Oracle® Fusion Middleware

Administering Oracle Mobile Security Suite

11*g* Release 2 (11.1.2.3) for All Platforms

E55606-04

August 2016

Oracle Fusion Middleware Administering Oracle Mobile Security Suite, 11*g* Release 2 (11.1.2.3) for All Platforms

E55606-04

Primary Author: Kevin Kessler

Contributing Author: Priscilla Lee, Barbara Buerkle

Contributor:

Contents

4 Managing Devices and Workspaces

5 Managing Users and Mobile Roles

6 Managing Mobile Apps

7 Managing Device Configurations

8 Managing Mobile Security Policies

Part III Preparing Apps for use With Oracle Mobile Security Suite

9 Using the Oracle Mobile Security Suite Application Containerization Tool

10 Customizing the Oracle Secure Workspace App

Part IV Managing Oracle Mobile Security Suite Settings and Configuration

11 Configuring Mobile Security Manager

12 Configuring Your Environment to Work With Mobile Security Manager

13 Troubleshooting Oracle Mobile Security Suite

Preface

This guide describes how to configure and manage Oracle Mobile Security Suite.

Audience

This document is intended for Oracle Mobile Security Suite administrators. It focuses on how to configure, manage, and troubleshoot Oracle Mobile Security Suite using the Mobile Security Manager component. It also includes sections that describe how to prepare apps for use with Oracle Mobile Security Suite.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Fusion Middleware 11*g* Release 2 (11.1.2.3) documentation set::

- *Oracle Fusion Middleware Identity Management Release Notes*
- *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*
- *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*
- *Oracle Fusion Middleware High Availability Guide for Oracle Identity and Access Management*
- *Oracle Fusion Middleware Administering Mobile Security Access Server*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*
- *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New in This Release?

This chapter lists product enhancements included in Oracle Mobile Security Suite 11.1.2.3.0.

Product Enhancements for Oracle Mobile Security Suite 11.1.2.3.0

The following enhancements are included in this release:

- New servers built on Oracle Fusion Middleware – The server components of the Oracle Mobile Security Suite (Mobile Security Manager and Mobile Security Access Server) have been rebuilt on top of the Oracle Fusion Middleware technology stack for this release. Oracle Mobile Security Suite can now be installed, configured, and managed using mechanisms that are consistent with other Oracle Identity and Access Management products.
- Support for additional LDAP directories – Oracle Mobile Security Suite now supports the following LDAP directories:
  - Microsoft Active Directory 2008, 2008R2, and 2012R2
  - Oracle Unified Directory 11gR2 (11.1.2.2+)
  - Oracle Internet Directory 11gR1 (11.1.1.7 and 11.1.1.9)
  - Oracle Directory Server Enterprise Edition (ODSEE) 11gR
- Support for additional Oracle Database versions – Oracle Mobile Security Suite now supports the following Oracle Database versions:
  - Oracle 11.1.0.7+
  - Oracle 11.2.0.1+
  - Oracle 12.1.0.1+
- Mobile Device Management – Provides the ability to secure corporate-owned mobile devices. Enforces device policies and restrictions that conform to corporate security policies and provides remote controls to manage mobile devices. Compliance with policies can be tracked and remediation actions enforced. Android MDM functionality relies on Google Cloud Messaging (GCM); iOS MDM functionality relies on Apple Push Notification Service (APNS). Features include:
  - Device Configurations – Use Device Configurations to create pre-configured E-mail, VPN, calendar, and Wi-Fi settings profiles that can be added to mobile security policies.
  - Device Restrictions – For Android: camera only. For iOS: camera, app install, assistant (Siri), cloud backup, cloud doc sync, cloud Keychain sync, diagnostic submission, explicit content, fingerprint unlock, lock screen control center,

lock screen notifications view, lock screen today view, ad tracking, iTunes, iTunes Store password entry, untrusted TLS prompt, Shared Stream, screenshot, Safari, Photo Stream, Passbook while locked, over-the-air PKI updates.

  - Device passcode – Passcode policy restrictions, including minimum length, history, idle timeout, failed attempts, expiry, expiry duration, and password complexity.
  - Android Device Encryption – Enables device encryption for Android devices.

- Oracle Access Manager console integration – The Oracle Mobile Security Suite and Oracle Access Manager UI consoles have been combined in this release into a single unified Policy Manager console. This provides centralized administrative, helpdesk, and self-service UI functionality from both products in a single place. A unified console improves the user experience and reduces management costs.

- Risk-based step-up authentication – Oracle Mobile Security Suite can now use the capabilities of Oracle Access Manager to perform context aware risk-based step-up authentication at the time the user registers or logs in to the Secure Workspace app. Step-up authentication is an additional authentication factor on top of the primary password, and can take the form of either Knowledge-Based Authentication (KBA) or a One Time Password (OTP). This feature is available when the Secure Workspace app is configured to use OAuth2 Mobile Client authentication. Information on configuring Oracle Access Manager with adaptive access is available in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*. Refer to "Understanding OAuth Services Plug-ins" in the "Understanding OAuth Services" chapter.

- Mobile Security Access Server as an OAM 11g WebGate – The Mobile Security Access Server can optionally be enabled as an Oracle Access Manager 11g WebGate. This allows the Mobile Security Access Server to protect access to otherwise unprotected web applications, and provide single sign-on to them relying on the standard Oracle Access Manager login page, tokens, and HTTP redirects.

- Containerized app recovery – If the Secure Workspace app has been deleted and reinstalled, automatic recovery of the encrypted data underlying containerized mobile apps is now supported. End-users can now reinstall the Secure Workspace app after mistakenly—or intentionally—deleting it without any loss of service or data by the set of containerized apps that are part of the Workspace.

- Support for Android 5.0 – This release of Oracle Mobile Security Suite now fully supports the Secure Workspace app and App Containerization on Android 5.0 devices.

- Containerization of Oracle Mobile Application Framework (MAF) apps – Oracle Mobile Security Suite 11g Release 2 (11.1.2.3.0) has a tight integration with the Oracle Mobile Application Framework (MAF) 2.1.3 or higher to support containerization of MAF apps across both iOS and Android. This integration enables the Oracle Mobile Security Suite functionality for secure networking, security storage, and data leakage prevention within MAF apps at both the virtualized and native levels.

- Localized User Interfaces – All Oracle Mobile Security Suite user interfaces, including both the MSM console UI, as well as the mobile UI exposed by the Secure Workspace app and the app containerization functionality, have been translated and can now be displayed in the standard sets of Oracle Fusion Middleware localized languages.

- Accessibility Compliance – This release includes significant investment across all components of the Oracle Mobile Security Suite in the area of accessibility. More information on the Oracle Accessibility Program is available at the following website:

  `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`

- Kiosk Mode (Workspace Launcher) – For Android devices only. Minimizes interaction with the operating system outside of the Workspace and prevents the user from closing the Secure Workspace app, making this mode suitable for public environments where supervision is minimal, such as lobbies, exhibit spaces, and show rooms. Addresses retail, healthcare, manufacturing, and other use cases where a single device is used by multiple users to access corporate data.
- Basic authentication for Secure E-mail and Mobile File Manager – Support for ActiveSync-enabled e-mail servers and file share servers that are protected with Basic authentication, in addition to the Windows authentication supported in previous releases. This enables Secure E-mail and Mobile File Manager features to be used in environments where the Secure Workspace app is configured for OAM/OAuth password authentication instead of Windows authentication.
- Shared Workspace Mode – Allows multiple users to share a single installed Secure Workspace app on a shared device. This feature addresses retail, healthcare, manufacturing, and other use cases where multiple users need to access online resources in an authenticated and secure fashion on a shared device, but do not require data to be locally stored between authenticated sessions. Locally stored data is securely wiped when each authenticated session ends or when the user logs out of the session.
- Android Kiosk Mode – The ability to configure the Secure Workspace app as an Android launcher is now supported. This makes the Secure Workspace app homepage the home screen of the Android device and enables Kiosk scenarios where administrators only want to expose certain limited functionality of the device to end users. Minimizes interaction with the operating system outside of the Secure Workspace and prevents the user from closing the Secure Workspace app, making this mode suitable for public environments where supervision is minimal, such as lobbies, exhibit spaces, and show rooms.
- Custom Web URLs for password management prior to Secure Workspace login – Ability to customize the Secure Workspace app with a set of web URLs accessible to the end-user prior to login. When selected, these web URLs will be opened in an embedded Web view within the Secure Workspace app, and can be used to expose password change, password reset, forgot userid, and other functionality that needs to be used prior to login. These web URLs can point to the password management functionality exposed by Oracle Access Manager, or another system.
- Secure storage using Android NDK containerization – A new native-level mechanism for securing the storage of containerized apps on Android devices. This native-level secure storage is complementary to the Java-level secure storage present in previous releases, and together they enable securing a wider range of locally stored data.
- Role-based mobile access authorization – Oracle Mobile Security Suite can now check for mobile-only access authorization. Use this feature to allow or deny users access to specific web apps and web services based on their role assignments. For example, you can deny users access to an internal resource if they have the Contractor role.

Part I

Introducing Oracle Mobile Security Suite

This part contains conceptual documentation that introduces Oracle Mobile Security Suite (OMSS).

This part contains the following chapters:

1

Understanding Oracle Mobile Security Suite

This chapter provides conceptual information about Oracle Mobile Security Suite. It includes the following sections:

- Understanding Oracle Mobile Security Suite
- Key Components
- Key Concepts
- Deployment Topologies
- Understanding the Oracle Mobile Security Suite Process Flows

1.1 Understanding Oracle Mobile Security Suite

Oracle Mobile Security Suite is an Enterprise Mobility Management (EMM) solution offering that is focused on simplifying and securing enterprise mobility by leveraging existing investments on Identity and Access Management platforms. It provides:

- Interfaces to manage and secure devices and apps.
- A Secure Workspace to secure enterprise apps and data. Productivity apps as part of OMSS include a browser, a file manager, and a corporate directory app. Available separately are a PIM app with e-mail, calendar, contacts, notes, and tasks functionality, and a doc viewer/editor app.
- An app containerization tool that provides secure access to third-party apps added to the Secure Workspace.
- An access server that brokers app authentication and establishes an app tunnel to allow access to corporate resources behind the firewall without the use of a device-level VPN.

The Secure Workspace isolates and protects corporate apps and data from other apps and data on the device, and policies control the movement of data into and out of the Workspace. Oracle Mobile Security Suite provides users with:

- Seamless access to intranet resources and corporate data using enterprise-grade security.
- Deep integration with Oracle Access Manager and directory server integration with Oracle Unified Directory server (OUD), Oracle Internet Directory (OID), Oracle Directory Server Enterprise Edition (ODSEE), and Microsoft Active Directory authentication for true single sign-on.

How IT departments deploy Oracle Mobile Security Suite typically depends on whether IT needs to manage the device, or just the corporate data on the device:

- If IT *does not* need to manage the device, users download the Secure Workspace app, which provides access to enterprise-approved resources that are appropriate to the user's role. In this deployment, the user administers the device and can access company resources only from within the Secure Workspace app that IT manages. This type of deployment is called a *Mobile Application Management* (MAM) deployment. It is typically used for BYOD (bring-your-own-device) deployments in which the user is deploying the client on a personally owned device.
- If IT needs to manage both the device and the data, the OMSS client provides device management capabilities such as the option to blacklist apps, disable device functions (such as the camera), or wipe compromised devices. The client provides the same comprehensive control over the Secure Workspace as found in the MAM deployment. This is a Mobile Device Management + Mobile Application Management (MDM+MAM) deployment. It is typically used for COPE (corporate-owned personally-enabled) deployments in which IT owns the device, but it can also be deployed to personally-owned devices if the user agrees to allow IT to manage their device.

MAM-only deployments can coexist with MDM solutions from other vendors. For customers who need to manage COPE devices, but do not have an MDM solution, Oracle Mobile Security Suite can address their device and application management requirements.

Oracle Mobile Security Suite is deployed as an Oracle Access Management component. Optionally, Oracle Mobile Security Suite, Oracle Access Management, and Oracle Identity Manager can be deployed together. For details, see Section 1.4, "Deployment Topologies."

1.2 Key Components

Oracle Mobile Security Suite consists of the following key components: the Secure Workspace app, the Mobile Security Access Server (MSAS) component, the Mobile Security Manager (MSM) component, the Policy Manager Server, the Mobile Security App Containerization Tool, and the Secure Mobile Mail Manager app for iOS/Android.

Secure Workspace

End-users install the *Secure Workspace* app on their iOS and Android devices. The Secure Workspace app acts as the security provider for all apps running within its realm. Any containerized app installed on the device makes use of the Secure Workspace app to take care of security functions such as authentication, single sign-on, data encryption, and a wide variety of access and data leakage prevention policy enforcement. The Secure Workspace app also comes with a set of embedded apps, like the Secure Browser for any corporate Web access, Secure File Manager for corporate data collaboration, and an option to view and install the list of apps that are allowed for that user. The Secure Workspace isolates and protects corporate apps and data from other apps and data on the device, and the movement of data into and out of the Workspace is controlled by policies.

Mobile Security Access Server

Mobile Security Access Server is a gateway component that typically sits in a DMZ to provide an AppTunnel connection from containerized apps. Some of the functions that the Mobile Security Access Server provides include the following:

- Acts as a mutually authenticated SSL tunnel from each Workspace app to provide secure access to any containerized apps.
- Serves as the authentication broker between devices and authentication servers such as Oracle Access Manager and Active Directory.

The Mobile Security Access Server can sit in front of other Oracle gateway servers (OAG) if needed and brokers authentication and proxy requests to their destinations. Multiple Mobile Security Access Server instances can be installed and active simultaneously. You can control these instances from a single browser-based administrative console. The Mobile Security Access Server is a Java SE server that runs the Grizzly NIO framework (`https://grizzly.java.net/`).

Mobile Security Manager

Mobile Security Manager is installed with Oracle Access Management on a WebLogic application server in the green zone. Some of the functions that the Mobile Security Manager provides include the following:

- Enrolls devices and Secure Workspaces
- Allows administrators and end-users to manage their devices/Workspaces remotely (for example, by locking or wiping a lost or stolen device)
- Allows administrators to define and enforce device and application level policies in conjunction with the Secure Workspace (for example, device/Workspace lifecycle management, mobile application management, mobile security policy administration, and so on)
- Allows administrators to manage the Mobile Application Catalog
- Hosts the file manager service

Unified Administrative Console

System Administrators manage the Mobile Security Manager and Mobile Security Access Server components using the Oracle Access Management console, which is hosted on the *Policy Manager Server*. Administrators open the console in a desktop web browser. The OAM console contains multiple LaunchPad pages, each of which contains a grid of boxes or *tiles* that can open additional console pages if selected. The *Mobile Security Manager console* and the *Mobile Security Access Server console* are two such consoles for System Administrators. The Policy Manager Server also hosts the *Help Desk console* for Help Desk administrators, and the *Self-Service console* for end-users in environments without Oracle Identity Manager. If Oracle Identity Manager is available, end-users instead use the OIM self-service console to make Mobile Security Manager requests, while System Administrators can use both the OIM console and the OAM console to manage Mobile Security Manager. (System Administrators can only use the OAM console to manage the Mobile Security Access Server.)

Mobile Security App Containerization Tool

The *Mobile Security App Containerization Tool* "containerizes" mobile iOS and Android apps by:

- Injecting the apps with Oracle app security services
- Signing the apps with the customer's enterprise distribution certificate, and
- Adding the apps to the Secure Workspace

The Containerization Tool is a MacOS application that features a command-line interface.

Secure Mobile Mail Manager

The *Secure Mobile Mail Manager* is an iOS and Android mobile personal information manager (PIM) app for secure mail, calendar, contacts, task, and notes apps. It syncs with corporate mail servers using ActiveSync. Note that this app is licensed separately and may not be included in your environment.

Secure White Pages App

The *Secure White Pages App* is an iOS and Android mobile corporate directory app that interfaces with existing LDAP directories and can be containerized and deployed as part of the Secure Workspace. The White Pages app only supports OAM-based authentication and is only applicable for OAM deployments.

1.3 Key Concepts

This section introduces Oracle Mobile Security Suite concepts.

Devices

Oracle Mobile Security Suite supports devices that run the iOS or Android mobile operating system. Forked Android devices like Kindle Fire are supported. Refer to the Oracle Mobile Security Suite certification matrix for updated information about supported devices.

Secure Workspace

The Secure Workspace is the security container deployed to the mobile device that provides secure access to your employer’s IT network. (The Secure Workspace is an app that runs on the device.) The Secure Workspace provides superior app-level encryption and a superior app-level VPN to protect enterprise data and networks. It also includes a set of trusted containerized apps that are isolated from the user’s personal apps on the mobile device.

MAM and MDM

In Oracle Mobile Security Suite, users enroll each mobile device as either a MAM-only device or a MDM+MAM device.

- A user with a MAM-only (Mobile Application Management) device retains control over the device and has total freedom with their personal apps and data. For this reason MAM-only devices are called *unmanaged devices*.
- A user with a MDM+MAM (Mobile Device Management+Mobile Application Management) device *does not* administer the device. For example, an MDM+MAM device can be remotely wiped by an administrator, and policies can enforce device restrictions, such as preventing the camera from being used. For this reason MDM+MAM devices are called *managed devices*.

On both managed and unmanaged devices, users access company resources from within the Secure Workspace.

Users and Roles

Users and roles are not managed by Oracle Mobile Security Suite but are managed using your existing directory or identity management tools. Users are assigned to one or more roles/groups on the directory server. User and role definitions are then referenced in real-time (without any sync from LDAP to the OMSS database). Administrators can use roles to send groups of users notifications to enroll a mobile device in Oracle Mobile Security Suite. System administrators can also use role assignments to manage devices and workspaces. For example, System Administrators can lock, unlock, and wipe devices and workspaces by role assignment.

Mobile Security Policies

System Administrators use security policies to provision apps and network access to mobile users, and to limit mobile users' actions and access to maintain security. Using the Mobile Security Manager console, administrators assign one or more policies to a role. If conflicting policies apply to a role, Mobile Security Manager enforces whichever policy rules are the most restrictive. All policies are managed on the server, but enforced on the client. Therefore policies are enforced even when the client is offline. Policies are defined using the Mobile Security Policies page, which consists of six configuration tabs: a tab for defining general policy info, a tab for assigning the policy to roles, a tab for configuring device policy settings, and three tabs for configuring Workspace policy settings.

Device Policy

A policy that either restricts device features or specifies device authentication details is a device policy. Device policies are only enforced on managed devices. When defining a policy, all MDM and device-specific settings are located together on the **Device** tab, so a device policy is any policy that has the **Device** tab configured. A device policy can disable almost two dozen iOS features, such as access to the Safari browser, or device access to the Apple app store. Android's device management capabilities are limited to disabling the camera. For a list of device restrictions, see "Restrictions" in the *Help Reference for Oracle Mobile Security Suite Consoles*.

Workspace Policy

Workspace policies affect the operations of the Secure Workspace and are enforced on both managed and unmanaged devices. System Administrators use Workspace policies to specify Workspace authentication details, allow (or restrict) Workspace privileges, enable and configure application settings, and set Time Access and Geo Access settings. Policies that specify enrollment requirements, compliance rules, and app assignments are also Workspace policies. In terms of the Mobile Security Policies page, a workspace policy is a policy that has any tab *other than* the **Device** tab configured.

Compliance Policies

Following enrollment, compliance policies verify that a device is policy compliant. Compliance policies can look for blacklisted apps, check for extended periods of inactivity, detect if a device is jailbroken, and enforce passcode compliance. Compliance policies also continue to check that enrollment requirements are being met. If a device is flagged as non-compliant, the policy can lock the Workspace, wipe the device/Workspace, or take no action, depending on configuration. Compliance polices are enforced on both managed and unmanaged devices. On unmanaged devices, the compliance policy can wipe the Workspace, but it cannot wipe the device.

App Catalog

The Secure Workspace includes a corporate app catalog that is managed by system administrators. The App Catalog can include native iOS or Android apps and virtual apps. Native apps are installed to the device at the OS level and can be containerized or uncontainerized. On an unmanaged device, a user can only install an app from the Mobile Application Catalog if the policies assigned to the user's role allow for access to the app. On a managed device, administrators can additionally blacklist specific apps. This feature checks for device compliance and, depending on how the policy is enforced, the system locks or wipes the device if blacklisted apps are found.

Device Configuration

Policies can define which apps and device configuration settings are available for users belonging to a specific role or group. Note: Device configuration settings (Wi-Fi, VPN, e-mail, calendar) are available for iOS devices only.

