MobileIron MDM API is protected by HTTPS and requires a user account that has been granted permission to the API. Ideally a specific account would be configured for ISE with a very strong password. In addition to this account, only a limited number of administrator accounts should be granted the ability to create new administrators or assign administrator roles. Creating local user accounts is accomplished in the User and Device section of the administrator console, as shown in [Figure 2](#).

Figure 2 Create ISE User Account



Once the account has been created, it is assigned roles to allow ISE access to the MDM API.

Figure 3 Assign API Role to ISE Account



Add MDM Server to ISE

Once the account has been defined on the MobileIron MDM server with the proper roles, ISE can be configured to use this account when querying the MDM for device information. ISE will contact the MDM to gather posture information about devices or to issue device commands, such as corporate wipe or lock. The session is initiated from ISE towards the MDM server. The URL for the MobileIron server is the same as the admin page and used earlier to export the certificate. The directory path is handled automatically by the system and is not specified as part of the configuration. The Instance Name field is used when subscribing to a MobileIron cloud service and uniquely identifies the cloud partition. MobileIron will provide the Instance to be used. The field should be left blank for an on-premise deployment. The port will should be configured for (TCP) 443 for HTTPS. The MDM cannot be configured to listen on a specific port for API users and any change will also impact both the admin and user portal pages.

Figure 4 *Configure the MDM API on ISE*

The screenshot displays the Cisco Identity Services Engine (ISE) Administration interface. The top navigation bar includes 'Home', 'Operations', 'Policy', and 'Administration'. The left sidebar shows 'Mobile Device Management' with 'External MDM Servers' selected. The main content area is titled 'External MDM Server List > mobileiron' and contains the 'MDM Server details' form. The form fields are as follows:

| Field | Value |
|--------------------|-------------------------------------|
| * Name | mobileiron |
| * Server host | me-mimdm.sdulab.com |
| * Port | 443 |
| Instance Name | |
| * User Name | ISEAPI |
| * Password | ***** |
| Description | MobileIron MDM |
| * Polling Interval | 0 (minutes) |
| Enable | <input checked="" type="checkbox"/> |

Below the form is a 'Test Connection' button. At the bottom of the form are 'Save' and 'Reset' buttons. The 'MDM' tab in the top navigation bar is highlighted with a red box.

293827

The polling interval specifies how often ISE will query the MDM for changes to device posture. Setting the value to zero will disable polling. Polling is used to periodically check the MDM compliance posture of an end station. If the device is found to be out of MDM compliance and the device is associated to the network, then ISE will issue a CoA, forcing the device to re-authenticate. Likely the device will need to remediate with the MDM although this will depend on how the ISE policy is configured. Note that MDM compliance requirements are configured on the MDM and are independent of the policy configured on ISE. It is possible, although not practical, to set the polling interval even if the ISE policy does not consider the MDM_Compliant dictionary attribute.

The advantage of polling is that if a user takes the device out of MDM compliance, they will be forced to reauthorize that device. The shorter the window, the quicker ISE will discover the condition. There are some considerations to be aware of before setting this value. The MDM compliance posture could include a wide range of conditions not specific to network access. For example, the device administrator may want to know when an employee on a corporate device had exceeded 80% of the data plan to avoid any overage charges. In this case, blocking network access based solely on this attribute would aggravate the MDM compliance condition and run counter the device administrator's intentions. In addition, the CoA will interrupt the user WiFi session, possibly terminating real-time applications such as VoIP calls.













The polling interval is a global setting and cannot be set for specific users or asset classes. The recommendation is to leave the polling interval at 0 until a full understanding of the MDM's configuration is attained. If the polling interval is set, then it should match the device check-in period defined on the MDM. For example, if the MDM is configured such that devices will report their status every four hours, then ISE should be set to the same value and no less than half of this value. Over sampling the device posture will create unnecessary loads on the MDM server and reduced battery life on the mobile devices. There are other considerations with respect to scan intervals. Changing MDM timers should be done only after consulting with MobileIron's best practices.

Verify Connectivity to MDM

The Test Connection button shown in [Figure 4](#) can be used to isolate and resolve common problems prior to developing MDM-based authorization policy. ISE will attempt to log in to the API and report back the result. Completing the test successfully is required prior to saving the settings. If the test does not complete successfully, the settings can still be saved, but the Enable box will be deselected and the MDM will not be active.

Some common problems found while testing the connection to the MDM server are shown in [Table 1](#).

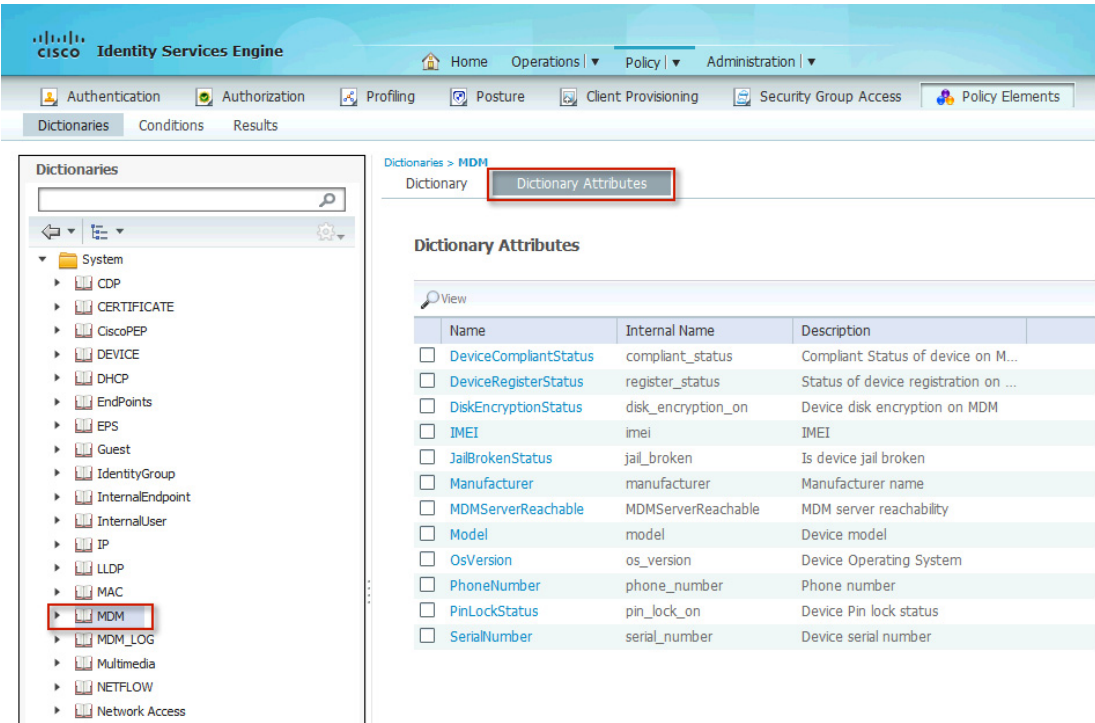
Table 1 **Connection Messages**

| Message | Explanation |
|---|---|
|  Connection Failed: Please check the connection parameters.  | A routing or firewall problem exists between the ISE located in the data center and the MDM located in either the DMZ or Cloud. The firewall's configuration should be checked to confirm HTTPS is allowed in this direction. |
|  Connection Failed 404 : Not Found  | The most likely cause of an HTML 404 error code is that an instance was configured when it was not required or that the wrong instance has been configured. |
|  Connection Failed 403 : Forbidden  | The user account setup on the MobileIron server does not have the proper roles associated to it. Validate that the account being used by ISE is assigned the REST API MDM roles as shown above. |
|  Connection Failed 401 : Unauthorized  | The user name or password is not correct for the account being used by ISE. Another less likely scenario is that the URL entered is a valid MDM site, but not the same site used to configure the MDM account above. Either of these could result in the MobileIron server returning an HTML code 401 to ISE. |
|  Connection Failed: There is a problem with the server Certificates or ISE trust store.  | ISE does not trust the certificate presented by the MobileIron website. This indicates the certificate was not imported to the ISE certificate store as described above or the certificate has expired since it was imported. |
|  The MDM Server details are valid and the connectivity was successful.  | The connection has successfully been tested. The administrator should also verify the MDM AUTHZ dictionary has been populated with attributes. |

Review MDM Dictionaries

When the MobileIron MDM becomes active, ISE will retrieve a list of the supported dictionary attributes from the MDM. Currently MobileIron supports all of the attributes that ISE can query. The dictionary attributes are shown in [Figure 5](#).

Figure 5 Dictionary Attributes



Enterprise Integration

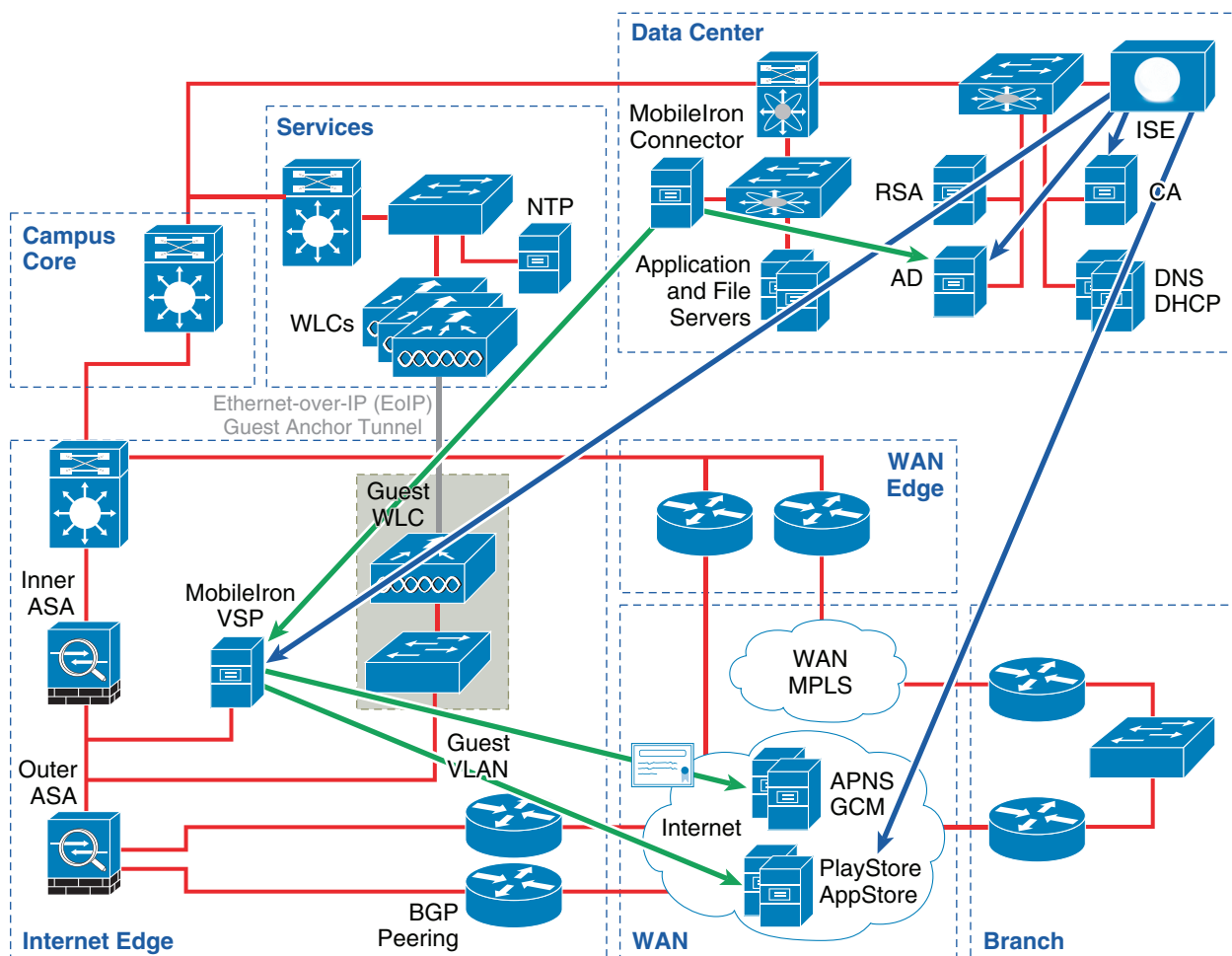
Both ISE and MDM must be integrated into a common enterprise environment. At the basic level, this involves sharing the same directory structure. A common directory simplifies the operational aspects of the overall system, but also allows a consistent policy structure around AD group membership. For example, if a user is a member of the AD group FULL_ACCESS, that membership should result in a policy from both ISE and MDM that is consistent with the group and cognizant of the other component. If the MDM installs an application on a device, then ISE should allow the application on the network for members of that AD group.

MobileIron offers a Connector component. This is ideally suited for a cloud model, but could also be used in on-premise scenarios where the MobileIron server resides in the DMZ, but does not have access to all enterprise services, such as LDAP. The service establishes outbound HTTPS connections to the MobileIron server running in the cloud. The connector will periodically update the VSP with directory information that can be used to develop an integrated device policy. The installation is straightforward and fully documented by MobileIron.

On-premise deployment models can also make use of the Connector server. Most often, the MobileIron VSP server is located in the DMZ. However this requires that the enterprise allow LDAP from the DMZ through the firewall. Company policy may not permit this. In this case, the

Connector is deployed behind the firewall, eliminating the need to allow LDAP inbound from the DMZ. In addition to LDAP, the VSP server will establish outbound connections to the Apple Push Notification Services (APNS) and Google Cloud Messaging (GCM). The MobileIron Administrator's guide provides information on what firewall policies are required to provide access to the Internet push servers.

Figure 6 *MobileIron Topology with On-Prem Connector*



Socket Requirements

There are several flows that need to be allowed between the various components. The full list is available from MobileIron or the device vendors. [Table 2](#) summarizes the required sessions.

Table 2 *Common Socket Requirements*

| Source | Destination | TCP Port | Purpose | Comment |
|--------|-------------|----------|-----------------------------|---------------------|
| MDM | APNS | 2195 | Apple Push Notification | Cert Required |
| MDM | APNS | 2196 | Apple Push Feedback Service | APNs Message Status |

Table 2 **Common Socket Requirements**

| Source | Destination | TCP Port | Purpose | Comment |
|-----------------------|--------------|-----------|-------------------------|--------------------------|
| MDM | GCN | 5228 | Google Push | |
| MDM | LDAP (sLDAP) | 389 (636) | Directory | |
| Mobile Device | ISE | 8443 | Captive Portal | On-Boarding, Remediation |
| Mobile Device | MDM | 8080 | Provisioning | Configurable |
| Mobile Device | MDM | 9997 | Sync TLS | Configurable |
| Mobile Device | MDM | 9998 | Help Desk | Configurable |
| Mobile Device | MDM | 9999 | Sync Service | Configurable |
| Mobile Android Device | GCM | 5228 | Google Cloud Messaging | |
| Mobile iOS Device | APNS | 5223 | Apple Push Notification | |

AD/LDAP Integration

With either an on-premise or cloud model, integrating ISE and the MDM to a common directory is important for the overall operations. One benefit is the ability to set a requirement that a user periodically change their directory password. If the MDM were using a local directory, it would be nearly impossible to keep the accounts in synchronization. But with a centralized directory structure, password management can be simplified. The main advantage is the ability to establish complementary network and device policy based on group membership. The CVD provides examples of how AD groups can be used to establish a user's entitlement to network resources. Likewise, the same group membership can be used to differentiate access to device resources and mobile applications.

Prior to configuring policy based on AD group membership, the VSP must have a binding to LDAP. This is done from the LDAP Setting page, as shown in [Figure 7](#).

Figure 7 Adding LDAP to Integrate AD Groups

The screenshot shows the MobileIron web interface. The top navigation bar includes 'USERS & DEVICES', 'APPS & CONFIGS', 'POLICIES', 'EVENTS', 'TROUBLESHOOTING', 'SETTINGS' (highlighted), and 'LOGS'. Below this, a secondary navigation bar lists 'Preferences', 'Sentry', 'Connector', 'LDAP' (highlighted), 'BES Servers', 'Operators', 'Local Certificate Authorities', 'Service Diagnostic', and 'Templates'. The main content area shows the 'LDAP' configuration page with a table of LDAP entries. The 'Add New' button is highlighted. A modal window titled 'Modifying LDAP Setting' is open, showing the following fields:

- Directory Connection**
 - Directory URL:
 - Directory Fallback URL:
 - Directory UserID:
 - [Change Password](#)
 - Search Results Timeout: Seconds
 - Chase Referrals: ☐ Enable ☒ Disable
 - Admin State: ☒ Enable ☐ Disable
 - Directory Type: ☒ Active Directory ☐ Domino ☐ Other
 - Domain:
- Directory Configuration - OUs**
 - OU Base DN:
 - OU Search Filter:
- Directory Configuration - Users**
 - User Base DN:
 - Search Filter:

At the bottom of the modal are buttons for 'Test', 'Save', and 'View LDAP Browser'.