De-Register or Wipe the Device

To protect the device, administrators can de-register or wipe the device. *De-registering* the device removes the provisioning profiles installed on the device along with the Secure Workspace app while retaining the remaining personal data. *Wiping* the device restores the device to the factory-installed state, wiping the entire device, its contents, and any custom configuration settings.

Lock or Wipe the Workspace

To protect data in the Secure Workspace, administrators can lock or wipe the Workspace. *Locking* the Workspace logs the user out of any existing sessions and does not allow re-authentication or access to Workspace apps, but all data remains and the Workspace can be unlocked and accessed again. *Wiping* the Workspace removes all data and configuration settings for Workspace apps and returns them to their original (pre-Workspace) system state without touching any personal data located outside of the Workspace.

Encryption

Data in transit and data stored locally inside containerized apps on the mobile device is encrypted. Encrypted data storage includes cache, file store, user preference, and database data. OMSS encryption is certified FIPS140-2 Level 1 encryption.

OMSS employs a sophisticated key hierarchy to protect data. All keys are derived from user credentials that are never stored. The key hierarchy involves multiple keys to support different sensitivity of data. As an example, a unique key is used for the user's authentication certificate, which is allowed to be open for a very short period of time. A different key is used for the browser cache, which must remain decrypted for an entire session. The main Oracle Secure Container distributes and manages keys for the complete set of apps in the user's Secure Workspace.

App Containerization

The App Containerization Tool injects code around third-party native apps that provide enhanced security services for authentication and networking, secure storage, and policy enforcement. Containerization does not require any code changes. Instead, the tool injects code into standard framework calls. To be eligible for containerization, apps must use the HTTP/HTTPS protocol. Apps that use lower-level networking are not supported. The same enterprise certificate must be used to sign the Oracle Secure Container and all containerized apps.

Data Sharing and Data Leakage Protection (DLP)

Oracle Mobile Security Suite provides eleven data leakage protection settings for Workspace apps, including Restrict Copy/Paste, Allow/Deny E-mail, Allow/Deny Instant Messaging, Restrict File Sharing, and others. Administrators can define policies that restrict sharing to Workspace apps only, or allow data to leak out to non-Workspace apps.

Security Certificate

Part of the trust model for the Secure Workspace is having all Workspace apps signed by the same enterprise certificate. You must use the same certificate to sign the Oracle Secure Container and all containerized apps. Signing the apps makes them available for enterprise distribution in the enterprise app store. Public app stores do not allow containerized apps that perform OS-level introspection (for example, encryption and network introspection).

When the enterprise certificate expires, the Secure Workspace app will no longer launch. Distribution certificates are valid for three years from the date of issue, or until the Enterprise Developer Program expires, whichever comes first. You can have two distribution certificates active at the same time, each independent from the other. The second certificate provides an overlapping period during which you can update your apps before the first certificate expires. Users will need to upgrade or reinstall the app (without uninstalling the old app) before the certificate expires. System Administrators can use the console to upload a new version of the app and alert users to upgrade.

Workspace Apps

The Workspace apps that are bundled with OMSS include a secure browser, a file manager for iOS (an Android file manager is available from a third-party vendor), a corporate white pages app, and an App Catalog. These apps can be enabled or disabled by policy as needed. An e-mail (PIM) app is also available as an add-on.

The Mobile File Manager app uses WebDAV to communicate directly to any WebDAV compliant file share. The Mobile File Manager Server converts WebDAV to CIFS to communicate with CIFS shares like Microsoft file shares.

The Oracle White Pages app is a corporate directory app that connects with an existing LDAP store. It is only compatible with OAM SSO.

Finally, you can add your own apps to the Workspace. This includes custom and third-party device-native apps, and "virtual apps"—either a web app that displays in a web browser, or a shared folder app that connects to a network file share.

PIM Apps

A PIM (personal information manager) client named Oracle Secure Mobile Mail Manager is available as an add-on to the suite. The app is an OEM product from Nitrodesk/Symantec that uses the Microsoft ActiveSync mechanism to sync e-mail. Any ActiveSync mail server is supported, including Microsoft Exchange, IBM Lotus Domino, and Zimbra. In addition to mail, the app provides calendar, contacts, task, and notes apps.

AppTunnel

Allows access to corporate resources behind the firewall without the use of a device-level VPN. AppTunnel provides seamless integration with Active Directory and Oracle Access Manager for authentication, zero code-wrapping technology, extensive policies, and a complex key hierarchy that eliminates the need to cache the user password in the device. The AppTunnel is a mutually authenticated SSL tunnel from each Workspace app that provides secure access to any containerized app. The AppTunnel encrypts all data in transit and provides protection from rogue apps on a user’s mobile device that device-level mobile VPNs are subject to. Data in transit in the AppTunnel in encrypted with OpenSSL FIPS.

Authentication and Single Sign-on Support

Oracle Access Manager and Microsoft Active Directory are supported for authentication. For single sign-on, OMSS supports Kerberos and NTLM in Microsoft environments, and OAM basic, OAuth, and SAML in Oracle Access Management environments. If using OAM, OAM can be configured to understand Kerberos token, effectively providing seamless SSO between the two authentication protocols. For more information on this configuration, refer to "Configuring Access Manager for

Windows Native Authentication" in the *Administrator's Guide for Oracle Access Management*.

1.4 Deployment Topologies

There are two OMSS deployment topologies: OMSS + OAM Server, and OMSS + OAM Server + OIM.

Oracle Access Management 11.1.2.3 server is required to run Oracle Mobile Security Suite because OMSS utilizes the OAM server console. If you use Active Directory for authentication and single sign-on, you do not need to enable OAM authentication and single sign-on. Otherwise, OAM authentication is required if using any of the other supported directory servers.

If Oracle Access Management 11.1.2.2 is already deployed in your environment, Oracle Mobile Security Suite can use that version for authentication and SSO, provided that OMSS is deployed on Oracle Access Management 11.1.2.3 on a separate WLS domain. Note that if you use Oracle Internet Directory, this topology requires at least Oracle Internet Directory 11.1.1.7. Older versions of Oracle Internet Directory will need to updated to work with Oracle Mobile Security Suite.

Oracle Mobile Security Suite (including the OAM server) can also be integrated with Oracle Identity Manager 11.1.2.3. Customers can use the OIM Admin Console to manage mobile devices, policies and apps. Self-service users can use the OIM self-service console to manage mobile devices and Workspaces, in addition to managing their user profiles and accounts.

1.5 Understanding the Oracle Mobile Security Suite Process Flows

This section briefly covers Oracle Mobile Security Suite system architecture and how data flows between system components and the user's mobile device.

One or more Mobile Security Access Servers are located on the far side of a firewall in a DMZ (Figure 1–1). On the other side of the firewall, Mobile Security Manager and Oracle Access Management are deployed together in a WebLogic domain in the Green Zone. OAM and OMSS use the same LDAP store for users and groups. For system data, OAM and OMSS maintain separate data stores on a database server.

The Mobile Application Management (MAM) Data Flow

This diagram illustrates the data flow if Oracle Access Management provides authentication. Data flows between components as follows:

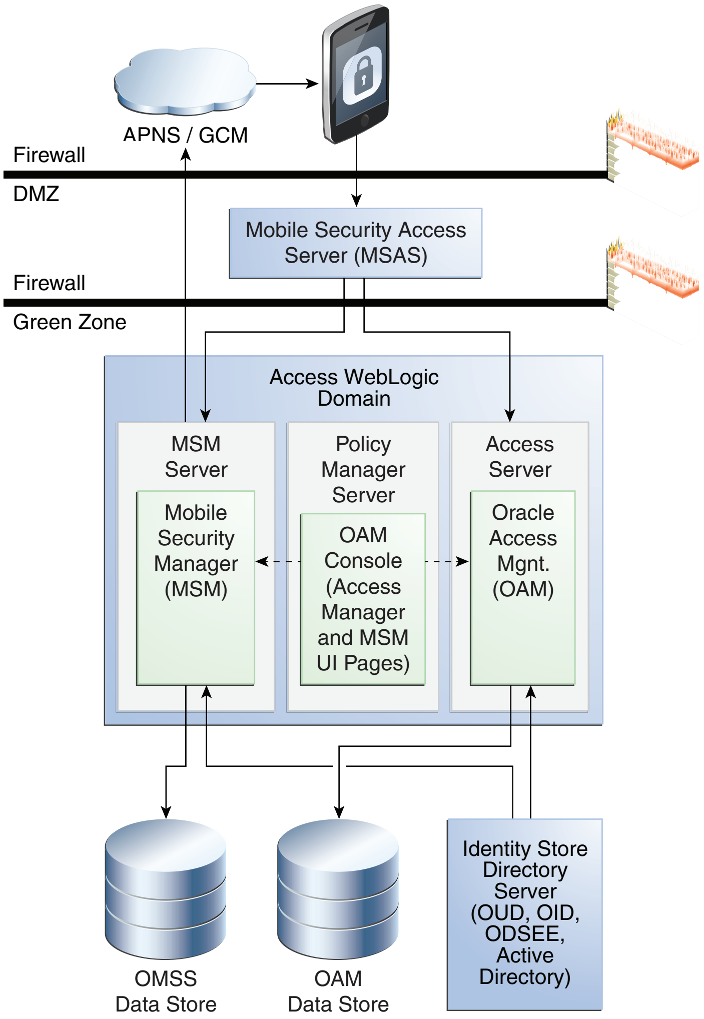
- Users log in to the Secure Workspace app installed on the mobile device. The Secure Workspace first connects to the Mobile Security Access Server gateway component, which forwards the connection past the firewall to Access Manager for single sign-on authentication and authorization. SSO support is provided across Access Manager 11g and 10g WebGate-protected resources. Mobile Security Manager polices enforce each users' mobile access privileges.
- The Oracle Access Management console, which includes the Mobile Security Manager console, the Help Desk console, and the Self-Service console, is hosted on the Policy Manager Server.
- Oracle Access Management and Mobile Security Manager utilize the same identity store.
- In the case of an MDM-enrolled device, Mobile Security Manager sends notifications, such as "device-policy update notifications," to the mobile device

using either Apple Push Notification Service (APNS) for iOS devices, or Google Cloud Messaging (GCM) for Android devices. When the device receives its notification, it downloads the updated policy from the server.

Note: For details about APNS and GCM, refer to the following Web pages:

- Apple Push Notification Service
- Google Cloud Messaging

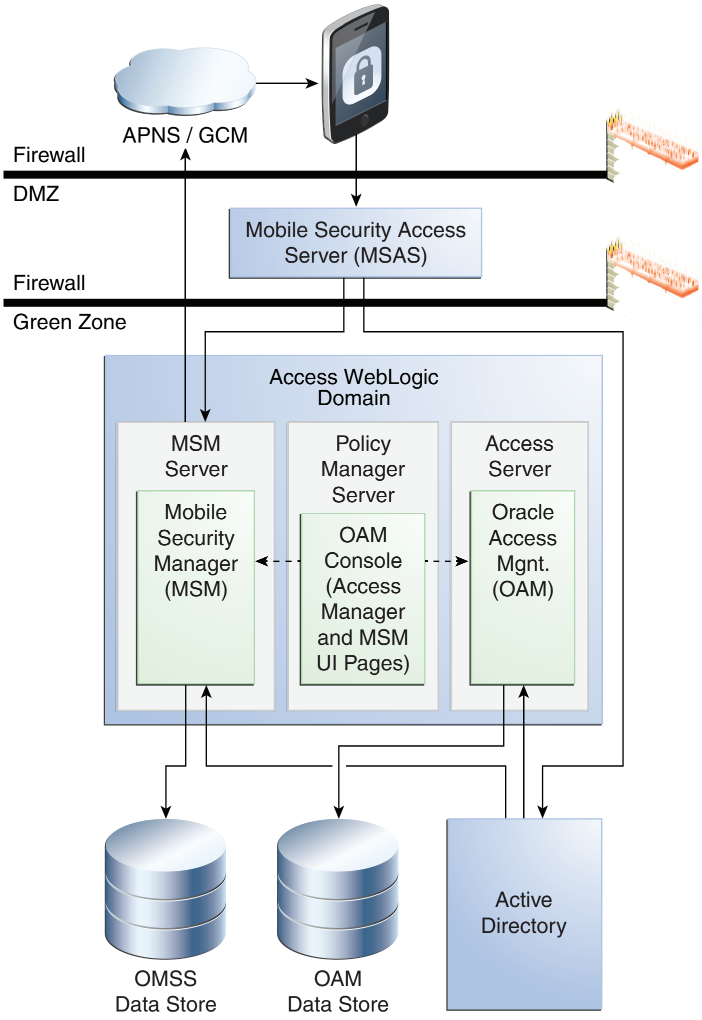
Figure 1–1 Logical diagram showing Oracle Mobile Security Suite required components


APNS / GCM
Firewall
DMZ
Mobile Security Access Server (MSAS)
Firewall
Green Zone
Access WebLogic Domain
MSM Server
Mobile Security Manager (MSM)
Policy Manager Server
OAM Console (Access Manager and MSM UI Pages)
Access Server
Oracle Access Mgnt. (OAM)
OMSS Data Store
OAM Data Store
Identity Store Directory Server (OUD, OID, ODSEE, Active Directory)

Note: If Oracle Access Management 11.1.2.2 is used for SSO and authentication, install Oracle Access Management 11.1.2.3 with Oracle Mobile Security Suite in a separate domain.

If Active Directory is the authentication provider, the Mobile Security Access Server (MSAS) interacts directly with Active Directory. The KINIT/PKINIT registration uses this flow for Workspace (MAM) registration, and the Active Directory Authenticator uses this flow for managed device (MDM) registration.

Figure 1–2 Logical diagram showing data flow if Active Directory is the identity store



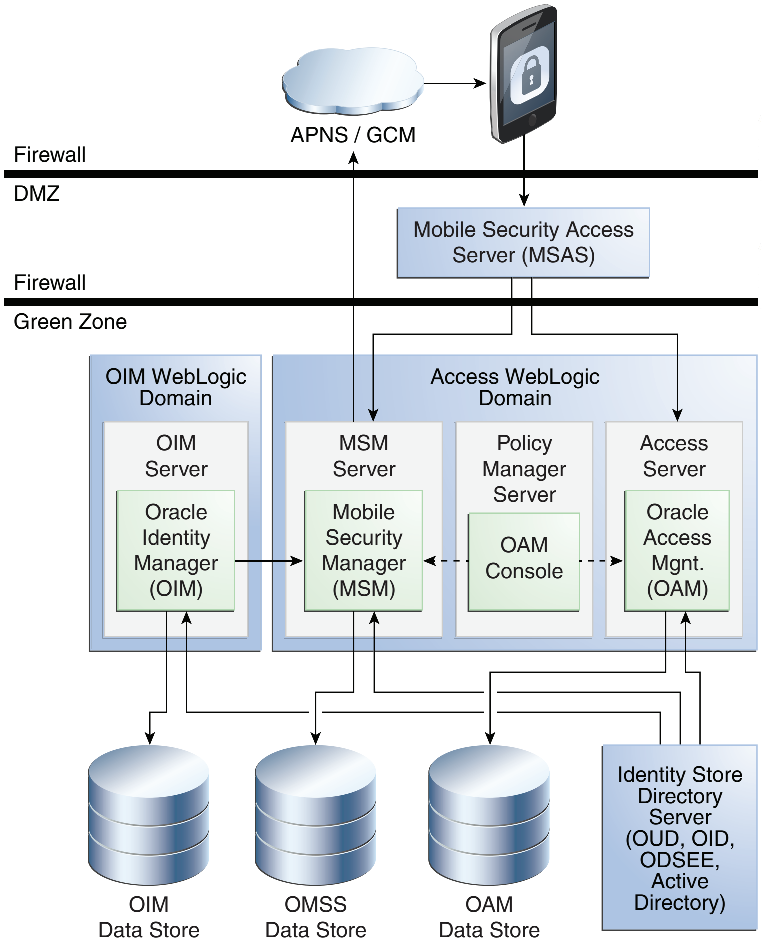
If Deployed With Oracle Identity Manager

In this topology (Figure 1–3), the directory server provides LDAP updates to the Access Manager, Mobile Security Manager, and Oracle Identity Manager components. Users can manage their Oracle Mobile Security Suite accounts from Oracle Identity Manager. Oracle Identity Manager persists its system data in the OIM data store.

> **Note:** For details about integrating Oracle Identity Manager and Oracle Mobile Security Suite, see "Integrating Oracle Mobile Security Suite and Oracle Identity Manager."

If Oracle Identity Manager is integrated with Oracle Mobile Security Suite, install Oracle Identity Manager in a WebLogic domain separate from the Oracle Access Management WebLogic domain.

Figure 1–3 Logical diagram showing Oracle Identity Manager integrated with Oracle Mobile Security Suite



Part II

Working With Mobile Security Manager

This part contains documentation that describes how to use the Mobile Security Manager to manage Oracle Mobile Security Suite.

This part contains the following chapters:

2

Getting Started Working With Mobile Security Manager

This section contains topics that will help you start using Mobile Security Manager. It is organized into the following sections:

- Administrator Roles
- Working With the Mobile Security Manager Console Pages

2.1 Administrator Roles

There are two administrator groups in Mobile Security Manager: the System Administrator group, which has full administrative privileges, and the Help Desk Administrator group, which has limited privileges. System Administrators are tasked with advanced operations, such as configuring the system, defining policies, and managing mobile roles; Help Desk Administrators are tasked with routine operations, such as inviting users to enroll a device in the mobility program, resetting passwords and passcodes, and unlocking locked Workspaces. Table 2–1 lists the different privileges that are granted to System Administrators and Help Desk Administrators.

Table 2–1 Comparison of System Administrator, Help Desk Administrator, and End-User privileges in Mobile Security Manager

Privileges	System Administrator	Help Desk Administrator	End User
Device Privileges			
Search for and view mobile devices	Yes	Yes	Yes (Own device only.)
Lock, Wipe, De-register, Sync, and Reset/Clear Passcode on mobile devices	Yes	Yes	Yes (Own device only.)
Workspace Privileges			
Search for and view Workspaces	Yes	Yes	Yes (Own Workspace only.)
Lock, Unlock, Wipe, and Reset Passcode on Workspaces	Yes	Yes	Yes (Own Workspace only; cannot Unlock.)
Mobile Users Privileges			
Search for users and view basic user information in the connected Identity Store	Yes	Yes	No
Invite mobile users to enroll a device in the mobility program	Yes	Yes	No
Mobile Roles Privileges			
Search for and view roles in the connected Identity Store	Yes	Yes	No

Table 2–1 (Cont.) Comparison of System Administrator, Help Desk Administrator, and End-User privileges in Mobile Security Manager

Privileges	System Administrator	Help Desk Administrator	End User
View policies assigned to roles	Yes	Yes	No
Assign policies to (or remove policies from) roles	Yes	No	No
Invite users by role assignment to enroll a device in Oracle Mobile Security Suite	Yes	No	No
Lock, unlock, and wipe devices and Workspaces by role assignment	Yes	No	No
Mobile App Catalog Privileges			
Search for and view apps in the Mobile App Catalog	Yes	Yes	No
Add, edit, or delete apps in the Mobile App Catalog	Yes	No	No
Mobile Device Configurations Privileges			
Search for and view e-mail, VPN, calendar, and/or Wi-Fi device configurations	Yes	Yes	No
Add, edit, or delete e-mail, VPN, calendar, and/or Wi-Fi device configurations	Yes	No	No
Mobile Security Policies Privileges			
Search for and view Mobile Security Policies	Yes	Yes	No
Create, edit, and delete Mobile Security Policies	Yes	No	No
Other Administrative Privileges			
View Mobile Security Manager settings	Yes	No	No
Change Mobile Security Manager settings	Yes	No	No
Access the end-user self-service console	Yes	Yes	Yes

2.1.1 How to Add Mobile Security Manager System Administrators and Help Desk Administrators

You can configure admin groups during or after installation. To configure the System Administrator and Help Desk Administrator groups during installation, specify the LDAP groups that should map to the `OMSS_IDSTORE_ROLE_SECURITY_ADMIN` and `OMSS_IDSTORE_ROLE_SECURITY_HELPDESK` roles respectively.

To configure admin groups after installation, open the **Identity Store Settings** tab (to learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page") and update the LDAP group name(s) in the **System Admin Groups** table and the **Helpdesk Groups** table.

> **Note:** Administrator role changes might take up to 10 minutes to take effect.