AD Group Memberships

Three possible AD groups are presented in the CVD to illustrate their usage—Domain Users, BYOD_Partial_Access, and BYOD_Full_Access. ISE establishes the device’s network access based on the associated user’s membership.

Figure 8 shows the policies presented in the CVD.

Figure 8 CVD Use Policies

| Policy | AD Group | ISE | Compliant MDM | Permission | |
|------------------------|---------------------|-----|---------------|---------------|---|
| Personal_FullAccess | BYOD_Full_Access | YES | YES | Full | ✓ |
| Personal_PartialAccess | BYOD_Partial_Access | YES | YES | Partial | ⚠ |
| Personal_InternetOnly | Domain Users | YES | YES | Internet Only | 🌐 |
| Corporate Devices | | YES | YES | Full | ✓ |

These groups can be extended to the MDM such that members are issued profiles that complement their level of network access. As an example, Table 3 shows some arbitrary policies that can be established and enforced based on the CVD use cases.

Table 3 **Policies Based on CVD Cases**

| Ownership | User Group | Restrictions |
|------------------------|---------------------|---|
| Employee-Owned Device | Domain Users | Internet Only, personal devices are not required to on-board with the MDM. |
| | BYOD_Partial_Access | Fairly restrictive policy that isolates corporate data into containers. Restrictions prevent users from disabling the policy. |
| | BYOD_Full_Access | Trusted users are offered a slightly less restrictive policy. Corporate data is still isolated in containers. |
| Corporate-Owned Device | All Users classes | Very restrictive device policy disabling non-essential business functions such as the game center. |

Domain_Users is the default AD group. By definition, every user defined in the directory is a domain user. While it is possible to create the reciprocal group on the MDM, it is not needed. The CVD treats non-domain members as temporary guests that are unlikely to need MDM management. More important, if a user is not a domain member, then the MDM administrator will need to define a local user account. This is likely a very small set of users that are handled as an exception, such as distinguished guests. Domain_Users are essentially everyone with an account on the MDM, including members of BYOD_Partial_Access and BYOD_Full_Access.

MDM profiles and ISE AuthZ rules are fundamentally different with respect to AD Groups. ISE policy may include the AD group match as a condition for establishing a specific and single policy. MDM profiles are not a singular result. Most devices will be provisioned with multiple profiles based on various attributes. Members of the BYOD_Full_Access and Domain_Users can each be configured for a specific profile. But if a user happens to have membership in both BYOD_Partial_Access and BYOD_Full_Access, then that user's device is provisioned with both profiles. In addition, everyone will be provisioned with basic security restrictions. ISE will check the device to ensure these restrictions are met before granting network access. These restrictions establish ISE compliance and are defined here as required PIN lock, encrypted storage, and non-jail broken or rooted device.

MDM Profiles

Apple and Android differ in how device management is implemented.

Apple defines profiles that are an important concept of mobile device management. They are a foundational component of Apple's mobile device management protocol that is implemented by the operating system. The concept can be extended to application profiles, but as discussed here, they are found under the settings of the device. Each profile can contain one or more payloads. A payload has all the attributes needed to provision some aspect of built-in system functions, such as PIN lock. One special payload is the MDM payload that defines the MDM server as the device administrator. There can only be one MDM payload installed on any device. In iOS 5 and earlier, the profile containing the MDM payload cannot be locked and the user is free to delete it at any time. When this occurs, all other profiles installed by the MDM are also removed, essentially resulting in a corporate wipe. The MDM may lock any profile that it installed to prevent the user from removing them individually. The MDM is allowed to inspect other profiles such as the WiFi profile installed by ISE, but is not allowed to remove any profile that it did not install, including

the WiFi as detailed in the BYOD CVD. Because multiple profiles can be installed on a device and profiles have payloads, it is possible to have a payload collision. Devices with multiple security payloads will install all the payloads by aggregating the most secure settings from each. In most other cases the first payload is installed and subsequent payloads are ignored or multiple payloads are accepted. For example, the device can have multiple VPNs provisioned but only one can be named XYZ.



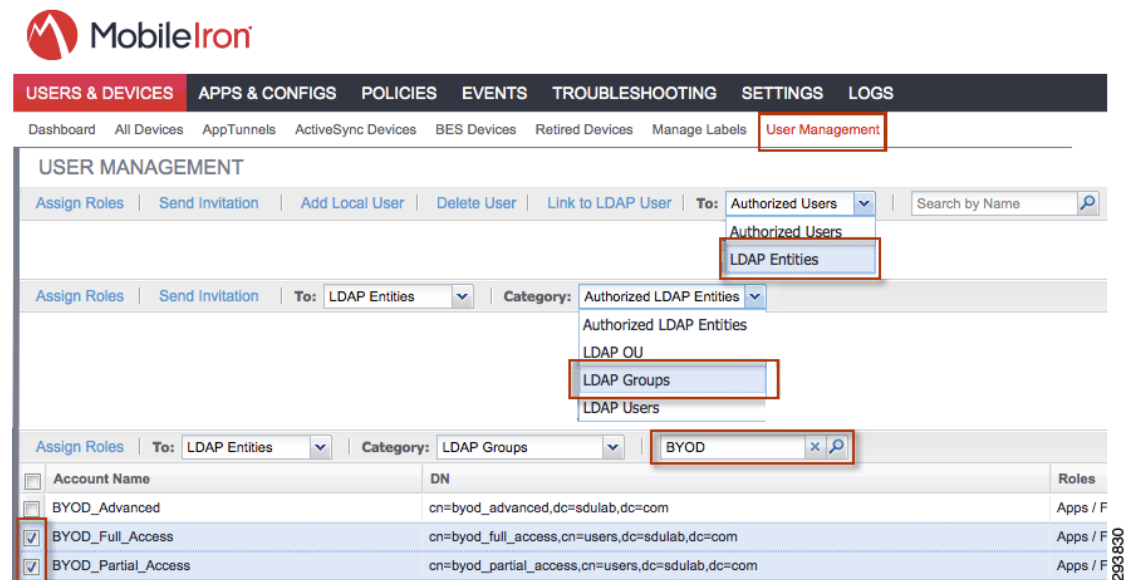
Note

Starting in iOS6, Apple does allow the MDM payload to be locked if the user has not set a PIN lock, although many MDMs—including MobileIron—are not yet supporting locked MDM profiles.

Android devices generally implement device management functions through a specific set of APIs, most of which are manufacturer- or model-specific. For example, Samsung uses their SAFE API, while HTC uses its One APIs. The results it that there are some variances in terms of what can be configured. For the most part, the MobileIron console handles device profiles in a unified way such that Android and Apple devices are configured in the same manner.

MDM profiles can be applied to devices associated to users that belong to a user group. Configuring this with MobileIron first requires the creation of the user group. Afterwards, a profile can be distributed to that user group. This is shown in [Figure 9](#).

Figure 9 Add User Groups from AD



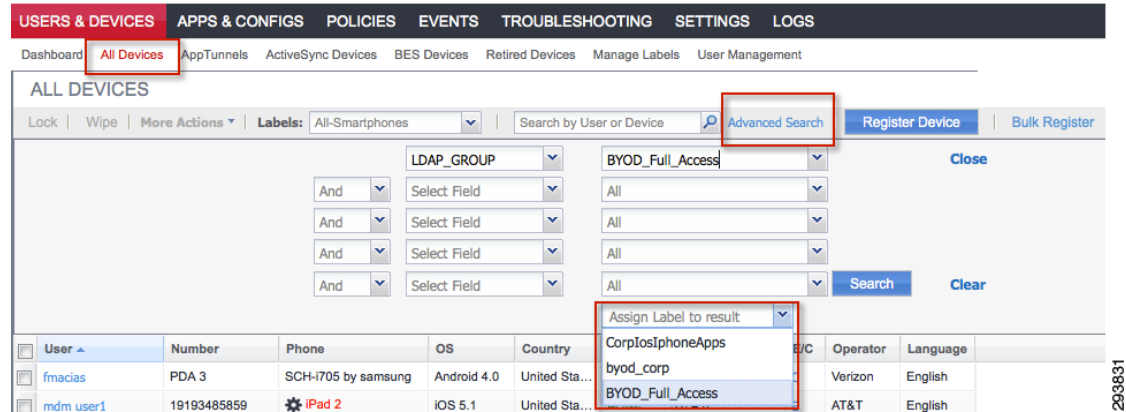
Using Labels to Form Groups

MobileIron provides labels to create virtual grouping of objects found on the system. Labels are a key component used to distribute policy to a select subset of devices and form the logical structure of policy management. There are several ways to configure labels to create policy. The examples presented here for were created for academic purposes to illustrate how the MDM and ISE can integrate together to form a comprehensive enterprise wide policy.

A label must be created before it can be applied. Labels are managed under the Users & Devices menu. There are several built-in label groups that should not be modified, including Company-Owned and Employee-Owned. Each label has an associated matching criterion that defines the devices or users to

which that label will be applied. Once a label has been created, it can be applied dynamically to AD group members. This is done by with the advanced search tool that can assign a label to the results.

Figure 10 *Assign Label to Active Directory Group*



With the example configuration shown above, users that belong to BYOD_Full_Access will be assigned a label with the same name.

Device Ownership

BYOD by definition mixes personal and corporate devices together on the same network. A key component is the ability to set device policy that is appropriate for both the user and the ownership of the device. The built-in ownership labels are used to accomplish this. The ownership of a device is generally declared during enrollment. MobileIron offers several enrollment models, including admin-assisted registration, in-app registration, and employee registration for additional devices. The ISE workflow utilizes the In-App registration model. In this model, the client application is loaded onto the phone first and the user enrolls through that application. Devices enrolled through the ISE workflow will default to corporate ownership. Only the administrator can change a devices ownership after the device has been enrolled. Future releases are likely to improve the tools available to monitor device ownership.

Both the ISE and the MDM have the concept of asset classes. This allows corporate devices to be distinguished from all other devices in the system. Ownership is an important aspect of BYOD. For example, support staff should not be allowed to issue a Full_Wipe of personal devices or track the devices location. However, corporate devices may get full wipes as a matter of normal operation and may be used to track location, especially if travel is a key component of the job. Having the ability to handle the information gathered from personal and corporate devices differently is important.

In this first release, there is not a tight integration between assets classed defined on ISE and those defined on the MDM. The API does not support such a device attribute. Complicating matters somewhat is the unique key index used to identify objects. Within ISE, this is the device's MAC address that is unique across the network. The MDM uses the devices UDID, which is unique over all devices. ISE determines corporate devices through an identity group referred to as the Whitelist, which contains all the MAC addresses of corporate assets. Discovering the MAC address of Android and Apple devices is typically a manual process. Apple lists the MAC on the

Settings > General > About page. MobileIron does allow devices to be bulk-imported into the system using a device serial number or phone number. An enterprise may need to create a list of corporate MAC addresses and the associated serial numbers to bulk-import them as corporate devices on both systems.

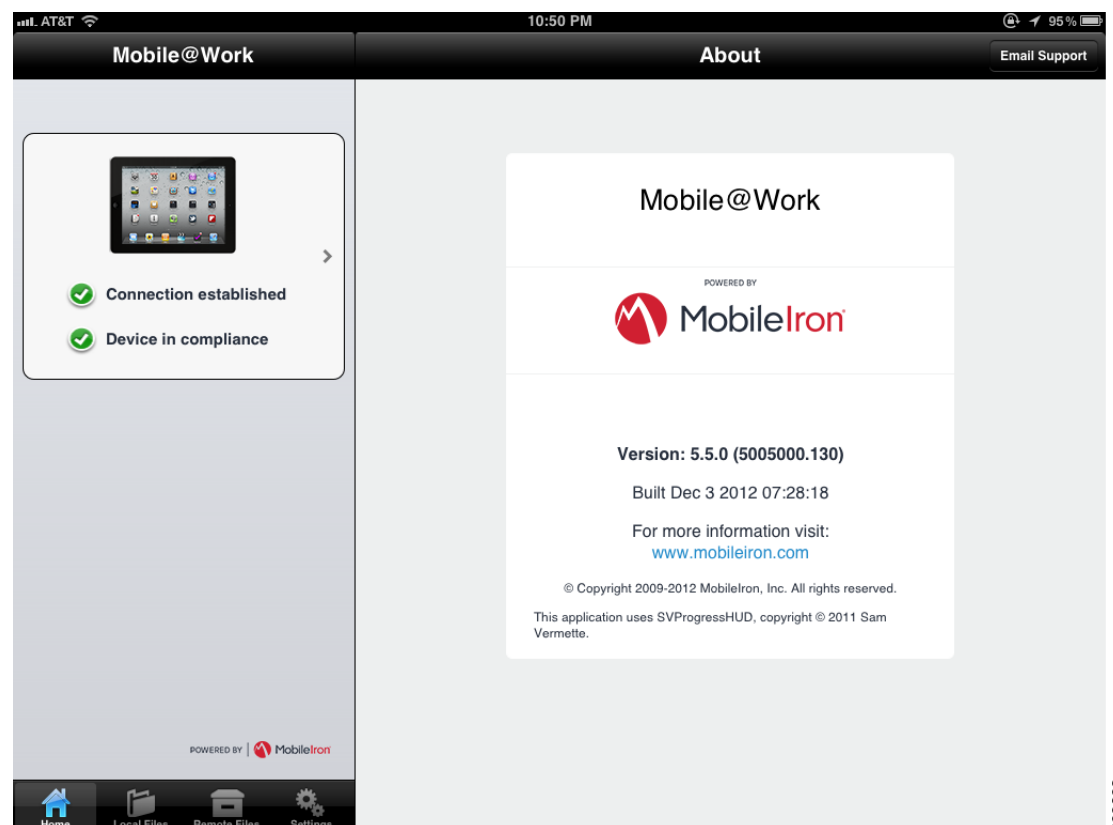
SCEP

The MobileIron MDM can provision certificates onto the device via SCEP. This allows profiles to contain a payload that provisions a service that requires authentication via a certificate and another payload contains the associated certificate. One such example is VPN payload for either AnyConnect or Cisco IPsec. This is discussed in more detail in [Application Distribution](#).

Mobile Client App

The majority of MDM features are implemented directly through the operating system and do not require the mobile device client application. However some of the advanced functionality does require a client running on the phone. In particular, jailbreak and rooted detection require the MDM client. Because ISE depends on these features for policy enforcement, the MobileIron client is a mandatory application, loaded during the ISE on-boarding workflow, and used to enroll the device with the MDM.

Figure 11 *Mobile Client @Work Application*



The Mobile@Work application offers the end users some useful information concerning the status of their devices. For example, users can measure the performance of their WiFi by measuring the network bandwidth: the Test Download Speed is launched from the Mobile Activity Map found under the Settings menu.

Another useful feature is the ability to manually refresh the device's posture to the server. The need arises when the device has been placed in MDM quarantine due to a compliance violation. For example, the device may not have a PIN lock when one is required. When the user configures the device with a PIN lock, the OS will not trigger an update to the MDM server. The change will be detected the next time the device is polled. This could result in ISE continuing to place the device in quarantine even after the user has corrected the issue. Rather than waiting for the MDM to poll the device for an update, the user could use the mobile application to initiate a forced check-in with the server.

In addition, the Mobile@Work client application contains two file folders, as seen in [Figure 11](#). One is used for local files and attachments, and the other allows access to a remote share. These folders are part of the Docs@Work feature set, which provides e-mail attachment control and remote SharePoint access. The MobileIron Administrator's guide has more information concerning the use of these features.

Creating Structured Policy

As mentioned earlier, labels provide a method to create groups based on specific attributes, such as AD group membership. This section explains how policy can be bound to labels to create a device policy that extends the use cases presented in the CVD. There are four specific BYOD use cases:

- Corporate devices are allowed full network access.
- Basic domain users are allowed Internet only access on their personal devices.
- Members of AD Group BYOD_Partial_Access are allowed access to specific network locations when using their personal device.
- Members of AD Group BYOD_Full_Access are allowed full network access when using their personal device.