2.2 Working With the Mobile Security Manager Console Pages

This section includes the following topics:

- About the Mobile Security Manager Console Pages
- Opening the Mobile Security Manager Console Pages

- Accessing Online Help

2.2.1 About the Mobile Security Manager Console Pages

System Administrators, Help Desk Administrators, and end-users each have their own management consoles that they use to interact with Mobile Security Manager. All three consoles should be viewed in a Web browser.

Note: When entering information into the management consoles, do not use the < (less-than sign) or > (greater-than sign) except to define content in notification templates. The < and > characters are restricted.

System Administrators log in to the Mobile Security Manager console, and Help Desk administrators log in to the Help Desk console. The Help Desk console provides a limited interface that only contains the functionality needed for the Help Desk admin role.

Note: See "Administrator Roles" for detailed information about how the two admin roles differ.

Both the Mobile Security Manager console and the Help Desk console are deployed on the Oracle Access Management console. If Oracle Mobile Security Suite is integrated with Oracle Identity Manager, the console pages are also integrated with the Oracle Identity Manager console, and you can manage Mobile Security Manager from either console.

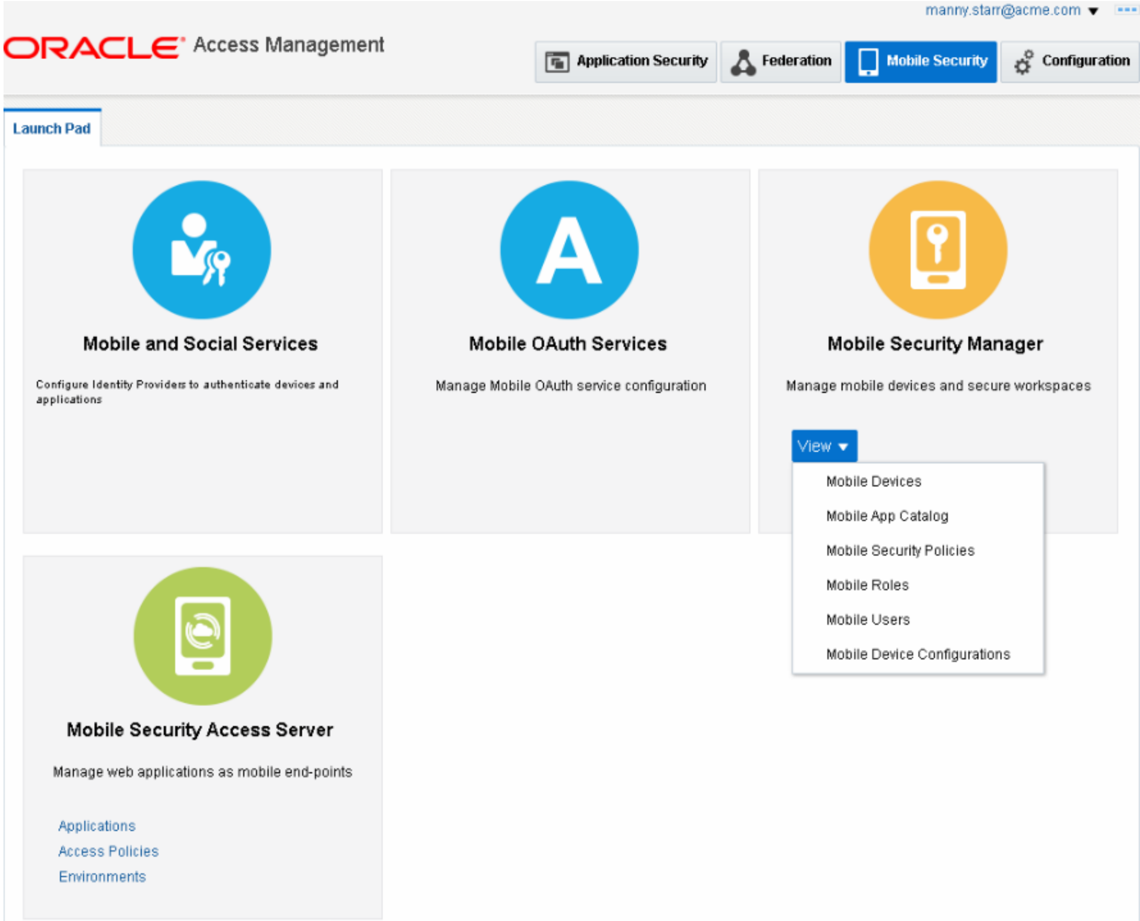
The Mobile Security Manager Console

The Mobile Security Manager console consists of six pages:

- **Mobile Devices** - View the devices and Workspaces registered by a user and take security actions against a device or Workspace (lock, un-lock, wipe, and so on).
- **Mobile App Catalog** - Add and remove apps in the catalog and edit app details.
- **Mobile Security Policies** - Create, edit, and remove mobile security policies, and associate roles with policies.
- **Mobile Roles** - Invite users by role assignment to register a device in Oracle Mobile Security Suite; lock, unlock, and wipe devices and Workspaces by role assignment; and assign policies to a role (or remove policies from a role).
- **Mobile Users** - View basic user information and invite a user to register a device/Workspace with Oracle Mobile Security Suite.
- **Mobile Device Configurations** - Add a new e-mail, VPN, calendar, or Wi-Fi configuration, or edit or remove an existing configuration.

The Mobile Security Manager Settings page is located in the **Configuration** section of the Oracle Access Management console.

Figure 2–1 The Mobile Security Manager console shown in the Oracle Access Management console



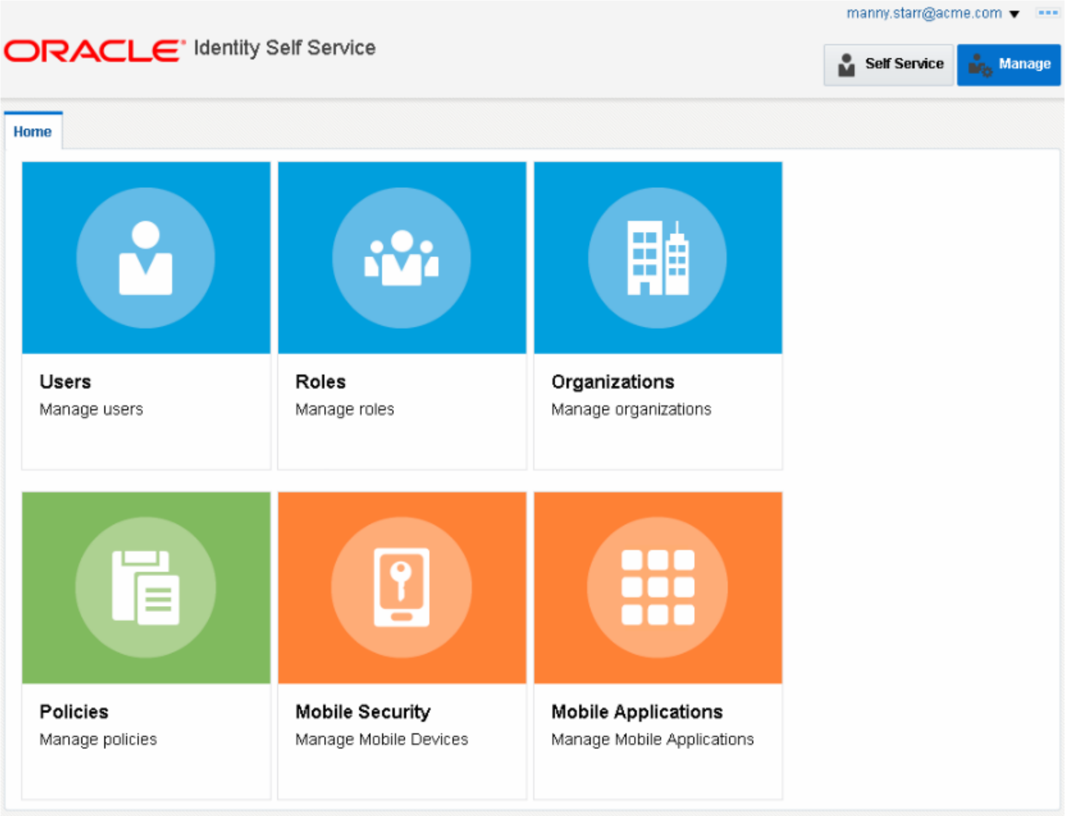
Configure Identity Providers to authenticate devices and applications

Manage Mobile OAuth service configuration

Manage mobile devices and secure workspaces

Manage web applications as mobile end-points

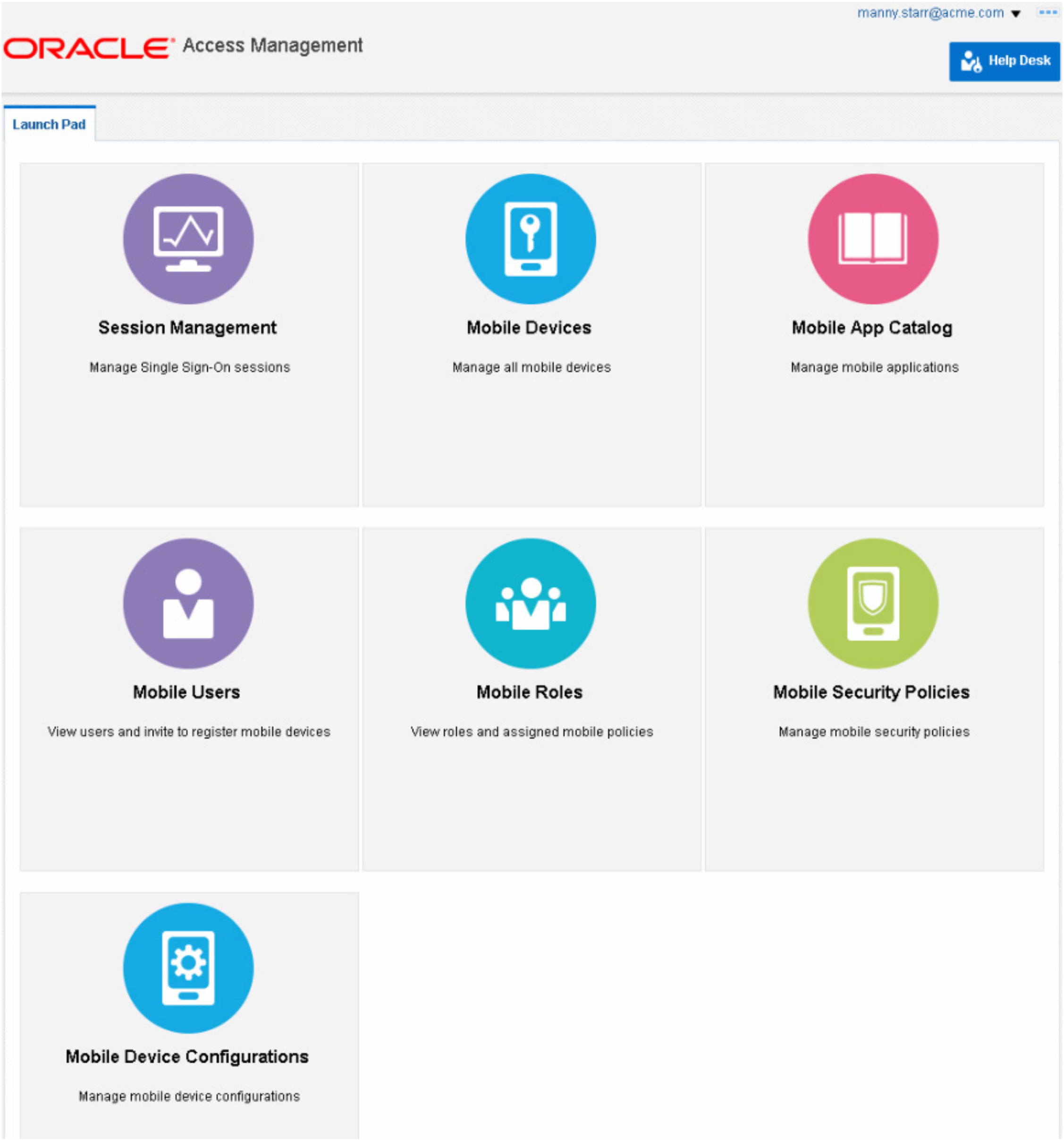
Figure 2–2 The Mobile Security Manager console pages as shown in the Oracle Identity Manager console



The Help Desk Console

The Help Desk Console is comprised of the six Mobile Security Manager console pages and a Session Management admin page for Access Manager. The Help Desk Console does not include the Mobile Security Manager Settings page.

Figure 2–3 The Help Desk console



ORACLE Access Management
manny.starr@acme.com
Help Desk
Launch Pad
Session Management
Manage Single Sign-On sessions
Mobile Devices
Manage all mobile devices
Mobile App Catalog
Manage mobile applications
Mobile Users
View users and invite to register mobile devices
Mobile Roles
View roles and assigned mobile policies
Mobile Security Policies
Manage mobile security policies
Mobile Device Configurations
Manage mobile device configurations

The Mobile Security Manager Self-Service Console

Oracle Mobile Security Suite features a Self-Service Console that end-users can use to:

- Register devices with Oracle Mobile Security Suite
- View their device and workspace details
- Perform self-service management operations, such as lock, wipe, de-register, reset passcode, and so on

System Administrators and Help Desk Administrators can also log in to the Self-Service Console to manage their devices, provided that they are registered with Oracle Mobile Security Suite as end-users.

Figure 2–4 The Self-Service Console page as shown in the Oracle Access Management console

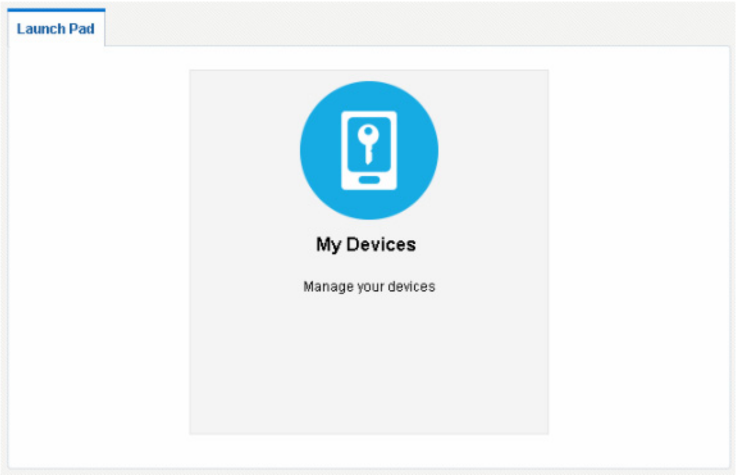
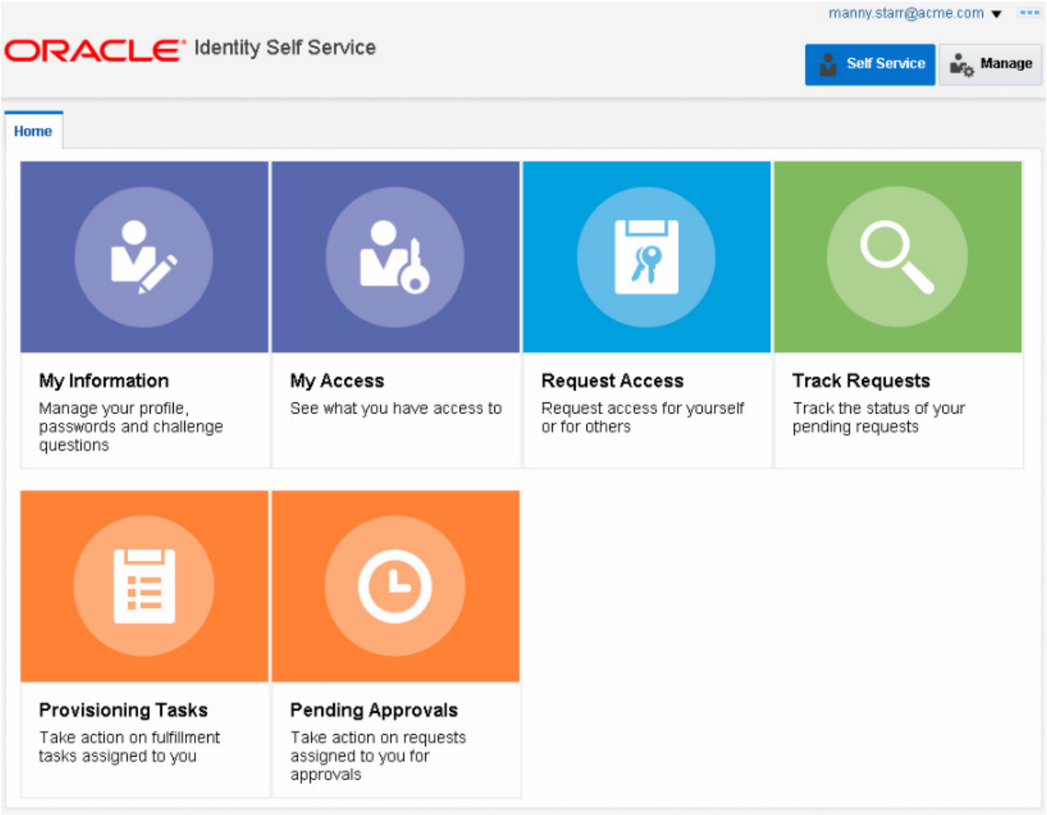


Figure 2–5 The Self-Service console as shown in the Oracle Identity Manager console


manny.starr@acme.com
ORACLE Identity Self Service
Self Service
Manage
Home
My Information
Manage your profile, passwords and challenge questions
My Access
See what you have access to
Request Access
Request access for yourself or for others
Track Requests
Track the status of your pending requests
Provisioning Tasks
Take action on fulfillment tasks assigned to you
Pending Approvals
Take action on requests assigned to you for approvals

2.2.2 Opening the Mobile Security Manager Console Pages

This section includes the following topics:

- Opening the Mobile Security Manager Console and Help Desk Console
- Opening the Mobile Security Manager Console in Oracle Identity Manager

Opening the Mobile Security Manager Console and Help Desk Console

Use these steps to open the Mobile Security Manager console pages in the Oracle Access Management console. If you are a Help Desk administrator, the Help Desk console opens instead.

1. In a browser window, open the Oracle Access Management console using the appropriate protocol (HTTP or HTTPS). For example:

   ```
   https://hostname:policy-manager-port/access
   ```

   or:

   ```
   https://oam.example.com:14150/access
   ```

   For details, see "Working with the Oracle Access Management Console" in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.

2. Log in with your user name and password.

   Depending on your role either the Mobile Security Manager console or the Help Desk console opens.

3. Choose from the following options:

   - If logged in as a Help Desk Administrator, the Help Desk console opens. Click the tiles in the Launch Pad grid to open the Mobile Security Manager pages.
   - If logged in as a System Administrator, click **Mobile Security**.

     The Mobile Security Launch Pad opens.

     Under **Mobile Security Manager**, click **View** and choose from the Mobile Security Manager console pages in the menu.

Opening the Mobile Security Manager Console in Oracle Identity Manager

Use these steps to open the Mobile Security Manager console pages in the Oracle Identity Manager console.

1. Open the Oracle Identity Manager console in a browser using the appropriate protocol (HTTP or HTTPS).

   ```
   https://oim-server-host:oim-server-port/identity
   ```

   or:

   ```
   https://oim.example.com:14000/identity
   ```

2. Log in with your user name and password.

3. Click **Manage** in the top right corner.

   *The Manage **Home** page opens.*

4. The Mobile Security Manager console pages are integrated with Oracle Identity Manager as follows:

   - Click **Policies** and choose **Mobile Security Policies** from the menu.
   - Click **Mobile Security** and choose either **Devices** or **Device Configurations** from the menu.
   - Click **Users** or **Roles**. Mobile Security Manager tabs are built into the Users page and Roles page.
   - Click **Mobile Applications** on the Oracle Identity Manager console Home page.

2.2.3 Accessing Online Help

At any time while using the consoles, you can click the **Help** link located in the drop-down menu at the top right part of the page under the user name. The system opens a Help page that describe the console page being viewed. Mobile Security Manager field-level Help descriptions are also documented in the *Help Reference for Oracle Mobile Security Suite Consoles.*

For general information about using Help, see "Accessing Online Help" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.

3 Enrolling Devices and Workspaces

This chapter documents the process of enrolling devices and Workspaces. It is organized into the following sections:

3.1 About Device and Workspace Enrollment

Users enroll mobile devices and workspaces with Oracle Mobile Security Suite to access corporate apps and resources (for example, shared folders). A user's eligibility to enroll a device/Workspace is determined by Mobile Security Policies. Each Mobile Security Policy has one or more roles assigned to it, and users that belong to these roles (either directly or indirectly) can enroll a devices/Workspace in Oracle Mobile Security Suite.