MobileIron can bind profiles to labels, effectively binding policy to groups of devices and users. This can be used to effectively extend the use cases in the CVD to cover device policy. There are a few caveats to discuss. First, it is not possible to combine labels into Boolean structures (like AND, OR, NOT). If a device is corporate owned, it will get the policies for corporate devices. If the associated user of the corporate device is also a member of the AD group Partial_Access, they will get the policies associated to the label Partial_Access.



Note

These labels were named based on the BYOD CVD use cases and are not meant to imply MDM profiles will influence network access.

It is still possible to create effective policy that compliments the CVD after understanding how policies are combined together when two or more profiles are assigned to the same device based on different label memberships. As specified in the CVD, the AD groups BYOD_Full_Access and BYOD_Partial_Access apply to personal devices. But in this case, the dynamic labels based on AD groups will be applied to both corporate- and employee-owned devices. This works with the CVD use cases when restrictions placed on corporate devices are more restrictive than those

placed on employee owned devices. When an end device receives two policies, each with restrictions, the device will aggregate and keep the more restrictive settings. [Table 4](#) shows how restrictions are combined to determine the device's effective restrictions.

Table 4 *Aggregation of Restrictions*

| Profile | PIN lock | Encryption | Disable Cloud Sync | Disable Camera | Disable Music | Disable Games |
|-----------------|----------|------------|--------------------|----------------|---------------|---------------|
| Corporate Owned | X | X | X | X | X | X |
| AD Group Labels | X | X | X | | | |
| Result | X | X | X | X | X | X |

This means that the device ownership restrictions will outweigh the user-based restrictions in the example presented in the CVD. Policy can also be structured based on this aggregation. Global policy will include the restrictions for all devices, such as those required for ISE compliance. Below this, a device will get restrictions based on ownership and additional restrictions based on the AD group. These profiles do not need to repeat the restrictions set as part of global policy and can focus on what restrictions are unique to the specific label in use. One fictitious example for a user in the BYOD_Partial_Access AD group using a corporate-owned device is shown in [Table 5](#).

Table 5 *Aggregation of Restrictive Policies*

| Profile | PIN lock | Encryption | Disable Cloud Sync | Disable Camera | Disable Music | Disable Games |
|---------------------|----------|------------|--------------------|----------------|---------------|---------------|
| Global | X | X | | | | |
| All Employee Owned | | | X | | | |
| BYOD_Partial_Access | | | | X | | |
| Result | X | X | X | X | | |

Combining restrictions is handled differently than combining payloads in other profiles. Two cases must be considered. The first case is profiles where multiple payloads of the same type are allowed. Consider a profile that provisions the VPN settings. The initial profile provisions the settings for the VPN connection named ABC and the second profile provisions the settings for the connection names XYZ. The device will end up with two named VPN connections. The second case is payloads with the same name. Extending the VPN example, if the second profile also contains the setting for the named connection ABC, then the device will ignore the second profile. MobileIron uses label priority to determine which settings will be used.

User Experience

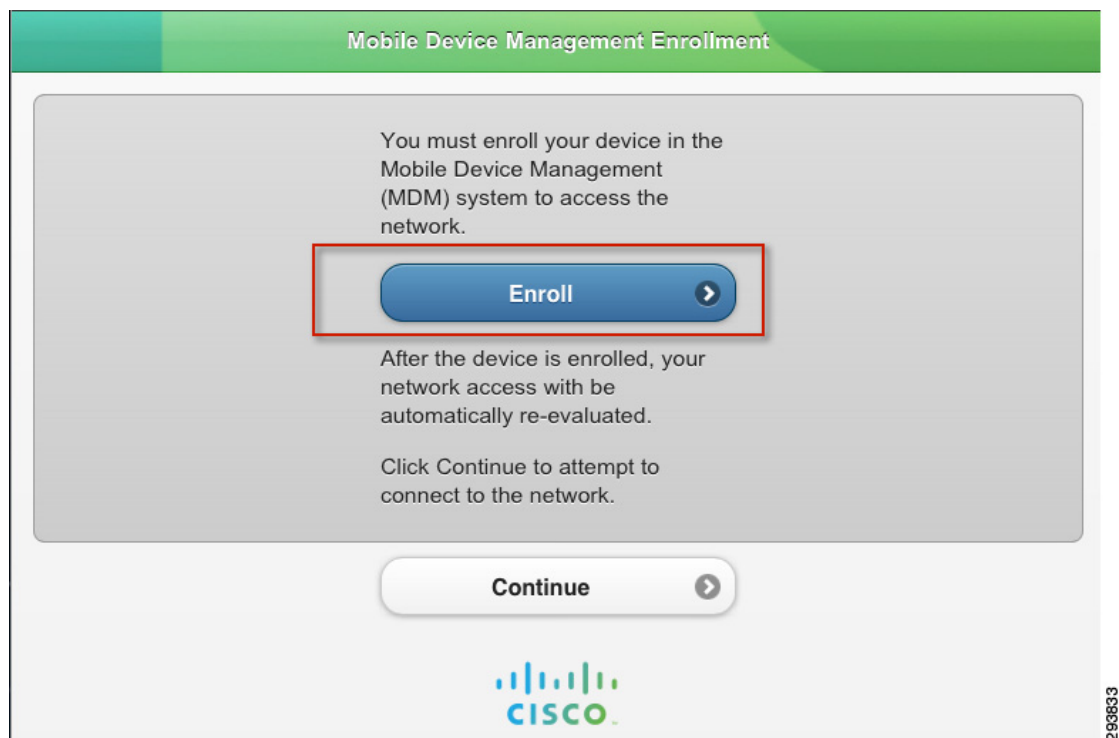
For the most part, the fact that a device is under management is seamless to the user. If they are running the mobile client application as recommended for ISE compliance checks, then the user will have some additional information about their device that will be useful for troubleshooting with ISE. Users will also be required to complete the onboarding procedure.

MDM On-boarding

The workflow that users must complete to onboard their device is set by the ISE policy. As presented in the CVD, the user will first on-board with ISE. When the user first joins the BYOD_Employee SSID, ISE will check the device's MDM Registration status through the MDM API. If the device is not registered, then a captive ACL is activated. This ACL will allow Internet access, but will capture any attempts to access corporate resources. A full explanation is provided in the CVD. The device requires Internet access to complete the MDM onboarding process including downloading the client application from either the Play Store or App Store.

When the device is captured, the user is redirected to the screen in [Figure 12](#), which includes two buttons. The Enroll button redirects the client to the MDM registration page. The Continue button issues a CoA to force a re-evaluation of the AuthZ policy after MDM enrollment completes.

Figure 12 **MDM Redirect For Enrollment**



After the device has enrolled, the server will request a check-in. During the initial check-in, additional profiles, applications, or Web Clips will be provisioned on the device.

Pass Code Complexity

The user may be required to configure a PIN lock on their device during the on-boarding process if the device is not already configured with one. When this occurs, the user will need to launch the client app and send data. This is explained in more detail in [Device Compliance/Restrictions](#). The MDM administrator can choose the minimum password length and complexity. The natural tendency is to require very strong passwords, however there may be unintended consequences. The PIN lock will need to be entered any time the employee wants to use their phone. While texting and driving is illegal in many locations, the PIN lock is also required to make phone calls.

If the user is required to navigate through several keyboards to enter the PIN lock, the administrator may be creating an environment of risk taking. There may be legal implications outside the scope of this document that should be considered. The more likely scenario is that the user will opt-out of the BYOD network for their personal devices. Devices not managed could have no PIN lock at all and yet still contain corporate data that the employee improperly put on the device. A practical approach is to require a simple 4- digit PIN on personal mobile phones. Corporate tablets can still be profiled with complex passcodes including special characters. This provides a balanced approach and will not discourage participation. Four-digit PINs or the last four digits of a SSN are used fairly often to provide some level of security.

Application Stores

The MobileIron MDM server can install public applications from either the App Store or Google Play or private applications that are uploaded to the catalog. Cisco AnyConnect can be provisioned directly in the VPN payload as shown in the last section of this document. This is a two payload profile that is deployed along with the application. One payload is the AnyConnect settings, account information, group, etc. The second payload is the user certificate to authenticate with the ASA. Public applications can be part of a Volume Purchasing Program (VPP) where the enterprise can purchase licenses in bulk.