Upon enrolling the device, the policies under which the user enrolled the device will be enforced on the device and/or Workspace. For more information, see Section 4.4, "Managing Devices and Workspaces With Policies."

Enrolling a device/Workspace is a two step process:

1. Invite - The system sends an invite e-mail to the user that contains links and (optionally) a One Time Password to be used for enrollment.
2. Enrollment - The user follows the instructions and links in the invite e-mail and enrolls their device/Workspace.

> **Note:** Before performing either a device or Workspace enrollment, the mobile device must trust the SSL certificate used by the OMSS server. If the SSL server certificate is issued by a *publicly-trusted certificate authority* that the mobile device already trusts, then no specific action is required. If the SSL server certificate is issued by an *enterprise certificate authority*, or if the certificate is self-signed, then the user must install and trust the certificate authority root certificate on the mobile device before starting enrollment. System Administrators can provide the certificate authority root certificate to users by including an installation link in the invite e-mail, or by a separate out-of-band process.

3.2 Preparing and Sending the Invite Template

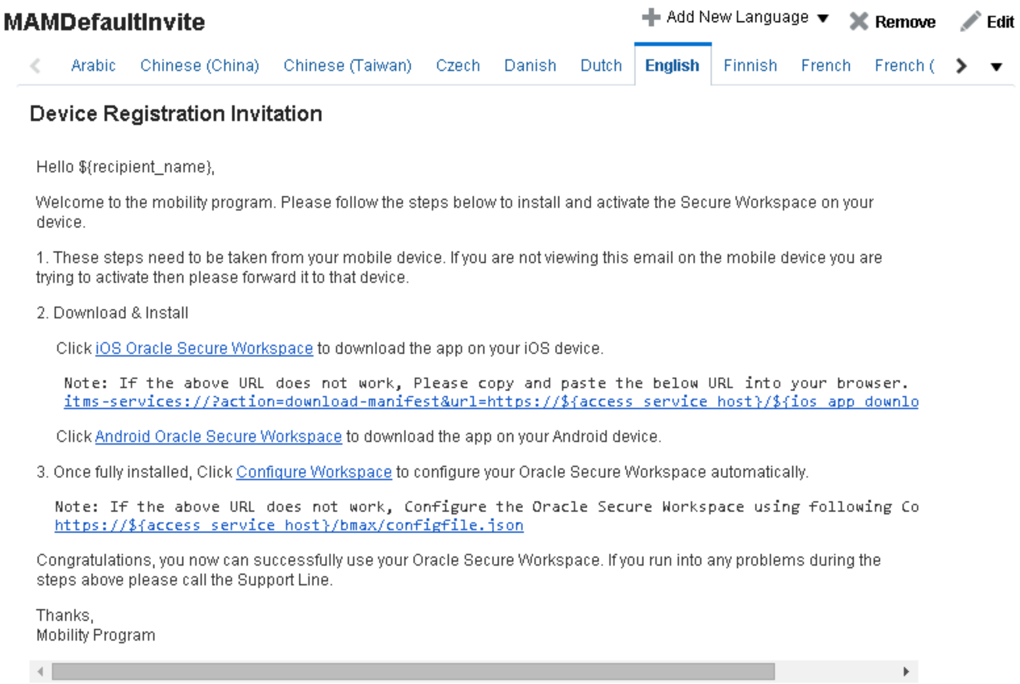
This section includes the following topics:

- Configuring the Invite Template
- Attaching Invite Templates to a Policy
- Inviting the User
- How to Invite a Group of Users by Role Assignment

3.2.1 Configuring the Invite Template

System Administrators can configure enrollment invite templates using the Mobile Security Manager console. Mobile Security Manager ships with a default template for each of the two types of enrollment: MAM-only enrollment and MDM+MAM enrollment.

Figure 3–1 Default MAM-only invite template



Device Registration Invitation

Hello ${recipient_name},

Welcome to the mobility program. Please follow the steps below to install and activate the Secure Workspace on your device.

1. These steps need to be taken from your mobile device. If you are not viewing this email on the mobile device you are trying to activate then please forward it to that device.

2. Download & Install

Click iOS Oracle Secure Workspace to download the app on your iOS device.

```
Note: If the above URL does not work, Please copy and paste the below URL into your browser.
itms-services://?action=download-manifest&url=https://${access_service_host}/${ios_app_downlo
```

Click Android Oracle Secure Workspace to download the app on your Android device.

3. Once fully installed, Click Configure Workspace to configure your Oracle Secure Workspace automatically.

```
Note: If the above URL does not work, Configure the Oracle Secure Workspace using following Co
https://${access_service_host}/bmax/configfile.json
```

Congratulations, you now can successfully use your Oracle Secure Workspace. If you run into any problems during the steps above please call the Support Line.

Thanks,
Mobility Program

The invite e-mail contains a link with configuration information that tells the Workspace app what servers to communicate with and what types of authentication to display to the user. The user can either click this link in the invitation e-mail, or manually enter the text in the first screen of the Workspace app when it is opened for the first time.

> **Note:** In Figure 3–1, the template contains a sample link that should be modified to match the actual URL. The sample link is:
>
> `https://${access_service_host}/bmax/configfile.json`
>
> Note that `${access_service_host}` is a special placeholder for use with invite templates. When the system sends a notification to a user, it replaces the placeholder with data configured in the system. In this case, the ${access_service_host} placeholder is replaced with the MSAS Runtime Server Base URL.

The Mobile Security Access Server hosts multiple configuration links, and it is up to the administrator to ensure that the correct link for the environment is included in the invite e-mail sent to users.

Table 3–1 Purpose of JSON configurations links for use in invites.

JSON Link Name	Description	Example URL
`bmconfig_oam_pwd.json`	Instructs the workspace to use OAuth Confidential Client authentication.	`https://`*`msas.host:port`*`/bmax/bmconfig_oam_pwd.json`
`bmconfig_oauth2_oam.json`	Instructs the workspace to use OAuth Mobile Client authentication.	`https://`*`msas.host:port`*`/bmax/bmconfig_oauth2_oam.json`
`bmconfig_kinit_kinit.json`	Instructs the workspace to use Kerberos password authentication.	`https://`*`msas.host:port`*`/bmax/bmconfig_kinit_kinit.json`
`bmconfig_pkinit_tlp.json`	Instructs the workspace to use Kerberos PKI-based authentication, with initial registration using a time-limited passcode.	`https://`*`msas.host:port`*`/bmax/bmconfig_pkinit_tlp.json`

3.2.1.1 How to Create and Edit Notification Templates

Follow these steps to create or edit notification templates that the system uses to send e-mail invitations to users. Notification templates can be created and saved in multiple languages.

1. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."
2. Click **Notification Templates** on the menu bar. (If Notification Templates is not visible, use the arrow buttons to scroll the menu bar to the right. Or, click ▼ to view additional menu items.)
3. Choose from the following:
   - To create a new template, click **Create Template**.
   - To edit a template, click the template name to open it, then click **Edit**.

   Note that invitations require separate installer links depending on if the user's device is an iOS or Android device, and if the user is enrolling in the MAM-only program or the MDM+MAM program. Use placeholder values to configure these and other variables in the notification template. For descriptions of all of the placeholder values that you can include in the template, click Help or see the "Notification Templates" section in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

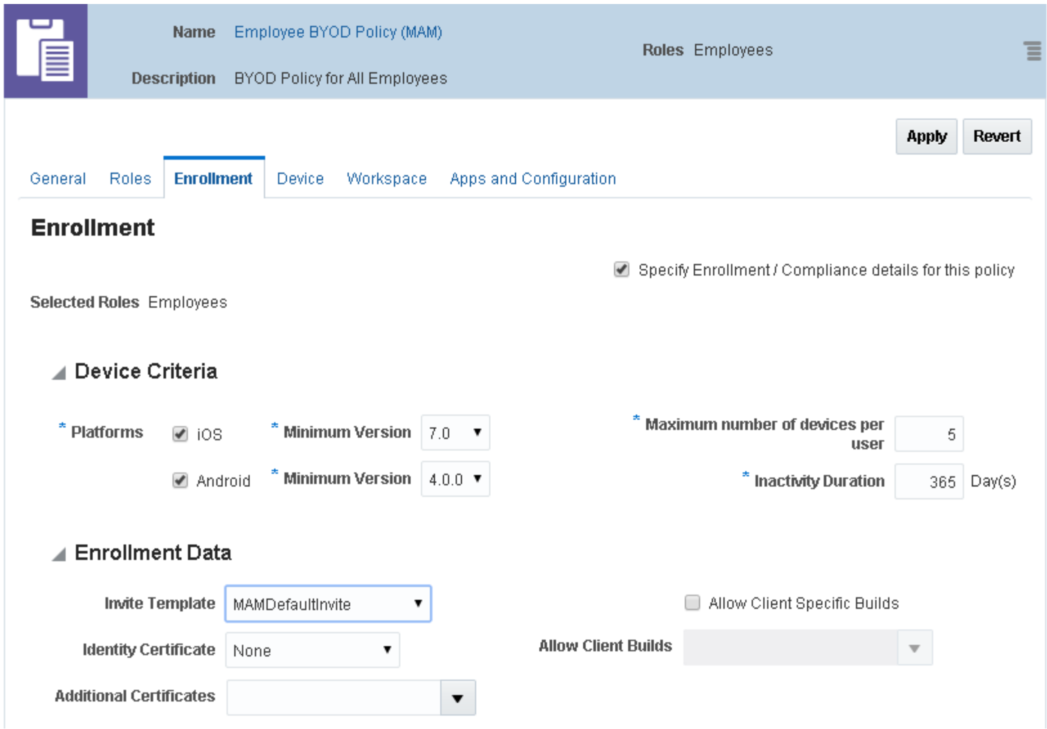
3.2.2 Attaching Invite Templates to a Policy

The invite template sent to a user is based on policy. Use the following steps to attach an invite template to a policy.

3.2.2.1 How to Attach an Invite Template to a Policy

1. Open the Mobile Security Policies page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the policy that you will attach the invite template to, then click the policy record to expand the policy details section.
3. Click the **Enrollment** tab.
4. In the **Enrollment Data** section, locate the **Invite Template** field and select the invite template from the drop-down menu to attach it to the policy.
5. Click **Apply** to save your changes.

Figure 3–2 Attaching an invite template to a policy



Name Employee BYOD Policy (MAM)

Description BYOD Policy for All Employees

Roles Employees

Enrollment

☑ Specify Enrollment / Compliance details for this policy

Selected Roles Employees

Device Criteria

* Platforms ☑ iOS * Minimum Version 7.0

☑ Android * Minimum Version 4.0.0

* Maximum number of devices per user 5

* Inactivity Duration 365 Day(s)

Enrollment Data

Invite Template MAMDefaultInvite

Identity Certificate None

Additional Certificates

☐ Allow Client Specific Builds

Allow Client Builds

3.2.3 Inviting the User

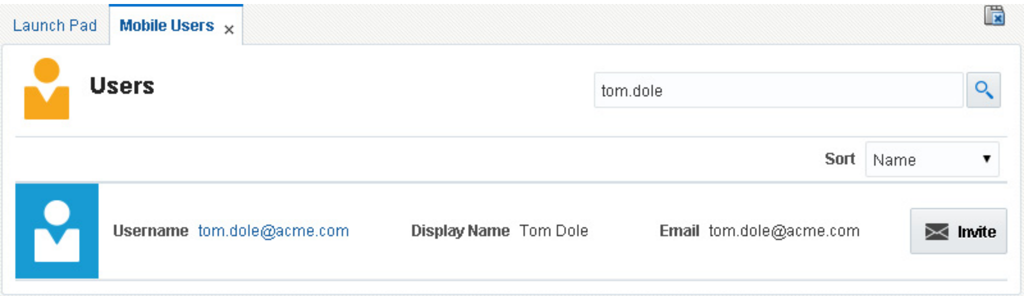
Eligible users can send themselves a mobility program invite, or an administrator can send users an invite.

Administrator Initiated Enrollment

System Administrators and Help Desk Administrators can invite eligible users to enroll a device. If the user is eligible to register a device, an **Invite** button appears with the user record on the Mobile Users page. Clicking the **Invite** button opens a pop-up window so that the e-mail notification can be previewed before it is sent.

Note: The **Invite** button is disabled if the user's LDAP record does not include an e-mail address, or if the user account is disabled.

Figure 3–3 Invite button is part of the user record on the Users page.



tom.dole

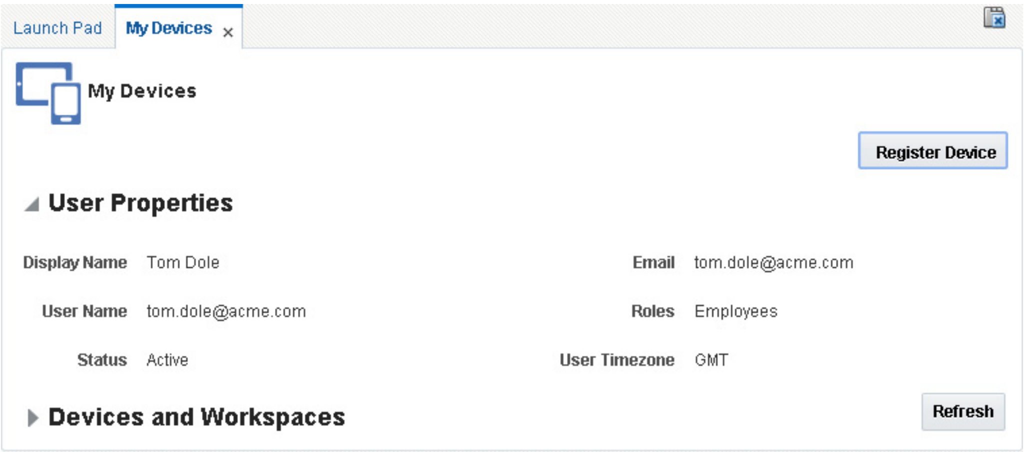
Sort Name

Username tom.dole@acme.com Display Name Tom Dole Email tom.dole@acme.com

Device Enrollment Using the Self-Service Console

Eligible users can send themselves an invite to the mobility program. Users should open the Oracle Mobile Security Suite "My Devices" self-service console (if using the Oracle Access Management console), or the Self-Service Interface (if using the Oracle Identity Manager console). Upon clicking **Register Device** and send, the enrollment e-mail is sent to the user's e-mail address. As described above, the e-mail notification includes steps that the user needs to complete on the mobile device to be enrolled.

Figure 3–4 Register Device button on the "My Devices" self-service console



Display Name Tom Dole

Email tom.dole@acme.com

User Name tom.dole@acme.com

Roles Employees

Status Active

User Timezone GMT

3.2.4 How to Invite a Group of Users by Role Assignment

System Administrators can invite users by role assignment to register with Oracle Mobile Security Suite.

Note: System Administrators can only send registration invites to users in a role that has an associated Mobile Security Policy.

---

1. Open the Mobile Roles page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the role that you want to send the invitation to. To learn how, see Section 5.4.1, "How to Search for a Role in Mobile Security Manager."
3. Click the ≡ at the far right side of the role record row.

   A pop-up menu opens.
4. Choose **Invite**, which invites users with this role assignment to register a device/Workspace with Oracle Mobile Security Suite.

3.3 Enrolling a Workspace (MAM-Only Enrollment)

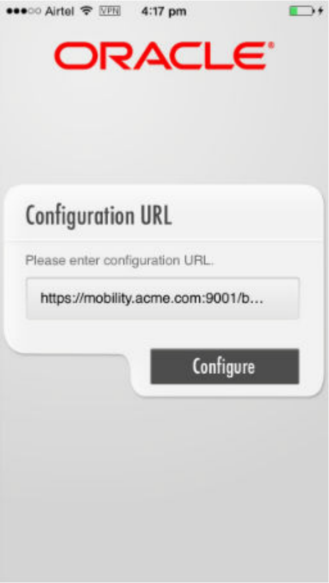
Enrolling a Workspace is a three step process:

1. The user downloads the Workspace app using the link in the invite e-mail.
2. The user launches the Workspace app with the configuration URL (configuration JSON URL) provided in the invite e-mail.
3. The user provides user credentials in the Workspace login page.

Details are provided below.

To download the Workspace app, the user opens the invite e-mail on the mobile device to be enrolled and clicks the download link. The user then launches the app and enters the configuration URL as shown below.

Figure 3–5 Entering the configuration URL (Workspace enrollment)

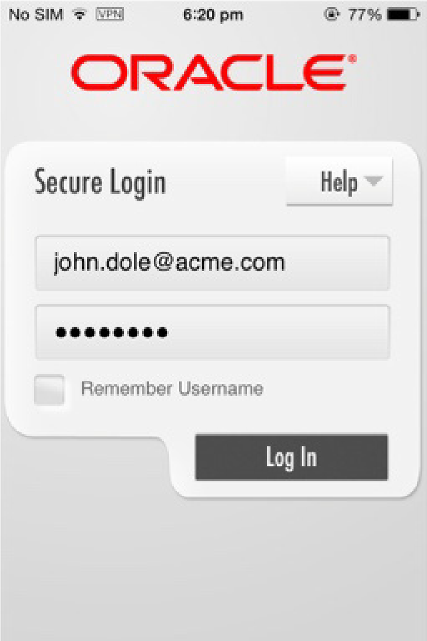


Please enter configuration URL.

https://mobility.acme.com:9001/b...

The Workspace prompts for the user's credentials.

Figure 3–6 Entering user credentials (Workspace enrollment)



john.dole@acme.com

●●●●●●●●●

☐ Remember Username

If authentication is successful, the workspace is enrolled and the Workspace home page opens.

3.4 Enrolling a Device and Workspace (MDM+MAM Enrollment)

To enroll a device as a managed device (MDM+MAM enrollment), a user must have an assigned policy with the "Device" tab enabled. This topic contains separate sections for iOS and Android because the managed device enrollment differs for iOS and Android devices:

- iOS Enrollment
- Android Enrollment

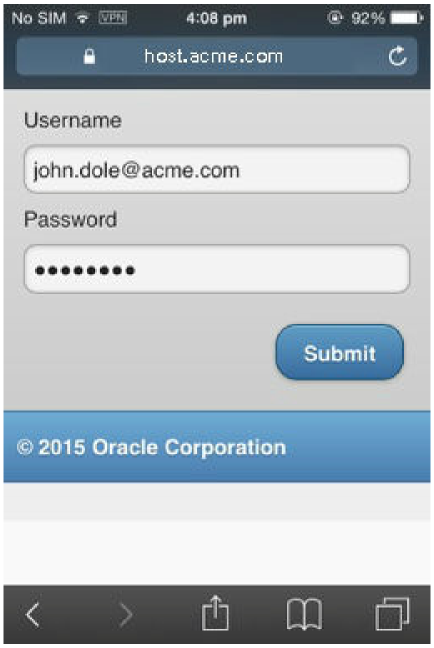
3.4.1 iOS Enrollment

The iOS managed device enrollment launches the iOS MDM Agent enrollment flow. Upon completing the MDM Agent installation, the device prompts the user to install the Secure Workspace app.

1. The user clicks the MDM registration link in the invite e-mail, which launches the Safari browser and opens a login page.

   The user enters their credentials and clicks **Submit**.

Figure 3–7 Entering user credentials (iOS Device and Workspace enrollment)



Username

john.dole@acme.com

Password

•••••••••

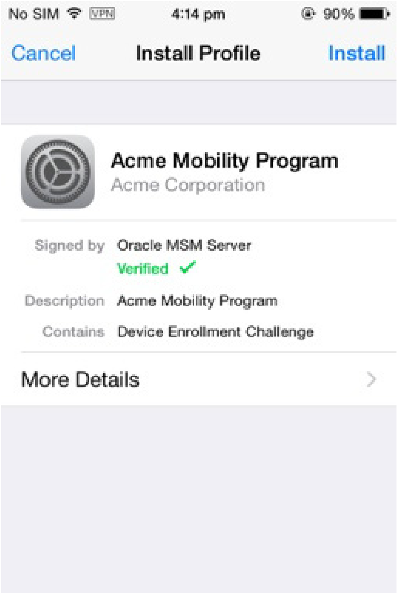
Submit

© 2015 Oracle Corporation

2. Upon successful authentication, the iOS MDM Agent enrollment flow launches. It installs the profile from the Mobile Security Manager server so that the server can manage the device.

   The user clicks **Install**.

Figure 3–8 The iOS MDM Agent enrollment launches



Acme Mobility Program
Acme Corporation

Signed by Oracle MSM Server
Verified ✓

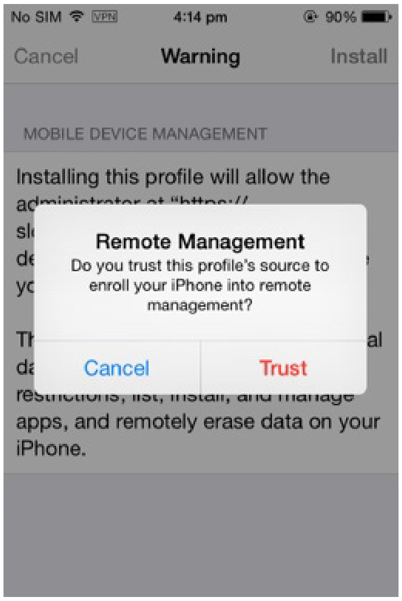
Description Acme Mobility Program

Contains Device Enrollment Challenge

More Details

3. The user clicks **Trust**.

Figure 3–9 The Remote Management Trust dialog (iOS Device and Workspace enrollment)



MOBILE DEVICE MANAGEMENT

Installing this profile will allow the administrator at "https://

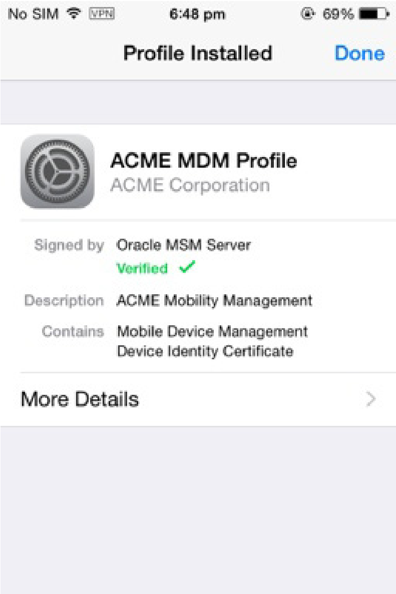
Remote Management

Do you trust this profile's source to enroll your iPhone into remote management?

apps, and remotely erase data on your iPhone.

4. The user clicks **Done**.

Figure 3–10 The Profile Installed screen (iOS Device and Workspace enrollment)



ACME MDM Profile
ACME Corporation

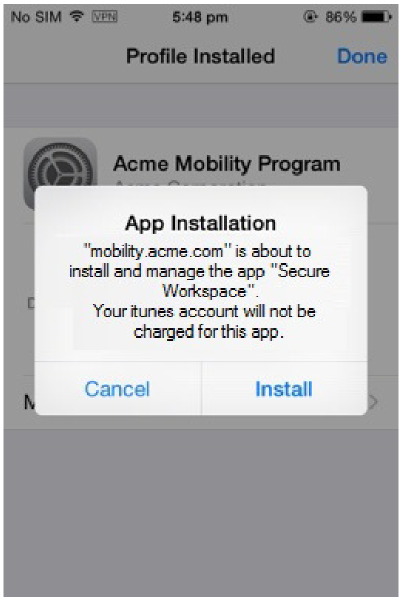
Signed by Oracle MSM Server
Verified ✓

Description ACME Mobility Management

Contains Mobile Device Management
Device Identity Certificate

5. Once the device is successfully enrolled, the device prompts the user to install the Secure Workspace app. The device may also prompt the user to configure additional registration items required by the Mobile Security Manager server. For example, the user may be asked to set a passcode.

Figure 3–11 The Install the Secure Workspace app dialog (iOS Device and Workspace enrollment)



App Installation

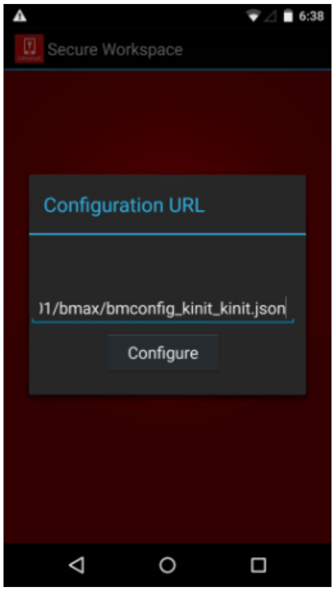
"mobility.acme.com" is about to install and manage the app "Secure Workspace".
Your itunes account will not be charged for this app.

6. The user launches the Workspace app and registers it. See Section 3.3, "Enrolling a Workspace (MAM-Only Enrollment)" for details.

3.4.2 Android Enrollment

1. The user downloads the Secure Workspace app.
2. The user launches the Secure Workspace app, enters the configuration URL provided in the invite e-mail, and clicks **Configure**.

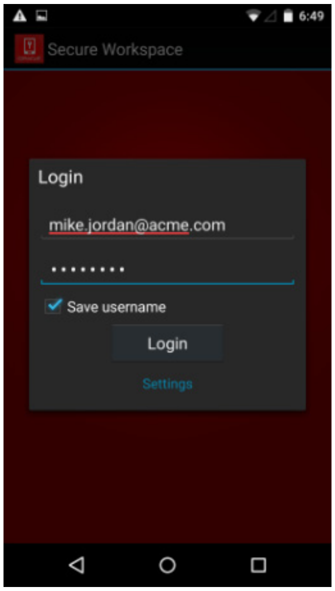
Figure 3–12 Entering the configuration URL (Android Device and Workspace enrollment)



)1/bmax/bmconfig_kinit_kinit.json

3. Once the app is successfully configured, the Login page opens.

Figure 3–13 Entering user credentials (Android Device and Workspace enrollment)



6:49

Secure Workspace

Login

mike.jordan@acme.com

••••••••

☑ Save username

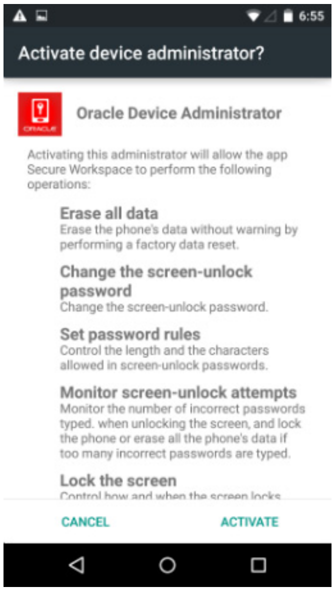
Login

Settings

4. The user enters their credentials and clicks **Login**.

5. Upon successful authentication, the device asks the user to activate the device administrator setting. The user should click **Activate**.

Figure 3–14 The Activate Device Administrator screen



6:55

Activate device administrator?

Oracle Device Administrator

Activating this administrator will allow the app Secure Workspace to perform the following operations:

Erase all data
Erase the phone's data without warning by performing a factory data reset.

Change the screen-unlock password
Change the screen-unlock password.

Set password rules
Control the length and the characters allowed in screen-unlock passwords.

Monitor screen-unlock attempts
Monitor the number of incorrect passwords typed. when unlocking the screen, and lock the phone or erase all the phone's data if too many incorrect passwords are typed.

Lock the screen
Control how and when the screen locks

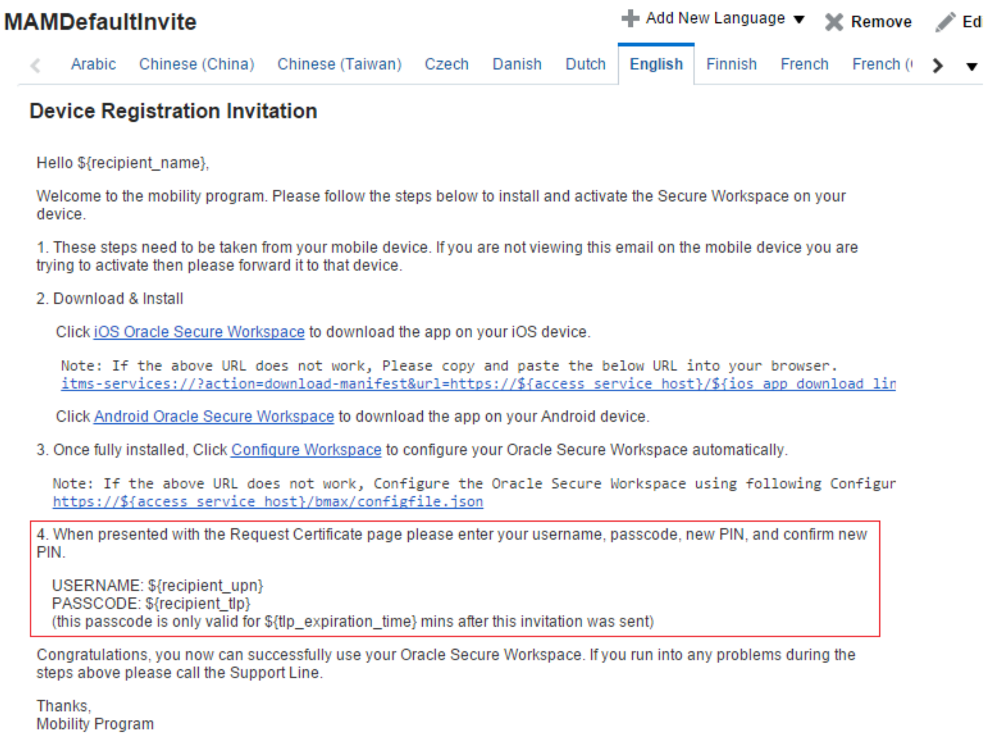
CANCEL ACTIVATE

Mobile Security Manager enrolls the device and Workspace, which are linked together in Mobile Security Manager.

3.5 Enrolling a Workspace in PKINIT Mode

If the authentication mode is PKINIT (Kerberos PKI Based Authentication), a time-limited password (TLP) is required during enrollment. Mobile Security Manager sends the time-limited password to the user in the invite e-mail. A System Administrator should configure the invite template to include the `${tlp_expiration_time}` placeholder value. For details, see "Notification Templates" in the *Help Reference for Oracle Mobile Security Suite Consoles*. A sample invite e-mail with the time-limited password placeholder value is shown in Figure 3–15.

Figure 3–15 Invite e-mail configured with a time-limited password placeholder value



Device Registration Invitation

Hello ${recipient_name},

Welcome to the mobility program. Please follow the steps below to install and activate the Secure Workspace on your device.

1. These steps need to be taken from your mobile device. If you are not viewing this email on the mobile device you are trying to activate then please forward it to that device.

2. Download & Install

Click iOS Oracle Secure Workspace to download the app on your iOS device.

```
Note: If the above URL does not work, Please copy and paste the below URL into your browser.
itms-services://?action=download-manifest&url=https://${access_service_host}/${ios_app_download_lin
```

Click Android Oracle Secure Workspace to download the app on your Android device.

3. Once fully installed, Click Configure Workspace to configure your Oracle Secure Workspace automatically.

```
Note: If the above URL does not work, Configure the Oracle Secure Workspace using following Configur
https://${access_service_host}/bmax/configfile.json
```

4. When presented with the Request Certificate page please enter your username, passcode, new PIN, and confirm new PIN.

USERNAME: ${recipient_upn}
PASSCODE: ${recipient_tlp}
(this passcode is only valid for ${tlp_expiration_time} mins after this invitation was sent)

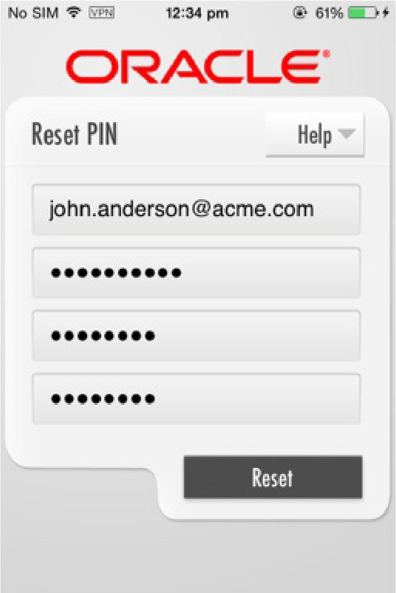
Congratulations, you now can successfully use your Oracle Secure Workspace. If you run into any problems during the steps above please call the Support Line.

Thanks,
Mobility Program

If enrolling a Workspace in PKINIT mode, after the user downloads the Workspace app using the link in the invite e-mail, and after the user launches the Workspace app with the provided configuration URL (configuration JSON URL), the mobile device displays the Reset PIN screen.

Here the user enters his or her user ID, the TLP password included in the invite e-mail, and the new PIN. Once the Workspace is enrolled, the user needs to use the new PIN as the Workspace password going forward.

Figure 3–16 Entering user credentials and resetting the PIN, PKINIT mode (iOS)



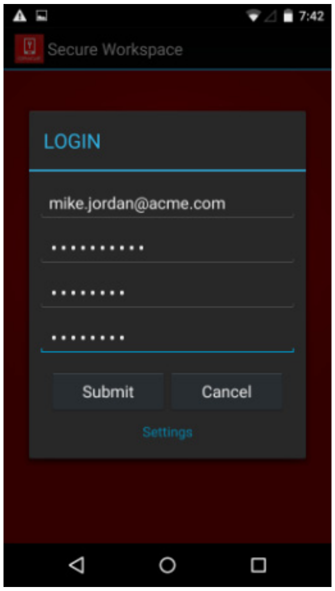
john.anderson@acme.com

●●●●●●●●●●

●●●●●●●●

●●●●●●●●

Figure 3–17 Entering user credentials and resetting the PIN, PKINIT mode (Android)



mike.jordan@acme.com

••••••••••

••••••••

••••••••

4 Managing Devices and Workspaces

This chapter documents device and Secure Workspace management topics. It is organized into the following sections:

- About the Mobile Devices Page in the Mobile Security Manager Console
- Managing Devices and Workspaces
- About Device Configurations
- Managing Devices and Workspaces With Policies
- Multi-User and Kiosk Mode

4.1 About the Mobile Devices Page in the Mobile Security Manager Console

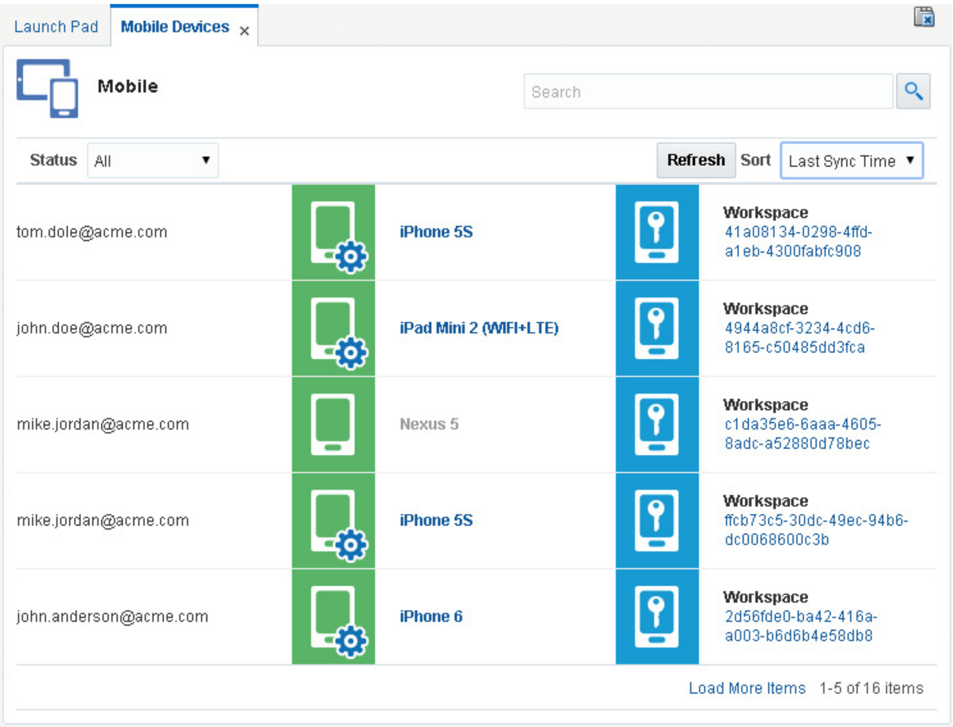
Use the Mobile Devices page to search for and view details about the devices and Workspaces registered to a user, and to perform routine administrative tasks on managed devices (Lock, Wipe, De-register, Sync, Clear/Reset Passcode) and Workspaces (Lock, Unlock, Wipe, Reset Passcode).

Note: To learn how to open the Devices page, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages" and choose **Mobile Devices** from the page menu.

This section includes the following topics:

- Searching for Devices and Workspaces
- How to View Details About a Managed Device or Workspace

Figure 4–1 Mobile Devices page



Status: All

Sort: Last Sync Time

tom.dole@acme.com	iPhone 5S	Workspace 41a08134-0298-4ffd-a1eb-4300fabfc908
john.doe@acme.com	iPad Mini 2 (WIFI+LTE)	Workspace 4944a8cf-3234-4cd6-8165-c50485dd3fca
mike.jordan@acme.com	Nexus 5	Workspace c1da35e6-6aaa-4605-8adc-a52880d78bec
mike.jordan@acme.com	iPhone 5S	Workspace ffcb73c5-30dc-49ec-94b6-dc0068600c3b
john.anderson@acme.com	iPhone 6	Workspace 2d56fde0-ba42-416a-a003-b6d6b4e58db8

Load More Items 1-5 of 16 items

4.1.1 Searching for Devices and Workspaces

A search box is located in the top portion of the Mobile Devices page. Use the search box to search for a device by user name, device ID, workspace ID, device / Workspace display name, or model. Search results are displayed in the bottom portion of the page.

All searches are "contains" searches and are case-insensitive. You cannot use wildcards but partial matches will return results, for example: *mith* will return results for "Smith."

Search results can be *filtered* by status as follows:

- **All** - Show all devices and Workspaces regardless of status.
- **Active** - Show devices and Workspaces that are enrolled in the system, plus *locked* devices and Workspaces. This is the default option.
- **Registered** - Show devices and Workspaces that are enrolled in the system, excluding *locked* devices or Workspaces.
- **De-registered** - Show devices and Workspaces that are no longer enrolled in the system. De-registered devices and Workspaces are omitted from the default search results. Use the **De-registered** status filter to view them.
- **Locked** - Show devices and Workspaces that have been locked and disallowed access to the server. Locked devices and Workspaces are *active*, but are neither *registered* nor *de-registered*.
- **Wiped** - Show devices and Workspaces with a status of wiped, which signifies that all data has been erased. Wiped devices and Workspaces are omitted from the default search results. Use the **Wiped** status filter to view them.

Search results can be *sorted* as follows:

- **Last Sync Time** - Sort Workspace search results chronologically using the date and time that the device was last synchronized. This is the default option.
- **Name** - Sort Workspace search results alphabetically by name.
- **User** - Sort Workspace search results alphabetically by user name.
- **State** - Sort managed device and Workspace search results alphabetically by registration state.
- **Identifier** - Sort Workspace search results alphanumerically by GUID.

In the search results, a white device icon on a green background indicates an *unmanaged device* record, whereas a device and gear icon on a green background indicates a *managed device*. (Managed and unmanaged devices are covered in Section 4.2, "Managing Devices and Workspaces.") A white device and key icon on a blue background indicates a Workspace record. Click an icon to open the device or Workspace record to view additional details.

Table 4–1 Device and Workspace icons

Icon	Description
📱	Indicates an unmanaged device. An unmanaged device is governed by Workspace policies only. Unmanaged devices are typically employee owned.
📱⚙	Indicates a managed device. A managed device is governed by device polices. Managed devices are typically employer owned.
🔑	Indicates a Secure Workspace. The Secure Workspace is the security container deployed to the mobile device that provides secure access to your employer's IT network.

Users with multiple devices will have multiple entries in the search results table (one entry per enrolled device). If a device is de-registered and re-registered, the system reassociates the device with its previous device record if the **Device/Workspace De-registration Policy** (located in Server Settings) is set to **Archive** and not **Delete**. Reassociating device records enables the same device to have multiple records, which can be helpful for tracking and auditing purposes.

Managing a device or Workspace from the Mobile Devices page is limited to actions such as locking an individual device/Workspace, wiping an individual device/Workspace, and so on. Otherwise, the management of device/Workspace enrollment and compliance is driven by mobile security policies. (Policies are covered in detail in the "Managing Mobile Security Policies" chapter.)