Corporate Data

MobileIron and ISE can work closely together to create a fairly comprehensive approach to managing corporate data. Data comes in two forms, data at-rest and data in-flight. Data at-rest is stored directly the mobile device and data in-flight is the movement of data. This can be extended to include moving data between two storage containers on the same device.

Data at-Rest

Android and Apple handle stored data differently. Android has an open file structure that allows content to be shared between applications. This creates a tight and integrated environment. Many Android devices also support external and removable storage in the form of SD Cards. Apple iOS creates a storage environment for each application. When an application is deleted, the partition holding that application's data is also removed. MobileIron has implemented secure document folders as part of Docs@Work that can store corporate data in the Mobile@Work application's storage structure. The data is encrypted. Once there, it enforces controls on which applications can share the data. The Docs@Work is equipped with a reader for most common file types.

Data in-Flight

Sharing data between applications is fairly common. Built-in system applications like Contacts can share their information. With Apple devices, the data is passed through owning application. Apple iOS now provides privacy settings to control access to system data stores. The common thread with both Android and Apple is tight application integration. This functionality presents challenges when trying to contain data. Through Docs@Work, the MobileIron Mobile@Work application restricts data sharing.

Certainly moving corporate data to and from the device is also concern. The most common tool is email attachments, although cloud storage services such as Dropbox are also a concern. MobileIron can blacklist these types of applications. This is most appropriate on corporate devices. ISE can deploy per-user ACL through the Wireless LAN Controller to enforce this policy at the network level for both corporate and personal devices.

MobileIron Sentry offers an intelligent gateway to manage email attachments for Microsoft Exchange ActiveSync and offers several options for handling attachments. For example, the attachment could be removed from the email, encrypted, and placed in the MyDocs@Work folder.

Mobile Content management is evolving. Future releases of this document may explore this area in more depth.

Corporate Wipe

Both ISE and MobileIron can remove corporate data from personal devices. MobileIron calls this a Selective wipe. ISE refers to it as a Corporate Wipe. With iOS 6 and above, MobileIron uses the “retire” action to selectively wipe devices. Other common terms used are selective wipe or partial wipe. When ISE issues this command, it is forwarded to MobileIron via the API. The MDM will then remove corporate applications using privileges granted to the MDM Profile. When these complete, the MDM profile is removed, which will remove all the associated sub-profiles. While it is also possible to leave some applications behind, all MDM profiles will be removed. Profiles not installed by the MDM are not deleted. This includes the CA certificate the WiFi profile installed by ISE. When an application is deleted, the associated data is also removed. This is especially effective when Docs@Work has been deployed because it is a centralized location that holds sensitive corporate data. If an application was blacklisted by the MDM, it will be restored. The relationship between the MDM profile, sub-profiles and applications is important to understand.

Figure 13 *Profile and Sub-Profiles*

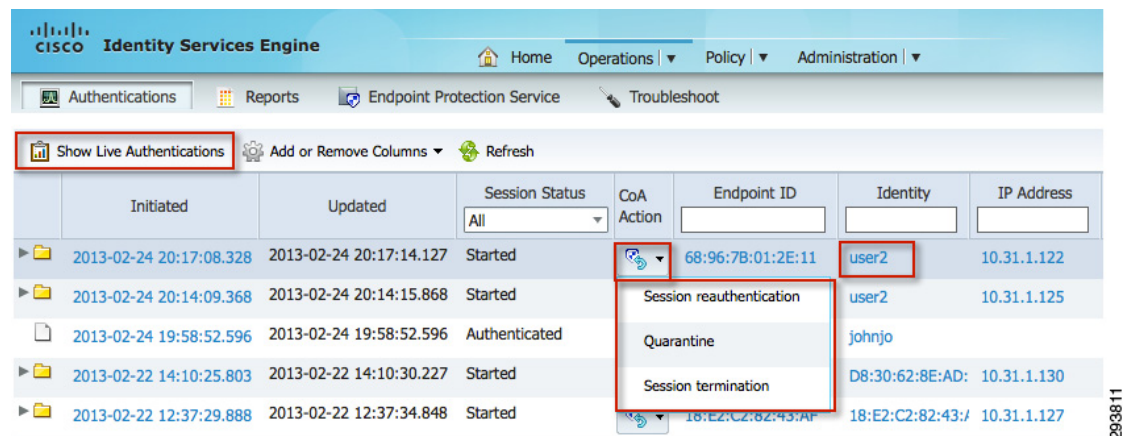


Figure 13 shows eight profiles and their relationships. The top two profiles are installed by ISE as part of the ISE enrollment process. Immediately below this a profile labeled “sdulab.com” and it is the first profile installed by the MobileIron MDM. This is the exclusive MDM profile that allows additional profiles called subprofiles to be installed or deleted transparently in the background via the Push service. The MDM profile also allows MobileIron to direct the user to install applications or Web Clips on the

device. Although the user is involved when applications are installed, they are not notified when an application is deleted. Also shown is the MobileIron Mobile@Work client application. This application can add and remove content to the users protected file folders.

The mobile client application can be removed. However, there may be cases where the administrator would like to leave the MobileIron client application on the device to allow an authenticated user to re-enroll the device. Corporate wipes by themselves do not blacklist the device from either the MDM or ISE. An ISE administrator, the MDM administrator, or the user from either the ISE MyDevices page or the MobileIron MyPhone@Work page may issue a selective wipe. If a corporate wipe is being issued as a result of an employee's termination, then additional steps must be undertaken, such as blacklisting the device with ISE and removing the user AD group memberships. This will prevent the user from re-enrolling the device. Optionally, the user certificate can be revoked on the CA server. The final action is to disassociate the user from the network, forcing them to re-authorize against ISE. The device may immediately try to re-associate, but will match the blacklist, thereby denying the device network access. The user will not be able to self-enroll this particular device until IT has removed the MAC address from the blacklist. [Figure 14](#) illustrates how a device can be removed from the network.

Figure 14 *Forced CoA from ISE*



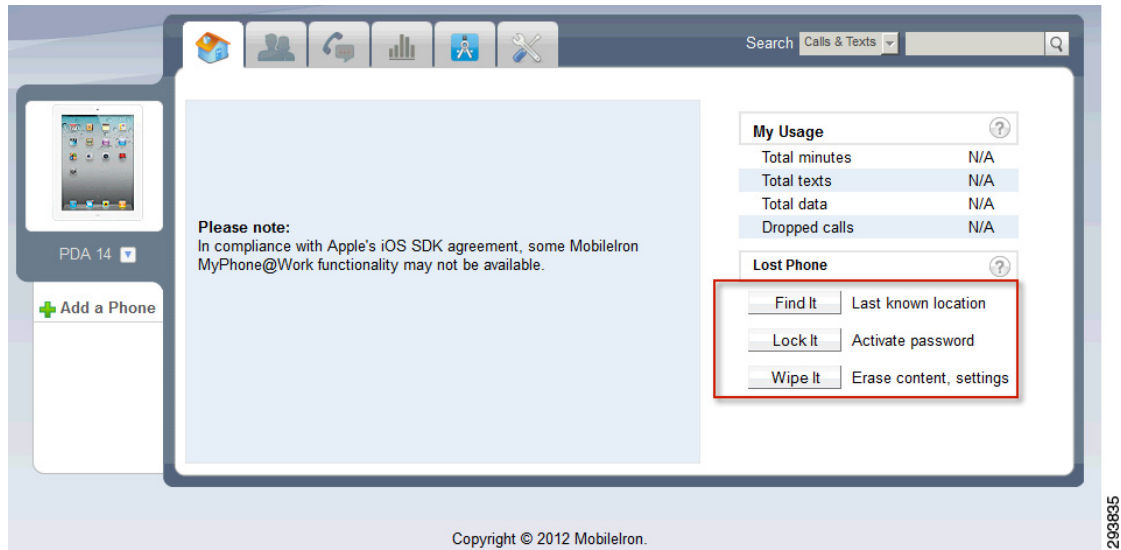
| | Initiated | Updated | Session Status | CoA Action | Endpoint ID | Identity | IP Address |
|--|-------------------------|-------------------------|----------------|--------------------------|-------------------|-------------------|-------------|
| | 2013-02-24 20:17:08.328 | 2013-02-24 20:17:14.127 | Started | | 68:96:7B:01:2E:11 | user2 | 10.31.1.122 |
| | 2013-02-24 20:14:09.368 | 2013-02-24 20:14:15.868 | Started | Session reauthentication | | user2 | 10.31.1.125 |
| | 2013-02-24 19:58:52.596 | 2013-02-24 19:58:52.596 | Authenticated | Quarantine | | johnjo | |
| | 2013-02-22 14:10:25.803 | 2013-02-22 14:10:30.227 | Started | Session termination | | D8:30:62:8E:AD | 10.31.1.130 |
| | 2013-02-22 12:37:29.888 | 2013-02-22 12:37:34.848 | Started | | 18:E2:C2:62:43:AF | 18:E2:C2:82:43:4F | 10.31.1.127 |

MyPhone@Work Portal

MobileIron offers a MyPhone@Work portal that allows the user to manage their devices. User Roles are defined to provide various level of functionality such as Find Device, Enterprise Wipe, or Device Wipe. These roles are typically applied to a user group. Web Clips are used to provide the user with a desktop shortcut.