4.1.2 How to View Details About a Managed Device or Workspace

This section includes the following topics:

- View Details About a Managed Device
- View Details About a Workspace
- View a List of Apps Installed on a Workspace

4.1.2.1 View Details About a Managed Device

Follow the steps in this section to view details about a specific managed device registered to a user. You cannot manage an unmanaged device.

1. Open the Mobile Devices page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for a user, device, or Workspace as described in Section 4.1.1, "Searching for Devices and Workspaces."
3. Click a managed device record on the Mobile Devices page to open it.

   The managed device details section opens.

   The following table describes the icons that identify each tabbed page.

   Table 4–2 Device management tab icons

Element	Description
📱	Details tab. Click to view device properties.
🪪	Credentials tab. Click to view details about certificates provisioned to the device. (The Credentials tab is not available if the device/Secure Workspace is registered with kinit authentication.)
📋	Policies tab. Click to view policies applicable to the device and the effective policy enforced on the device.

   A horizontal row of buttons called the Action Bar is located near the top the managed device record. Click an action button (for example, **Lock** or **Wipe**) to send a command to the device. See Section 4.2, "Managing Devices and Workspaces" for command descriptions.

4. Choose from the following:

   - To view device properties, click 📱 (the **Details** tab).

     The Device Details tab contains three sections: Basic Properties, Device Properties, and Smart Phone Properties.

     – *Basic Properties* reports mobile account attributes such as platform information, a timestamp indicating when the device was last synchronized with Mobile Security Manager, and **Compliance Level**, which indicates if the device is in compliance with the effective policy. If the device is out of compliance, click the ⓘ icon to view the reason(s) why.

     – *Device Properties* reports device attributes such as model, battery level, and storage capacity.

     – *Smart Phone Properties* reports device attributes such as `currentCarrierNetwork`, `SIMCarrierNetwork`, `carrierSettingsVersion`, and so on.

- To view details about certificates provisioned to the device, click 🪪 (the **Credentials** tab). The Credentials tab is only available if a certificate is issued to the device (PKINIT registration).
- To view policies applicable to the device and the effective policy enforced on the device, click 📋 (the **Policies** tab).

  The Policies tab contains two sections: Applicable Policies, and Effective Policy.

  – *Applicable Policies* list the mobile security policies that are applicable for the device. Click a policy in this section to open a pop-up window where you can view policy details.

  – *Effective Policy* contains limited information about the mobile security policy that is enforced on the device. Specifically, the Effective Policy is the merge of elements across all applicable mobile security policies that apply to the device.

    The **Apps** section provides information about the apps that are assigned to the managed device by the Effective Policy.

For more information, see "Device Management" in the *Help Reference for Oracle Mobile Security Suite Consoles*.

4.1.2.2 View Details About a Workspace

Follow the steps in this section to view details about a specific Workspace registered to a user.

1. Open the Mobile Devices page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for a user, device, or Workspace as described in Section 4.1.1, "Searching for Devices and Workspaces."
3. Click a Workspace record on the Mobile Devices page to open it.

   The Workspace details section opens.

   The following table describes the icons that identify each tabbed page.

Table 4–3 Workspace management tab icons

Element	Description
📱	Details tab. Click to view read-only device and Workspace properties.
▦	Apps tab. Click to search the mobile app catalog for mobile applications and to view a list of apps installed within the Secure Workspace.
🕒	Activity tab. Click to search the Secure Workspace log for events that meet the search criteria and view event details about activities performed on the Secure Workspace app.

Table 4–3 (Cont.) Workspace management tab icons

Element	Description
	Credentials tab. Click to view details about certificates provisioned to the device. (The Credentials tab is not available if the Secure Workspace is registered with kinit authentication.)
	Policies tab. Click to view policies applicable to the device and the effective policy enforced on the device, or to view policies applicable to the Workspace and the effective policy enforced on the Workspace.

A horizontal row of buttons called the Action Bar is located near the top the Workspace record. Click an action button (for example, **Lock** or **Wipe**) to send a command to the Workspace. See Section 4.2, "Managing Devices and Workspaces" for command descriptions.

4. Choose from the following:

- To view Workspace properties, click (the **Details** tab).

  The Workspace Details page includes two sections: Basic Properties and Workspace Properties.

  – *Basic Properties* list information about the Workspace such as the package name and version of the Secure Workspace app, and **Compliance Level**, which indicates if the device the Workspace is installed on is in compliance with the effective policy. If the device is out of compliance, click the ⓘ icon to view the reason(s) why

  – *Workspace Properties* lists additional information about the Workspace such as the configuration URL for the Workspace app hosted by the Mobile Security Access Server, and details about the device that the Workspace is installed on.

- To view apps deployed in the Workspace, click (the **Apps** tab).

  Use search to filter the apps list. If there are more than seven apps, click a radio button to view the apps. Click the app name to view app details in the pop-up.

- To view event details, click (the **Activity** tab).

  Search a log of Workspace activity from this tab. Search for an event by typing an event key word and search looks across fields for matching strings.

  The search results section displays events along with the following information:

  – **Event Source** indicates if the event was initiated by the Device, or if it was initiated by a command sent from the Mobile Security Manager.

  – **Initiated By** indicates the user or system that initiated the operation, for example the *SecureWorskpace*, or a specific user, such as the end user or an Admin user.

  – **Location** indicates the latitude and longitude coordinates where the event took place. If the latitude and longitude are not available then they will be indicated as -1,1.

- – **Date/Time** indicates the timestamp recorded on the device.

- To view details about certificates provisioned to the Workspace, click (the **Credentials** tab).

  The Credentials tab is only available if a certificate is issued to the Workspace (PKINIT registration). If there are auxiliary certificates, those are also shown on the Credentials tab. Click a certificate to view certificate details.

- To view policies applicable to the Workspace and the effective policy enforced on the Workspace, click (the **Policies** tab).

  The Policies tab contains two sections: Applicable Policies, and Effective Policy.

  - *Applicable Policies* list the mobile security policies that are applicable for the Workspace. Click a policy in this section to open a pop-up window where you can view policy details.

  - *Effective Policy* contains information about the mobile security policy that is enforced on the Workspace. Specifically, the Effective Policy is the merge of elements across all applicable mobile security policies that apply to the Workspace.

    For a description of the Workspace policy fields, see "Workspace Policies Tab" in the *Help Reference for Oracle Mobile Security Suite Consoles*.

For more information about the other Workspace Management tabs, see "Workspace Management" in the *Help Reference for Oracle Mobile Security Suite Consoles*.

4.1.2.3 View a List of Apps Installed on a Workspace

You can view a list of apps installed in the Workspace from the Mobile Devices console page. Search for the Workspace, then open the Workspace Management view. The list of installed apps is displayed on the Apps tab.

1. Open the Mobile Devices page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."

2. Search for a user, device, or Workspace as described in Section 4.1.1, "Searching for Devices and Workspaces."

3. Click a Workspace record on the Mobile Devices page to open it.

   The Workspace details page opens.

4. Click (the **Apps** tab).

   The list of installed apps is displayed. Use search to filter the apps list. If there are more than seven apps, click a radio button to view the apps. Click the app name to view app details in the pop-up.

4.2 Managing Devices and Workspaces

This section includes the following topics:

- Understanding Device Management
- Understanding Workspace Management
- How to Lock, Wipe, De-Register, or set a Passcode on a Device or Workspace
- How to Lock, Unlock, and Wipe Devices/Workspaces by Role Assignment

Understanding Device Management

Device management is concerned with managing the mobile device on which the Secure Workspace is installed. Mobile devices are either managed or unmanaged. A *managed device* is governed by MDM+MAM (Mobile Device Management + Mobile Application Management) policies. Typically, managed devices are owned by the employer and provided to employees. An *unmanaged device* is a device that is governed by MAM policies only. Typically, unmanaged devices are employee-owned. The employee retains total control of their apps and data and the device itself, while the employer owns the apps and data in the Workspace. Oracle Mobile Security Suite supports mixed deployments with both managed and unmanaged devices, and provides significant flexibility for employers to choose which devices they want to manage.

Administrators have extensive control over managed devices, but can only view basic information about unmanaged devices. On the Mobile Devices page, managed and unmanaged devices have different icons as shown in Table 4–1. (Managed devices have a gear, unmanaged devices do not.)

On the Mobile Devices page, you can perform the following actions on individual managed devices:

- **Lock the device**. The device PIN or password screen displays and the user is blocked from using the device until the user unlocks it by entering their PIN or password. System Administrators and Help Desk Administrators cannot unlock the device remotely. In the console the device status displays as "Registered"; it does not change to "Locked."
- **Wipe the device**. This command resets the device to its original factory settings, wiping all personal and corporate data. Wipe assures nothing remains on the device for anyone to access. All pending operations are cancelled, and certificates issued to the device are revoked. In the console the device status displays as "De-registered," unless the **Device/Workspace De-registration Policy** (located in Server Settings) is set to **Delete**, in which case the device record is deleted from the server.
- **De-register the device**. Removes the Workspace app, including certificates, restrictions, and other content that was provisioned by the Mobile Security Manager. Containerized apps no longer work, but the user must delete them manually. All pending operations are cancelled, and certificates issued to the device are revoked. Following this action the device is no longer controlled by the server. In the console the device status displays as "De-registered," unless the **Device/Workspace De-registration Policy** (located in Server Settings) is set to **Delete**, in which case the device record is deleted from the server.
- **Sync the device**. Forces the device to synchronize app, certificate, and restriction details with Mobile Security Manager, and to report the current state of 29 device attributes, including `manufacturerDeviceId`, `productName`, `deviceCapacity`, `availableDeviceCapacity`, `batteryLevel`, `phoneNumber`, and so on.
- **Clear Passcode / Reset Passcode**. This action is intended for use if the device user forgets their password.
  - For iOS devices, the **Clear Passcode** command removes the passcode and grants the user access to the device. The user must enter a compliant passcode within the time allotted by the Passcode Expiration setting (defined under Server Settings). The default value is 60 minutes. If a passcode is not entered in time, the device is marked as non-compliant and the system carries out the Passcode Compliance Action defined by policy.

- For Android devices, the **Reset Passcode** command resets the passcode to a new randomly generated passcode. If using the Self-Service Console, the new passcode is displayed on the screen. Otherwise, a Help Desk Administrator or System Administrator should communicate the new passcode to the user.

Understanding Workspace Management

Workspace management is concerned with managing the Secure Workspace installed on the device. Workspaces are indicated by a device and key icon on a blue background, as shown in Table 4–1.

On the Mobile Devices page, you can perform the following actions on individual Workspaces:

- **Lock** / **Unlock**. Locks or unlocks the Secure Workspace. If locked, the Workspace is disabled and the user is blocked from accessing containerized apps and Workspace data. To unlock the Workspace remotely when it is locked, click **Unlock**. To lock the Workspace remotely when it is unlocked, click **Lock**. If the Workspace is locked, the user can enter their password to verify that it is locked. To unlock the Workspace, the user must contact either a System Administrator or a Help Desk Administrator, who can unlock the Workspace remotely. Once the Workspace is unlocked, the user is required to enter their password to log in.
- **Wipe**. Resets the Workspace to its original system state by erasing all of the stored data. This action does not remove the Workspace app. The user can log in to the Workspace again by providing their credentials, but all previously stored data will be lost.
- **Reset Passcode**. Resets the passcode to a new randomly generated passcode that is displayed on the screen. The user must enter the new passcode the next time they open the Workspace.

  Note: The Reset Passcode button is only available if the Workspace is enrolled using certificate-based (PKINIT) authentication. This button is available if a certificate is present, even though a PIN may not be required.

Table 4–4 Summary of the actions an Administrator can take on devices and Workspaces

	Unmanaged Device	Managed Device	Secure Workspace
Lock / Unlock	NA	Locks the device. The device PIN or password screen displays and the user is blocked from using the device until the user unlocks it by entering their PIN or password. System Administrators and Help Desk Administrators cannot unlock the device remotely. The device status displays as "Registered"; it does not change to "Locked."	Locks or unlocks the Secure Workspace. If locked, the Workspace is disabled and the user is blocked from accessing containerized apps and Workspace data. If the Workspace is locked, the user can enter their password to verify that it is locked. To unlock the Workspace, the user must contact either a System Administrator or a Help Desk Administrator, who can unlock the Workspace remotely. Once the Workspace is unlocked, the user is required to enter their password to log in.
Wipe	NA	Resets the device to its original factory state by erasing all of the stored settings, data, and applications.	Resets the Workspace to its original system state by erasing all of the stored data.
De-register	NA	Removes the Workspace app, including certificates, restrictions, and other content that was provisioned by the Mobile Security Manager. Containerized apps no longer work, but the user must delete them manually. All pending operations are cancelled, and certificates issued to the device are revoked. Following this action the device is no longer controlled by the server.	NA
Sync	NA	Forces the device to synchronize with the Mobile Security Manager, to update the app, certificates, restrictions, and other content that was provisioned.	NA
Clear Passcode / Reset Passcode	NA	This action is intended for use if the device user forgets their password. ■ For iOS devices, the Clear Passcode command removes the passcode and grants the user access to the device. The user must enter a compliant passcode within the time allotted by the Passcode Expiration setting. ■ For Android devices, the Reset Passcode command resets the passcode to a new randomly generated passcode. If using the Self-Service Console, the new passcode is displayed on the screen. Otherwise, a Help Desk Administrator or System Administrator should communicate the new passcode to the user.	Resets the passcode to a new randomly generated passcode that is displayed on the screen. The user must enter the new passcode the next time they open the Workspace. The Reset Passcode button is only available if the Workspace is enrolled using certificate-based (PKINIT) authentication. This button is available if a certificate is present, even though a PIN may not be required.

4.2.1 How to Lock, Wipe, De-Register, or set a Passcode on a Device or Workspace

Use these steps to secure data on the device or Workspace.

1. Open the Mobile Devices page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for a user, device, or Workspace as described in Section 4.1.1, "Searching for Devices and Workspaces."
3. Click a device or Workspace record on the Mobile Devices page to open it.

   The device or Workspace details page opens.
4. Click an action button (for example, **Lock** or **Wipe**) to send a command to the device or Workspace. See Section 4.2, "Managing Devices and Workspaces" for command descriptions.

4.2.2 How to Lock, Unlock, and Wipe Devices/Workspaces by Role Assignment

Use these steps to issue a Lock, Unlock, or Wipe command that will affect all users with the selected role assignment.

1. Open the Mobile Roles page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the role that you want to act on. To learn how, see Section 5.4.1, "How to Search for a Role in Mobile Security Manager."
3. Click the ≣ at the far right side of the role record row.

   A pop-up menu opens.
4. Choose a command option from the menu:
   - **Lock** - Lock the devices/workspaces of users assigned to this role. Users can unlock the device by entering a PIN or password. Users cannot unlock the Workspace. To unlock the Workspace, the user must contact either a System Administrator or a Help Desk Administrator, who can unlock the Workspace remotely. Once the Workspace is unlocked, the user is required to enter their password to log in.
   - **Unlock** - Unlock the devices/Workspaces of users assigned to this role.
   - **Wipe** - Wipe the devices/Workspaces that belong to users assigned to this role. If the device is an *unmanaged device*, only the Secure Workspace and all associated user data is deleted; if the device is a *managed device*, all stored settings, data, and applications is erased and the device is reset to its original system state.

4.3 About Device Configurations

Device Configurations allow you to pre-configure e-mail, calendar, Wi-Fi, and VPN settings that you can then assign to policies so that they can be automatically provisioned to users' devices upon device enrollment.

> **Note:** Only *managed* iOS devices support Device Configurations. Device configurations are ignored on Android devices and unmanaged iOS devices.

- E-mail - Configure incoming and outgoing mail server settings.

- Calendar - Configure calendar server settings.
- Wi-Fi - Configure Wi-Fi settings including proxy server settings if a proxy server is required to access the Wi-Fi network.
- VPN - Configure VPN settings for a Layer 2 Tunneling Protocol (L2TP), Point-to-Point Tunneling Protocol (PPTP), or Internet Protocol Security (IPSec) VPN.

In the Mobile Security Manager console, use the Device Configurations page to manage Device Configurations, and on the Policy Management (Create Policy) page, open the **Apps and Configuration** tab to assign Device Configurations to a policy.

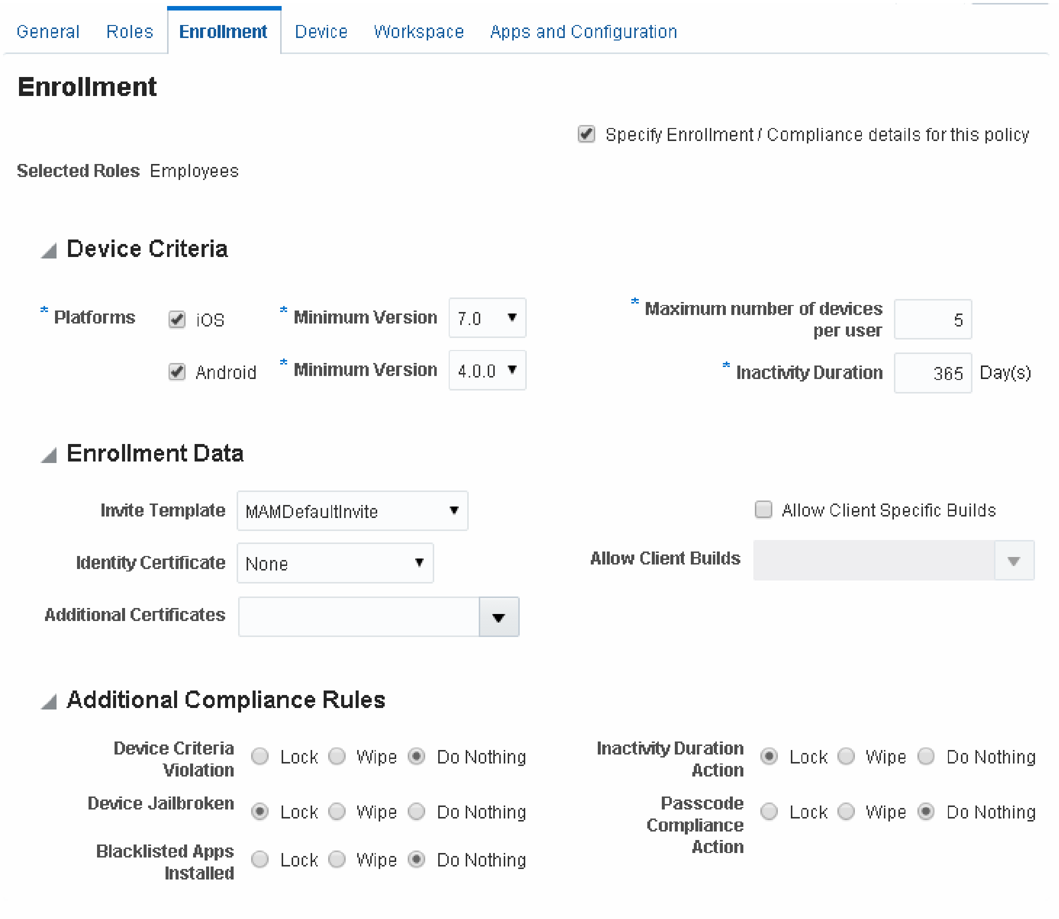
4.4 Managing Devices and Workspaces With Policies

Mobile Security Policies can take action if device or Workspace security properties are out of compliance with the parameters set by the policy. This section discusses how System Administrators can use policies to monitor and restrict devices and Workspaces. For more information about policies, see Chapter 8, "Managing Mobile Security Policies."

Enrollment Policy Settings

The settings on the **Enrollment** tab of the Policies page allow System Administrators to verify that a device is policy compliant when it enrolls with Mobile Security Manager, as well as on an ongoing basis following enrollment.

Figure 4–2 Policies page Enrollment tab



Enrollment

☑ Specify Enrollment / Compliance details for this policy

Selected Roles Employees

◢ Device Criteria

* Platforms ☑ iOS * Minimum Version 7.0 ▾ * Maximum number of devices per user 5

☑ Android * Minimum Version 4.0.0 ▾ * Inactivity Duration 365 Day(s)

◢ Enrollment Data

Invite Template MAMDefaultInvite ▾

Identity Certificate None ▾

Additional Certificates ▾

□ Allow Client Specific Builds

Allow Client Builds ▾

◢ Additional Compliance Rules

Device Criteria Violation ○ Lock ○ Wipe ● Do Nothing

Device Jailbroken ● Lock ○ Wipe ○ Do Nothing

Blacklisted Apps Installed ○ Lock ○ Wipe ● Do Nothing

Inactivity Duration Action ● Lock ○ Wipe ○ Do Nothing

Passcode Compliance Action ○ Lock ○ Wipe ● Do Nothing

The following device criteria can be configured:

- The minimum version of iOS or Android - Indicates the oldest version of the platform software that is eligible for enrollment under this policy. For example, for iOS, setting a value of 8.0 will block iOS 7.0.4 from enrolling.
- The maximum number of enrolled devices per user - Indicates the most devices that a user can enroll under this policy.
- The maximum number of days of inactivity allowed - Indicates the maximum number of consecutive days that the device/Workspace can go without contacting the Mobile Security Manager server before the security action specified in the **Inactivity Duration** element is carried out.