ISE also provides a MyDevices portal as detailed in the CVD. Currently the two sites are distinct and not cross-linked. Some of the functionality does overlap such as the MDM actions. But users will likely want a Web Clip to both locations. [Figure 15](#) shows the home screen a user will see when they log into the MyPhone@Work Web portal and some of the lost device features available from that page.

Figure 15 *MyPhone@Work Portal*




Verify Device Compliance

ISE Compliance versus MDM Compliance

There are two compliance checks required of the device. The first is defined by policy configured on ISE. This is specific to network access control (NAC); the other is defined on the MDM and specific to mobile device policy (MDP). The use of an MDM to determine NAC is a fairly new concept, first supported in ISE 1.2. Mobile device compliance policy is an essential component of MDM and has context outside of network access. This is similar to NAC compliance prior to the integration of the MDM. Integrating the components together does not negate the need for two distinct compliance policies with meaning only within their respective context. The network administrator has to be careful not to confuse ISE compliance with MDM compliance with respect to NAC.

The attributes shown in [Table 6](#) should help clarify the difference between compliance policies.

Table 6 **Compliance Attributes**

| ISE Compliance Attributes | MobileIron Security Compliance Attributes |
|---|--|
| <input type="checkbox"/> DeviceCompliantStatus <input type="checkbox"/> DeviceRegisterStatus <input type="checkbox"/> DiskEncryptionStatus <input type="checkbox"/> IMEI <input type="checkbox"/> JailBrokenStatus <input type="checkbox"/> Manufacturer  <input type="checkbox"/> MDMServerReachable <input type="checkbox"/> Model <input type="checkbox"/> OsVersion <input type="checkbox"/> PhoneNumber <input type="checkbox"/> PinLockStatus <input type="checkbox"/> SerialNumber | <div> Connectivity - All Platforms <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Out-of-contact with Server for X number of days <input checked="" type="checkbox"/> Out-of-policy for X number of days </div> <div> Device Settings - All Platforms <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Passcode is not compliant </div> <div> App Control - All Platforms <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Disallowed app found <input checked="" type="checkbox"/> App found that is not in Allowed Apps list <input checked="" type="checkbox"/> Required app not found </div> <div> Data Protection/Encryption - iOS - Android <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Data Protection/Encryption is disabled </div> <div> iOS <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Disallowed iOS model found <input checked="" type="checkbox"/> Disallowed iOS version found <input checked="" type="checkbox"/> Compromised iOS device detected <input checked="" type="checkbox"/> iOS Configuration not compliant <input checked="" type="checkbox"/> Restored Device connected to server <input checked="" type="checkbox"/> MobileIron iOS App Multitasking disabled by user <input checked="" type="checkbox"/> Device MDM deactivated (iOS 5.0 or later) </div> <div> Android <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Disallowed Android OS version found <input checked="" type="checkbox"/> Compromised Android device detected <input checked="" type="checkbox"/> Device administrator not activated for DM client or agent </div> |

MobileIron uses Events to trigger a response to a non-compliant condition. There are several categories of events defined on the MDM:

- International Roaming
- Threshold Reached
- SIM Changed
- Memory Size Exceeded
- System Event
- Policy Violations Event.

Of these, ISE only considers the Policy Violations for MDM Compliance checks. Each event category has specific triggers that can generate a response from the MDM. The response typically includes a notification via SMS, Push, or Email. The notification could be sent to the user, administrator, or both.

Device Compliance/Restrictions

Restrictions and compliance are distinct but related concepts. A user is not offered the option of not adhering to a restriction. If a PIN lock is required, the device will be locked until the user selects a PIN that meets the established complexity. If the camera has been disabled, the icon is removed and the user has no way of launching the camera application. Restrictions are policy elements that are enforced without exception. Compliance is when a device is operating outside of the established policy. Non-restrictive items that could cause compliance events are things such as the minimum OS version. The key point is that it is not possible to be non-compliant with a restriction, with the exception of restrictions that include a grace period.

Device Scanning Intervals

The MDM client application can periodically scan the device. There are several different scans that run on different intervals. They are also available as device queries and are:

- **Device Information**—General information about the device includes serial numbers, UDID, phone number, operating system, model, battery status, etc.
- **Security**—Includes encryption status, device compromised, data roaming, SIM card status, and the number of profiles installed but not active.
- **Profiles**—The installed profiles on the device, including those not installed by MobileIron.
- **Apps**—A complete inventory of all the applications installed on the device.
- **Certificates**—A list of the installed certificates on the device.

Scan information is available in device details screen. When a device periodically checks in with the MDM server, it will notify the server of the current scan results.

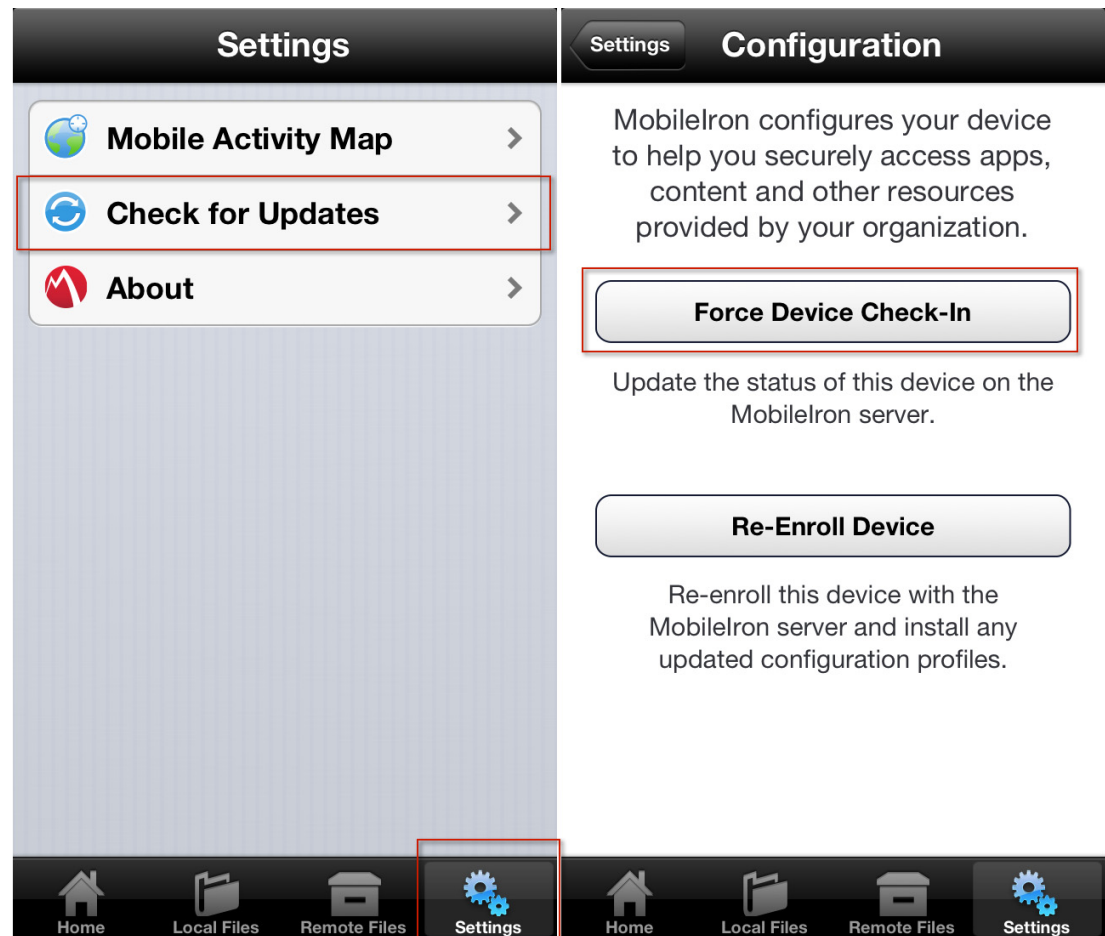
PINLockStatus

The PIN lock status is available to the API and can be used by ISE to set a minimum requirement for network access, as is shown in the CVD. Typically PIN lock is set as a restriction. But there are some cases where the MDM can set a compliance check against a restriction, specific to PIN lock. It is possible to set a PIN lock with a grace period. During this time, the MDM can poll the device for the PIN lock status. When set, the triggered action could be the installation of additional profiles. By doing this, the device could be on-boarded with the MDM but not granted full access until the user sets the password or the grace period expires.