If the properties are out of compliance, the System Administrator can configure the following settings to *lock* or *wipe* the device, or *do nothing*:

- Device Criteria Violation - Indicates the security action to take if either the minimum mobile OS setting or the maximum enrolled devices setting are out of compliance.
- Inactivity Duration - Indicates the security action to take if the Inactivity Duration value is exceeded.

In addition, the system can *lock* or *wipe* the device, or *do nothing* if the following rules are violated:

- Device jailbroken - Indicates the security action to take if the device operating system is found to be jailbroken.
- Blacklisted apps installed - Indicates the security action to take if an app marked as Blacklisted is installed. This compliance rule applies to managed devices, only.
- Passcode compliance action - Indicates the security action to take if the device password value is out of compliance with the policy.

Device Policy Settings

The settings on the **Device** policy tab only apply to managed devices. These settings do not work with unmanaged devices. For more information, see Section 8.3, "Understanding Workspace Policies and Device Policies." The Device policy tab includes the following sections:

- Restrictions - Allows the System Administrator to select device features to restrict. For Android devices, only the camera can be restricted, whereas for iOS devices the camera and 22 other options can be restricted.
- Authentication - Allows the System Administrator to *auto-lock* the device if a maximum idle time is exceeded, and *wipe* the device if the threshold for maximum failed authentication attempts is exceeded. Password policy settings for the device are also set in this section.
- Android Device Encryption - This section features a single option that enables device encryption for Android devices. This option cannot be turned off again once the policy is saved. This option is not available for devices running Android 5.0 (Lollipop) or higher because Device Encryption is always turned on automatically.

Workspace Policy Settings

The settings on the **Workspace** policy tab apply only to the Secure Workspace. The Workspace policy tab includes the following sections:

- Authentication - Allows the System Administrator to configure password requirements for the Secure Workspace, select single user or multi-user (shared Workspace) mode, and set how often the user needs to authenticate. In addition, the policy can *lock* or *wipe* the Secure Workspace, or *do nothing* if the threshold for maximum failed authentication attempts is exceeded.
- Workspace / Apps - Lets the System Administrator select which Workspace and Workspace Apps features to allow or restrict.
- Application Settings - Allows the System Administrator to configure Workspace Apps settings that affect the secure browser, the file manager app, and the PIM app.
- Time Access / Geo Access - Allows the System Administrator to restrict Workspace access by time of day and/or designate locations (cities, states, or countries) where Workspace access is allowed.

4.5 Multi-User and Kiosk Mode

Single User and Multi-User Support

The Secure Workspace can be configured for single user or multi-user use. In multi-user mode, multiple employees can log on to a single Workspace instance on the same shared device for use in environments such as retail outlets, manufacturing floors, and nurses stations. In this mode, local Workspace data is wiped away every time a user logs out of the Workspace so that user data is not shared between sessions. In single user mode, the Workspace is not wiped.

To Enable Single or Multi-User Mode

Multi-User mode (Shared Workspace mode) is enabled by configuring a Mobile Security Policy property.

1. Open the Mobile Security Policies page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the policy to be modified.
3. Click to expand the policy, then click **Workspace**.
4. In the **Authentication** section, locate the **Shared Workspace Mode** property.
   - Choose **Single User** if the Workspace will only be used by a single user on a given device.
   - Choose **Multi-User** if the Workspace will be shared by multiple users on a given device.

Kiosk Mode

Kiosk Mode is only supported on devices that run Android OS. When Kiosk Mode is enabled, users only see the Workspace and the apps within the Workspace. Interaction with the operating system outside the Workspace is minimized. The user cannot close the Secure Workspace app, making this mode suitable for public environments where supervision is minimal, such as lobbies, exhibit spaces, and show rooms.

To learn how to enable Kiosk Mode, see Section 10.2.11, "Enable Kiosk Mode."

5 Managing Users and Mobile Roles

This chapter documents mobile user and mobile role management topics. It is organized into the following sections:

5.1 About the Mobile Users Page and Mobile Roles Page in the Mobile Security Manager Console

Users and roles are managed using your existing directory server. Users are assigned to one or more roles/groups on the directory server; user and role definitions are then referenced by Oracle Mobile Security Suite. System Administrators use roles to associate policies with users, and to perform bulk actions on groups of users based on their role.

Use the Mobile Users page to:

- View basic user information from the connected Identity Store
- Invite a user to register a device/workspace

Use the Mobile Roles page to:

- View role information from the connected Identity Store
- Add Mobile Security Policies to a role (or remove policies from a role)
- Invite users by role assignment to register a device/Workspace in Oracle Mobile Security Suite
- Lock, unlock, and wipe devices and workspaces by role assignment

> **Note:** To learn how to open the Mobile Users page and Mobile Roles page, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages" and choose **Mobile Users** or **Mobile Roles** from the page menu.

5.2 About the Identity Store Directory Server

Oracle Mobile Security Suite requires a constant connection to a directory server to obtain identity information. The directory server is the authoritative source for user and role (group) information, and Mobile Security Manager synchronizes information with the directory server on a scheduled basis. Users and roles cannot be directly added to, or removed from, Mobile Security Manager. Instead perform these tasks using the management console for your directory server.

Supported directory servers include Oracle Unified Directory (OUD), Oracle Internet Directory (OID), Oracle Directory Server Enterprise Edition (ODSEE), and Active Directory. To configure the identity store connection, create an Identity Directory Service Profile in the Oracle Access Management console. Then, in Mobile Security Manager, set the **IDS Profile Name** on the Identity Store Settings tab.

See the logical diagrams in Section 1.5, "Understanding the Oracle Mobile Security Suite Process Flows" for a visual representation of how the identity store directory server interacts with the other Oracle Mobile Security Suite components.

5.3 Managing Mobile Users

This section includes the following topics:

- How to Search for a User in Oracle Mobile Security Suite
- Making Users Eligible to Register a Device
- Managing Passwords
- Disabling User Accounts

5.3.1 How to Search for a User in Oracle Mobile Security Suite

Follow these steps to find a user record in Oracle Mobile Security Suite.

1. Open the Users page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the user by user name, display name, or e-mail address.

   Users that meet the search criteria are listed in the Search Results section of the page.
3. Use the **Sort** menu to sort search results. The following options are provided:
   - **Name** - Sort search results alphabetically by user name.
   - **Display Name** - Sort search results alphabetically by display name.
   - **E-mail** - Sort search results alphabetically by e-mail address.
4. Click the user record to open and close additional user details.

5.3.2 Making Users Eligible to Register a Device

Users are eligible to register a device if there is an **Invite** button next to their user record on the Mobile Users page. To be eligible, the user must be assigned to a role that has an attached mobile policy. Use your directory server management console to assign the user to a role. (To learn how to assign a mobile policy to a role, see Section 5.4.2, "How to Assign a Mobile Policy to a Role.") You cannot use the Oracle Access Management console to add a user to Oracle Mobile Security Suite, nor can you use the console to assign a user to a role.

Note: The **Invite** button is disabled if the user's LDAP record does not include an e-mail address, or if the user account is disabled.

5.3.3 Managing Passwords

Single sign-on (SSO) functionality is provided by Oracle Access Manager. Administrators and users use their SSO credentials to log into the Oracle Mobile Security Suite consoles. To manage SSO passwords, users should use Oracle Identity Manager or a similar system.

Device passwords and Workspace PINs are managed from Mobile Security Manager.

- You can either clear or reset a device passcode if the user forgets it. See Section 4.2, "Managing Devices and Workspaces" for details.
- PIN and Password policy settings, such as password complexity requirements, password expiry settings, and so on are configured using policies. Device password requirements are configured on the **Device** tab, and Secure Workspace password requirements are configured on the **Workspace** tab. See "How to Create or Edit a Mobile Security Policy" for more information.

Note: To learn how to add a "Forgot Password" link to the login screen on mobile devices, see Section 10.1.6, "Customize Password Management" (for iOS), or Section 10.2.9, "Customize Password Management" (for Android).

5.3.4 Disabling User Accounts

You cannot delete a user account from Oracle Mobile Security Suite using the Mobile Security Manager console. The only way to delete a user from Oracle Mobile Security Suite is to delete the user from your directory server.

You can *disable* a user's device or Secure Workspace in Mobile Security Manager as follows:

- In Mobile Security Manager, either de-register the user's device(s) or lock the Secure Workspace(s). If a device is a *managed device*, wiping the device is an irreversible measure that should only be taken if it is necessary to erase all of the device's stored settings, data, and applications. If a device is an *unmanaged device*, only the Secure Workspace (and the user data that it contains) is irreversibly deleted. For details, see Section 4.2, "Managing Devices and Workspaces."
- On your directory server, remove the user from all roles that have a mobile security policy assigned in Oracle Mobile Security Suite.

5.4 Managing Mobile Roles

This section includes the following topics:

- How to Search for a Role in Mobile Security Manager
- How to Assign a Mobile Policy to a Role

The following role-related topics are covered in other chapters:

- Section 3.2.4, How to Invite a Group of Users by Role Assignment

- Section 4.2.2, How to Lock, Unlock, and Wipe Devices/Workspaces by Role Assignment

5.4.1 How to Search for a Role in Mobile Security Manager

Follow these steps to find a role record in Mobile Security Manager.

1. Open the Mobile Roles page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the role by name or description.

   Roles that meet the search criteria are listed in the Search Results section of the page.
3. Use the **Sort** menu to sort search results alphabetically by role name or description.
4. Click the role record to open and close additional role details.

5.4.2 How to Assign a Mobile Policy to a Role

Follow these steps to add a mobile policy to a role or to remove a policy from a role.

1. Open the Mobile Roles page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the role that you want to modify. To learn how, see Section 5.4.1, "How to Search for a Role in Mobile Security Manager."
3. Expand the role record by clicking it.
4. To assign a policy to a role:

   a. In the **Policy** section, click **Add**.

      A new policy row is added to the table.

   b. Type the name of the policy in the **Policy Name** field, then click **Apply**.

      or

      Click the 🔍 to open the policy-picker dialog. Search by role name, policy name, or policy description. Role name search is case sensitive. If you are searching by role name, enter the whole name using the exact sequence of upper and lowercase characters. In the search results, for each policy (or policies) that you are assigning to the role, select the **Add** checkbox, then click the **Add** button to move your selections to the policy table and close the dialog. Click **Apply** to update the role.

      A dialog box confirms the operation if successful.

   To remove a policy from a role:

   a. Click to highlight the policy to be removed.

   b. Click **Remove**.

6 Managing Mobile Apps

This chapter documents Mobile App management, which includes topics such as understanding the Mobile App Catalog, how to add apps to the Secure Workspace, and how to alert users that app updates are available. It includes the following sections.

6.1 About the Secure Workspace App

System Administrators are tasked with configuring the Secure Workspace app and adding it to the Mobile App Catalog. To get started download the Oracle Secure Workspace app from eDelivery and unzip the package to a local directory. Patches are published on ARU. Instructions that document how to configure the Secure Workspace app are located in Chapter 10, "Customizing the Oracle Secure Workspace App." The default Secure Workspace app name is "Workspace," but the "Customizing" chapter documents how to change this name and make other changes, as well.

There can be only one Secure Workspace app instance defined in the catalog. To this one instance you will need to upload an iOS distribution and an Android distribution (assuming your organization supports both platforms). When you add the Secure Workspace app to the Mobile Applications Catalog, select the **Secure Workspace App** property on the Mobile Applications Catalog console page to indicate to the system that the app is the Secure Workspace App. For more information, see Section 6.3.3, "How to Add the Secure Workspace App to the Mobile Applications Catalog."

> **Note:** The Android Secure Workspace app includes an MDM agent for device management. The iOS Secure Workspace app *does not* include an MDM agent because Mobile Security Manager uses the native iOS MDM Agent for device management.

6.2 About the Apps Bundled With the Secure Workspace

The Secure Workspace includes the following built-in apps that System Administrators can enable or disable as needed:

- A secure web browser.

- A mobile file manager for iOS devices. (An Android file manager is available from a third-party vendor.)
- The app catalog.

> **Note:** Oracle Secure Mobile Mail Manager is a personal information manager (PIM) app that can be licensed as an add-on to the suite. The app is an OEM product from Nitrodesk/Symantec that offers e-mail, calendar, contacts, and notes functionality. For details contact your Oracle customer representative.

Access to Workspace apps is managed with Mobile Security Policies (or *policies* for short). See Chapter 8, "Managing Mobile Security Policies" for more information.

This section includes the following topics:

- About the Secure Web Browser
- About the Mobile File Manager
- About the Secure White Pages App

6.2.1 About the Secure Web Browser

The Secure Workspace includes the Secure Web Browser to access corporate intranet resources securely. The Secure Browser ensures that all data (such as bookmarks, cookies, browsing history, and so on) are encrypted and stored within the Secure Workspace.

Secure Web Browser Configuration

You can define the following Secure Web Browser settings using policies:

- Show or hide the Secure Web Browser in the Workspace
- Allow or block downloading using the Secure Web Browser.

To configure these settings follow these steps.

1. Open the Mobile Security Policies page. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the policy to edit and open the Policy Management page.
3. Click **Workspace** > **Application Settings**.
4. Configure the settings:
   - To show the address bar in the Secure Browser, locate **Browser** and select **Address Bar Enabled**. Clear the option to hide the browser.
   - To allow file downloading using the Secure Web Browser, locate **Browser** and select **Download Bar Enabled**. Clear the option to disable file downloading.
   - To enter the path to the network file share, enter the URL in the **File Manager Server-Based URL** field.

Opening Protected URLs in the Secure Web Browser

You can configure the Secure Workspace to open protected URLs in the secure browser. (Or, in other words, block protected URLs from opening in an outside browser.) A protected URL is a web app that is protected behind the Mobile Security Access Server.

To configure this setting:

1. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."
2. Click the **Client Settings** tab.
3. Select the **Open URL in secure browser** option. If you do not select this option, URLs will open in the device's default browser.

6.2.2 About the Mobile File Manager

The Mobile File Manager app for iOS allows users to navigate a network file share from the Secure Workspace and, depending on permissions, download files. The app uses WebDAV to communicate directly to any WebDAV compliant file share. The Mobile File Manager Server converts WebDAV to CIFS to communicate with CIFS shares like Microsoft file shares.

> **Note:** The Mobile File Manager app included with OMSS supports iOS only. For Android devices, a WebDAV client is available from a third-party vendor. For details, contact your Oracle customer service representative.

To create a network file share, you or another admin can create a Shared Folder app on the Mobile Security Manager server that the Mobile File Manager app can connect to. For more information, see Section 6.3.5, "Adding Virtual Apps."

Mobile File Manager Configuration

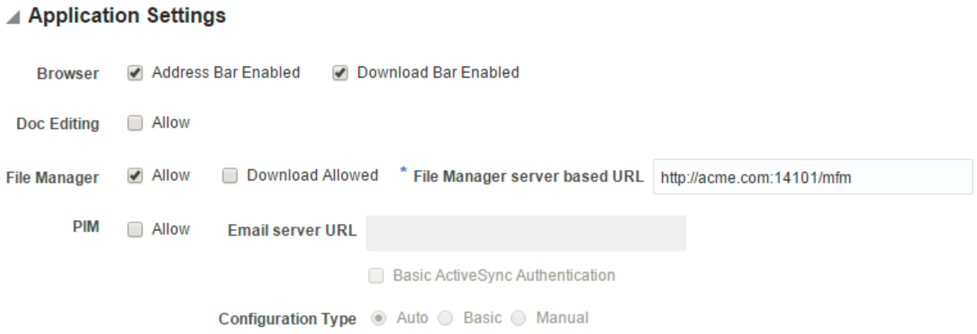
You can define the following Mobile File Manager settings using policies:

- Permission to use the Mobile File Manager app
- Permission to download files and save them locally
- The URL of the service that provides access to network file shares

To configure these settings follow these steps.

1. Open the Mobile Security Policies page. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the policy to edit and open the Policy Management page.
3. Click **Workspace** > **Application Settings**.
4. Configure the settings:
   - To give users full access to the Mobile File Manager app, locate **File Manager** and select **Allow**. Clear the option to restrict access.
   - To allow users to download files and save them locally, locate **File Manager** and select **Download Allowed**. Clear the option to restrict the user from downloading and saving files.
   - To enter the path to the network file share, enter the URL in the **File Manager Server-Based URL** field.

Figure 6–1 To configure the network file share, enter the URL in the File Manager server based URL field



Application Settings

Browser ☑ Address Bar Enabled ☑ Download Bar Enabled

Doc Editing □ Allow

File Manager ☑ Allow □ Download Allowed * File Manager server based URL http://acme.com:14101/mfm

PIM □ Allow Email server URL

□ Basic ActiveSync Authentication

Configuration Type ● Auto ○ Basic ○ Manual

Mobile File Manager Authentication Settings

Configure Mobile File Manager authentication settings on the Mobile Security Manager Settings page:

1. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."
2. Click the **Server Settings** tab.
3. Configure the settings in the **File Manager Settings** section.

   See Section 11.2.4.3, "Configuring Mobile File Manager Authentication Settings" for details.

6.2.3 About the Secure White Pages App

The Secure White Pages App is an iOS and Android mobile corporate directory app that interfaces with existing LDAP directories and can be containerized and deployed as part of the Secure Workspace. The White Pages app only supports OAM-based authentication and is only applicable for OAM deployments.

This app is tightly coupled with Oracle Access Management Mobile and Social. Even though Mobile and Social can be configured to use Active Directory directly, it performs a bind/compare for authentication. Kerberos is not supported.

6.2.3.1 Configure the Secure White Pages App

Before you begin - Verify that Oracle Access Management OAuth Services is enabled and that an Identity Domain is configured. See Configuring OAuth Services for more information.

Add a reports endpoint

1. In a browser window, open the Oracle Access Management console using the appropriate protocol (HTTP or HTTPS). For example:

   `https://`*`hostname`*`:`*`policy-manager-port`*`/access`

2. Click **Federation**, then click **OAuth Service**.

   Choose *Your configured OAuth Domain* > **Resource Servers** > *Your configured User Profile Service*.

3. Expand **Resource URIs** at the bottom of the page.
4. Click the **/users** tab. Do not modify the service endpoint.

5. Expand **entities**.
6. Check if a relationship is present with End Point reports.

   If present, verify that the settings match the following settings; if not, create a relationship with the following settings:

   - **Name** - people_reportees
   - **Identity Directory Service Relation** - reportee
   - **End Point** - reports
   - **Source Entity URI** - manager-uri
   - **Destination Entity URI** - report-uri

Verify LDAP attributes are present in scopes

1. In a browser window, open the Oracle Access Management console using the appropriate protocol (HTTP or HTTPS). For example:

   `https://`*`hostname`*`:`*`policy-manager-port`*`/access`

2. Click **Federation**, then click **OAuth Service**.

   Choose ***Your configured OAuth Domain*** **> Resource Servers >** ***Your configured User Profile Service***.

3. Expand Scopes.
4. Locate the table with the heading **Identity Attributes of the selected scope UserProfile.users for URI /users**. This table is populated with a list of attributes that can be accessed with the `UserProfile.users` scope.
5. Verify that the following attributes are present and add them if they are not:

   - firstname
   - lastname
   - uid
   - mail
   - displayname
   - title
   - manager
   - mobile
   - telephone
   - postaladdress
   - jpegphoto
   - country

6.3 Managing the Mobile App Catalog

System Administrators use the Mobile App Catalog to manage the apps provisioned to devices and Workspaces. The Mobile App Catalog supports the following kinds of apps:

- Native apps – You can either upload a mobile app binary to Mobile Security Manager, or reference an app store URL, such as the iTunes App Store for iOS devices, or Google Play for Android. If uploading a binary, you can upload both containerized and non-containerized apps.

- Virtual apps – Includes Web apps and Shared Folder apps:
  - Web app – A shortcut to a Web URL such as an URL to a corporate portal.
  - Shared Folder – Shortcut to a network file share.