There are some caveats to be aware of with respect to ISE creating a PIN lock requirement for network access. These are not specific to MobileIron, however the work around is. When users are issued instructions explaining the onboarding process, they should be asked to set a PIN lock on their device prior to starting the on-boarding process, rather than waiting for the forced PIN lock midway through the procedure. If the user does not follow this, they will likely end up in a quarantine state. There are two issues at play. First, the MDM server does not get a triggered update when a user creates a PIN lock. Because it is set as a restriction, the user is required to enter one, but it will be some time before polling interval before the server will become aware of the PIN lock. Second, the MDM on-boards by installing the MDM profile and certificate first. This secures the communications between the server and device. After this profile is issued, the server will send a check-in request to the device. Because the MDM payload is required to respond to check-in messages, this confirms the device is fully under management. On the initial check-in, the device is loaded with the remaining profiles, including the one containing the PIN lock restriction. Before this completes, the user will have hit the continue button on the MDM

redirect page, resulting in a CoA. This will re-authorize the device before the user has been prompted to enter a PIN lock and the user will end up being quarantined. The work around is to open the MobileIron client and wait for any current scans to compete. The device should show in compliance, providing PIN Lock has been included as a compliance condition in addition to a restriction. Once the client believes the device is compliant, the user should click the **Force Device Check-In** button shown in Figure 16 to update the server of the new posture. Then the user can try the continue button again or bounce their wireless to force a re-authorization.

Figure 16 **Update Device Status on MDM Server**



Jailbroken or Rooted devices

These are devices where the user has gained direct access to the operating system, bypassing the control imposed on the device by the service provider. Devices in this state are generally considered compromised and there has been some recent legislative action to prohibit users defeating locks imposed on the device by the providers. The BYOD CVD offers a policy that does not allow jailbroken or rooted devices on the network. This is based on the MDM API.

The MDM server will require a mobile client app installed on the device to determine the root status of the device. There are a few limitations to be aware of. Usually the process of rooting a device requires the user to reinstall the operating system. There is a good chance the user will uninstall the MobileIron

mobile client at the same time. Without the software, the server cannot determine if the device has been rooted, only that it has been compromised and is no longer under management. If the user also removes the MDM profile, then all of the child profiles are also removed with it, effectively resulting in a selective wipe. At this point, the user may attempt to onboard the device in a rooted or jailbroken state. The server will not be able to assess this condition until the mobile client is reinstalled on the device and has had a chance to complete a scan. There is a time delay between when a device is first compromised and when the MDM server will be first aware of a problem. There is no requirement in the MDM protocol that a device should contact the MDM when the MDM payload is removed. The server is left to poll for the condition periodically. This delay can carry forth into ISE policy because ISE can only respond to the attributes are they are returned by the MDM.

RegisterStatus

When a device is being on-boarded, ISE will check the RegisterStatus of the device with the MDM. If the device is not registered, then the user is redirected to the MobileIron enrollment page. Obtaining a status of registered with the MDM means that the device is known to the MDM and that an MDM payload, and the associated certificate, is on the device, and that the device has responded to at least one checkin request issued through APNS or GCM. A register status does not guarantee that all the profiles have been pushed to the device. Instead it indicates that the profile containing the MDM payload has been installed and that the device has responded to the initial check-in request. It is possible for profiles to be withheld until a posture assessment has been completed and reported back to the server. This could result in a registered device that is not equipped with the full set of intended restrictions.

Managing Lost/Stolen Devices

Corporate and personal devices require specific responses when lost or stolen. Personal devices reported as stolen should undergo an enterprise wipe to remove all corporate data. Lost devices may be handled in the same manner although the user may attempt to locate the device from the myDevices page first, but only if that service is allowed with the user's role privileges and location services are enabled on the mobile device. The user or Admin can also try to issue a "find device" if the either the mobile client app or Mobile@Work is installed on the device. If the device remains lost after an attempt to locate it, then an enterprise wipe is prudent. The device can be restored if later found by the user. The admin may also choose to blacklist the device on the network depending on the situation, forcing the user to call support to regain access.

Corporate devices have some more flexibility with respect to providing location information. If this information is available, then the administrator may have some options. They could choose to:

- Reassign the label membership to effectively remove all corporate applications and data, provision lock down profiles effectively rendering the device useless, yet leave the device under management such that forensic data is available in the event the enterprise would pursue legal options.
- Blacklist the device in ISE to prevent corporate access and issue an enterprise-wipe command to the device to remove all corporate data. This also removes the MDM profile. The device will become unmanaged, lifting all operational restrictions on the device including the ability to locate the device.
- Blacklist the device in ISE to prevent corporate access and issue a full wipe to the device to remove all information and return it to the factory default configuration. The carrier will need to be involved to prevent the now factory fresh device from having a resale value.

The exact response an enterprise would take in the event of a stolen device should not be public knowledge, especially when a full wipe is issued since the response could be an incentive to some criminals.




Application Distribution

Applications can be marked as required or optional. Required applications are usually automatically pushed to the device. Users can browse optional applications using the MobileIron App Catalog on their device. Applications can be from the public application store or developed in-house. Apple and Google both offer a volume purchasing program if paid applications are distributed. In-house applications can leverage MobileIron AppConnect technology to provide centralized configuration and additional security capabilities. Application management will be explored in future releases of this document. Readers are encouraged to refer to the AirWatch Administrator's guide for additional information.

Cisco Applications (Jabber, etc.)

Cisco offers a wide range of mobile business applications for both increased productive and security. [Table 7](#) shows some popular applications.

Table 7 *Popular Cisco Mobile Applications*

| | |
|---|---|
|  | AnyConnect—AnyConnect is a security application for improved VPN access, including on-demand domain-based split tunneling. |
|  | WebEx—WebEx is a productive application to allow mobile users to connect to online meetings. The application allows content sharing, video sharing, and VoIP or cellular audio. |
|  | Jabber—Jabber is a productivity application that integrates IP telephony, chat, and video conferencing using Cisco Call managers. |

MobileIron allows users to pre-provision the AnyConnect application using an application profile. Users can be prompted to enter their username and password or the profile can include a certificate payload that can be used to authenticate the users. The provisioning is found as part of a VPN profile, as shown in [Figure 17](#).

Figure 17 **AnyConnect Provisioning Profile**

Modify VPN Setting

Name: AnyConnect

Description: Secure Device Connectivity to the Corporate VPN Headend

Connection Type: Cisco AnyConnect ⓘ

Server: vpn.sdulab.com

User Name: \$USERID\$ ⓘ

Group: BYOD_USER

Login Group or Domain:

User Authentication: Certificate

Prompt for Password: ☐

Identity Certificate: System - IOS Enterprise

VPN on Demand: ☒

| Match Domain or Host ▲ | Connection Option |
|------------------------|-------------------|
| sdulab.com | Always |

Add New Delete

Proxy: None

203839

Conclusion

The integration of the network policy enforced by Cisco ISE and device policy offered by MobileIron's Mobile Device Manager offers a new paradigm for BYOD deployments where security and productivity are not competing objectives.

Disclaimer

The MobileIron configurations shown in this document should not be considered validated design guidance with respect to how the MobileIron MDM should be configured and deployed. They are provided as a working example that details how the case studies explored in the CVD can be carried forward to the MDM in an effort to provide a fully integrated and complementary policy across both platforms. This in turn will result in a comprehensive solution where the network and mobile devices are in pursuit of a common business objective. MobileIron is the only source for recommendations and best practices as it applies to their products and offerings.