The Secure Workspace app is also available in the Mobile App Catalog. The Secure Workspace app is a special app that is used for MAM (mobile application management) functionality. You can only have one Secure Workspace app in the App Catalog. For more information, see Section 6.1, "About the Secure Workspace App."

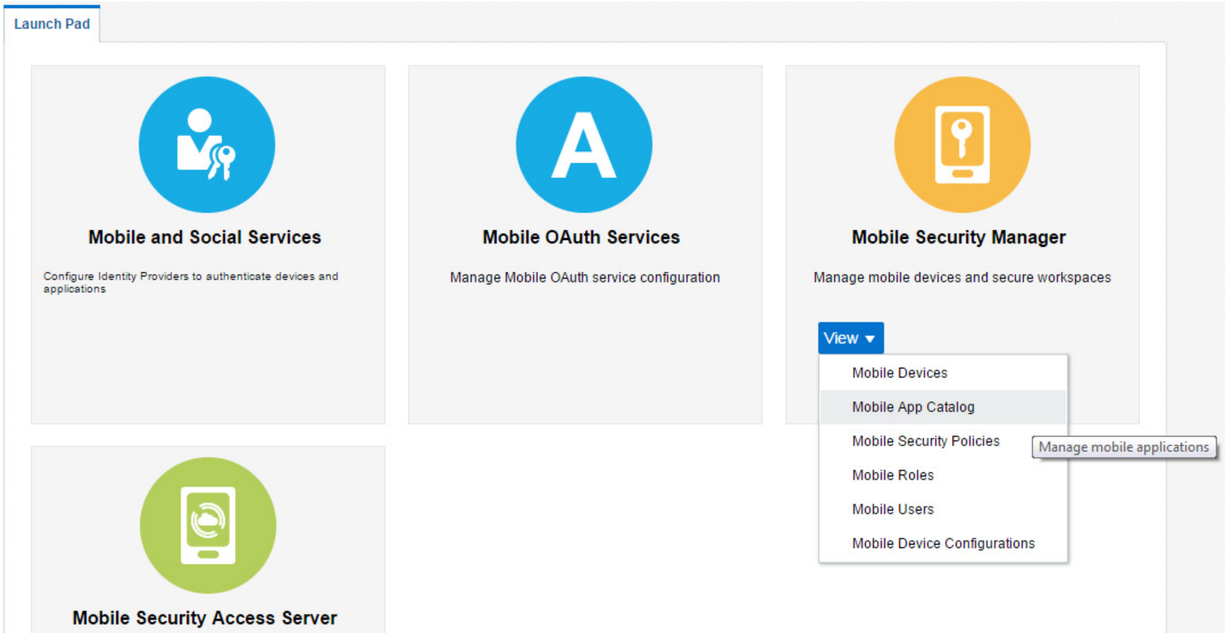
This section includes the following topics:

- Using the App Catalog
- How to Search for an App in the Mobile App Catalog
- How to Add the Secure Workspace App to the Mobile Applications Catalog
- Adding Native Apps
- Adding Virtual Apps
- Understanding the Dynamic App Catalog

6.3.1 Using the App Catalog

Open the App Catalog from the Mobile Security Manager console menu. (To learn how, follow the steps in Section 2.2.2, "Opening the Mobile Security Manager Console Pages" and choose **Mobile App Catalog** from the page menu.)

Figure 6–2 Opening the Mobile App Catalog in the Mobile Security Manager console

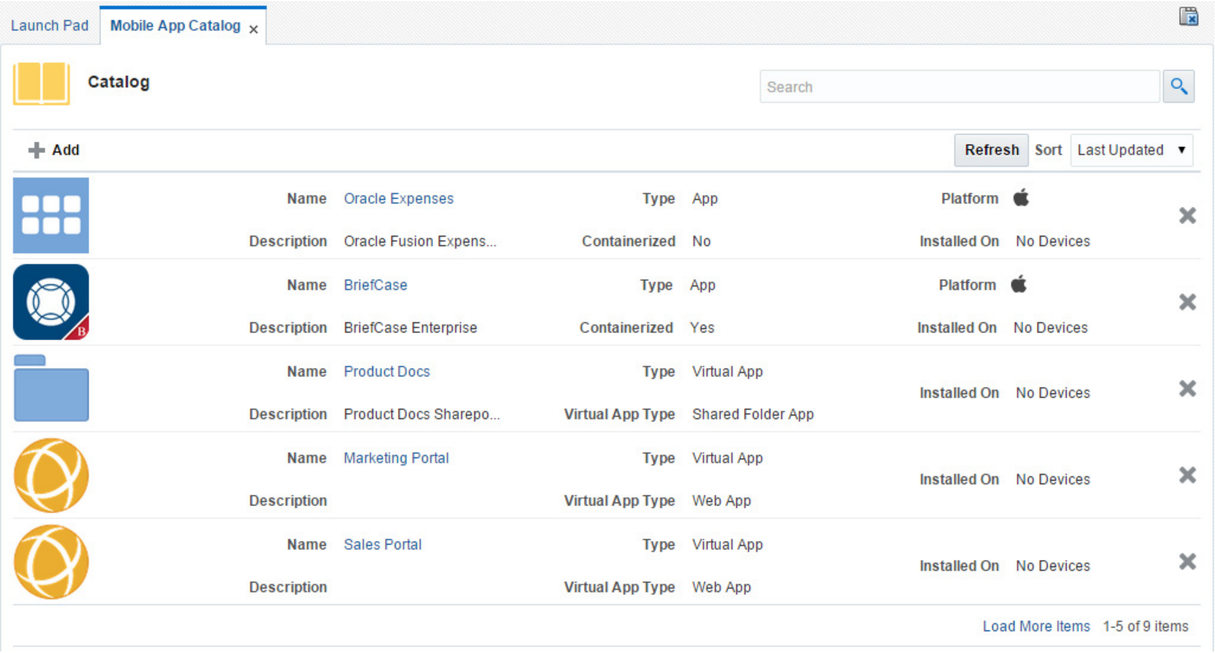

Launch Pad
Mobile and Social Services
Configure Identity Providers to authenticate devices and applications
Mobile OAuth Services
Manage Mobile OAuth service configuration
Mobile Security Manager
Manage mobile devices and secure workspaces
View ▾
Mobile Devices
Mobile App Catalog
Mobile Security Policies
Mobile Roles
Mobile Users
Mobile Device Configurations
Manage mobile applications
Mobile Security Access Server

The Mobile App Catalog homepage lists the apps currently in the catalog. The most recently updated/created apps are listed first.

Use the Mobile Application Catalog page to:

- View existing apps.
- Upload new apps.
- Update existing apps.

Figure 6–3 The Mobile App Catalog homepage

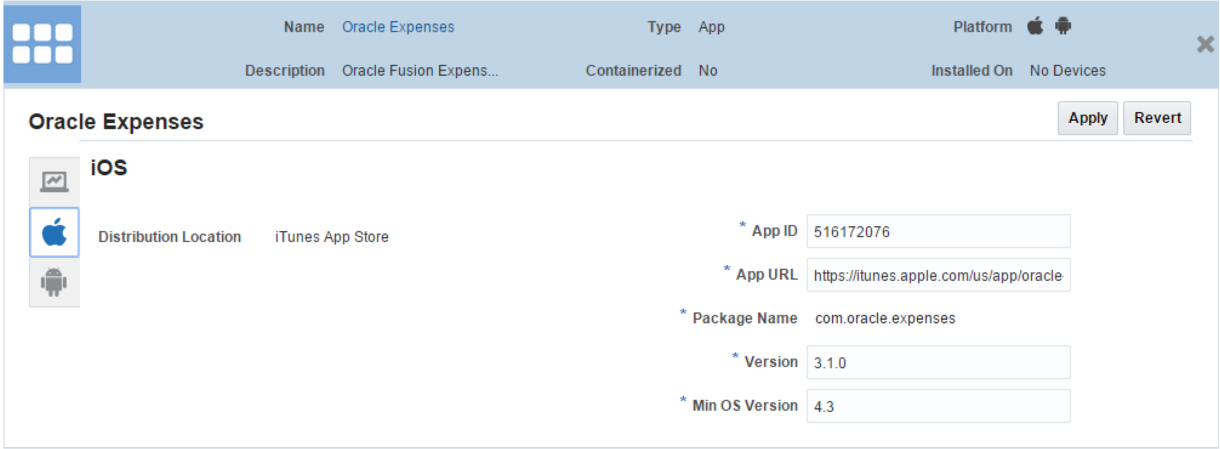


Name	Description	Type	Containerized	Virtual App Type	Platform	Installed On
Oracle Expenses	Oracle Fusion Expens...	App	No			No Devices
BriefCase	BriefCase Enterprise	App	Yes			No Devices
Product Docs	Product Docs Sharepo...	Virtual App		Shared Folder App		No Devices
Marketing Portal		Virtual App		Web App		No Devices
Sales Portal		Virtual App		Web App		No Devices

Load More Items 1-5 of 9 items

Note: App entries in the App Catalog are logical app records that can be referenced in policies. Logical apps can have physical, platform-specific distributions. Figure 6–4 shows a logical app, *Oracle Expenses*, and a tabbed section that details the iOS and Android physical distributions. A logical app record can link to one app for iOS (for example "Expense Wizard 123") and a different app for Android ("Expense King Plus"). The names for the app do not need to be exactly the same across platforms.

Figure 6–4 App detail view showing iOS physical distribution details



Name Oracle Expenses

Type App

Platform

Description Oracle Fusion Expens...

Containerized No

Installed On No Devices

Oracle Expenses

iOS

Distribution Location iTunes App Store

* App ID 516172076

* App URL https://itunes.apple.com/us/app/oracle

* Package Name com.oracle.expenses

* Version 3.1.0

* Min OS Version 4.3

6.3.2 How to Search for an App in the Mobile App Catalog

> **Tip:** To learn how to search a Workspace for a list of installed apps, see Section 4.1.2.3, "View a List of Apps Installed on a Workspace."

Follow these steps to find an app record in the Mobile App Catalog.

1. Open the Mobile App Catalog page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the app by name. Search looks for apps that contain the search text in either the Name or the Description fields.

   Apps that meet the search criteria are listed in the Search Results section of the page.
3. Use the **Sort** menu to sort search results. The following options are provided:
   - **Last Updated** - Sort search results chronologically based on the order that they were updated in the catalog.
   - **Display Name** - Sort search results alphabetically by display name.
4. Click a Name to open and close App Details.

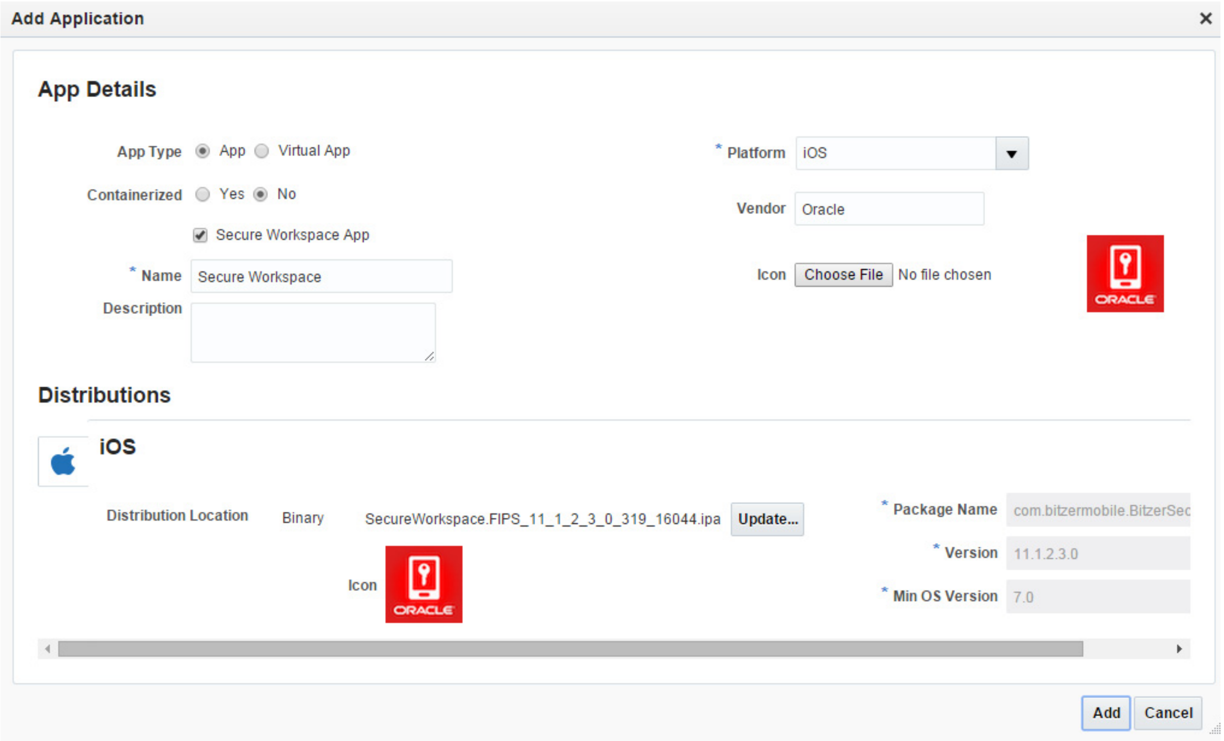
6.3.3 How to Add the Secure Workspace App to the Mobile Applications Catalog

This section describes how to upload the Secure Workspace app to the App Catalog. See Section 6.1, "About the Secure Workspace App" for additional information.

1. Customize the iOS and/or Android distributions of the Secure Workspace app as needed, then finalize the app for uploading to the Mobile Applications Catalog. See Chapter 10, "Customizing the Oracle Secure Workspace App" for details.
2. Open the Mobile App Catalog page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
3. Click **Add**.

   The Add Application dialog opens.

Figure 6–5 Adding the Secure Workspace app in the Add Application dialog



App Details

App Type ● App ○ Virtual App

Containerized ○ Yes ● No

☑ Secure Workspace App

* Name: Secure Workspace

Description:

* Platform: iOS

Vendor: Oracle

Icon: No file chosen

Distributions

iOS

Distribution Location: Binary SecureWorkspace.FIPS_11_1_2_3_0_319_16044.ipa

Icon

* Package Name: com.bitzermobile.BitzerSec

* Version: 11.1.2.3.0

* Min OS Version: 7.0

4. In the **App Type** field, choose **App**.
5. Complete the form as follows:
   - **Containerized**: No
   - **Secure Workspace App**: Checked (Selected)
   - **Name**: Secure Workspace
   - **Description**: The Secure Workspace App
   - **Platform**: Choose **iOS** or **Android** or **All** depending on your requirements
   - **Vendor**: Oracle
   - **Icon**: Optional field—if you do not upload an icon, the icon available in the binary is used. Upload an app icon that will display alongside the app name in the Mobile Security Manager console. Click **Choose File** and navigate to the icon file. The icon file should be in the PNG format. The recommended icon size in pixels is 114 x 114.
6. Click either the iOS or Android tab to configure the distribution. Configure the form as follows:
   - **Distribution Location**: Click **Choose File** to upload the app binary.
   - **Package Name**: The app's complete package name, for example: `com.oracle.secureworkspace`. This read-only value is parsed from the uploaded binary file.
   - **Version**: The version name or number that you want to assign to the app. This read-only value is parsed from the uploaded binary file.

- **Min OS Version**: The minimum version of the mobile operating system software needed to run the app. This read-only value is parsed from the uploaded binary file.

If you are supporting both iOS and Android, repeat to configure the other distribution.

7. Click **Add**.

6.3.4 Adding Native Apps

Administrators can upload Android and iOS apps to the catalog *containerized* or *uncontainerized*. Containerization provides secure storage, policy enforcement, and enhanced security services for authentication and networking. Containerized apps are subject to OMSS security policies; uncontainerized apps are not. To learn how to containerize mobile apps, see Chapter 9, "Using the Oracle Mobile Security Suite Application Containerization Tool."

There are two ways to add a native app to the Catalog: you can upload an app binary, or you can reference an app on a vendor app store. Containerized apps must be uploaded to the Catalog as a binary.

Uploading a Native App as an App Binary

1. Open the Mobile App Catalog page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."

2. Click **Add**.

   The Add Application dialog opens.

3. Complete the form as follows:

   - **App Type**: App
   - **Containerized**: Yes, if the app is containerized; otherwise, No.
   - **Name**: The name of the app. The name is referenced when assigning the app to policies.
   - **Description**: Optional app description.
   - **Vendor**: The name of the app vendor.
   - **Icon**: Click **Choose File** to upload an icon file if you want to use your icon (for example, if the app does not have an icon). By default, the icon bundled in the app is used.
   - **Platform**: Choose the platforms on which the app is supported. You can add additional platforms later on.

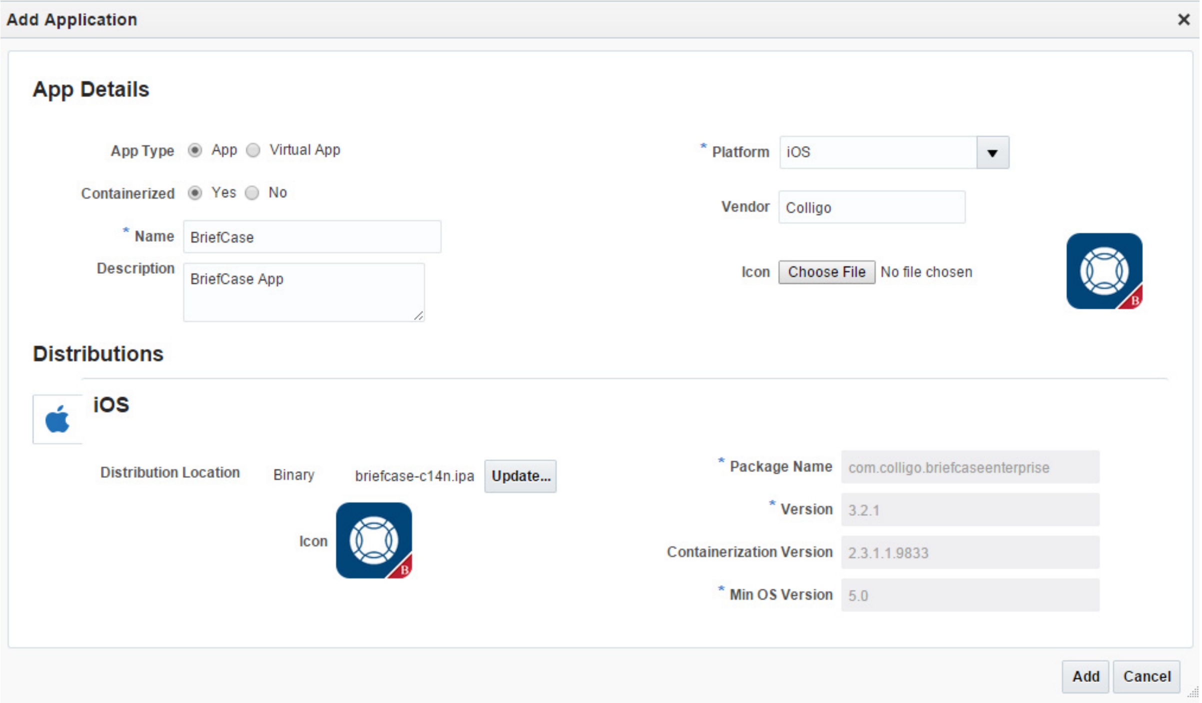
4. Upon selecting **Platform**, the **Distributions** section opens.

   Click **Browse** to upload the app binary.

5. Click **Add**.

6. After the app upload completes and the app is successfully added to the catalog, the icon and other properties (**Package Name**, **Version**, **Containerization Version**, **Min OS Version**) are automatically populated. Note that the UI does not update right away after the app is uploaded. The UI fields are only updated after the app is successfully added to the catalog.

Figure 6–6 An app binary that has been added to the Catalog



Add Application ✕

App Details

App Type ● App ○ Virtual App

Containerized ● Yes ○ No

* Name: BriefCase

Description: BriefCase App

* Platform: iOS

Vendor: Colligo

Icon: Choose File No file chosen

Distributions

iOS

Distribution Location: Binary briefcase-c14n.ipa Update...

Icon

* Package Name: com.colligo.briefcaseenterprise

* Version: 3.2.1

Containerization Version: 2.3.1.1.9833

* Min OS Version: 5.0

Add Cancel

Adding a Reference to a Native App on an App Store

1. Open the Mobile App Catalog page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."

2. Click **Add**.

   The Add Application dialog opens.

3. Complete the form as follows:

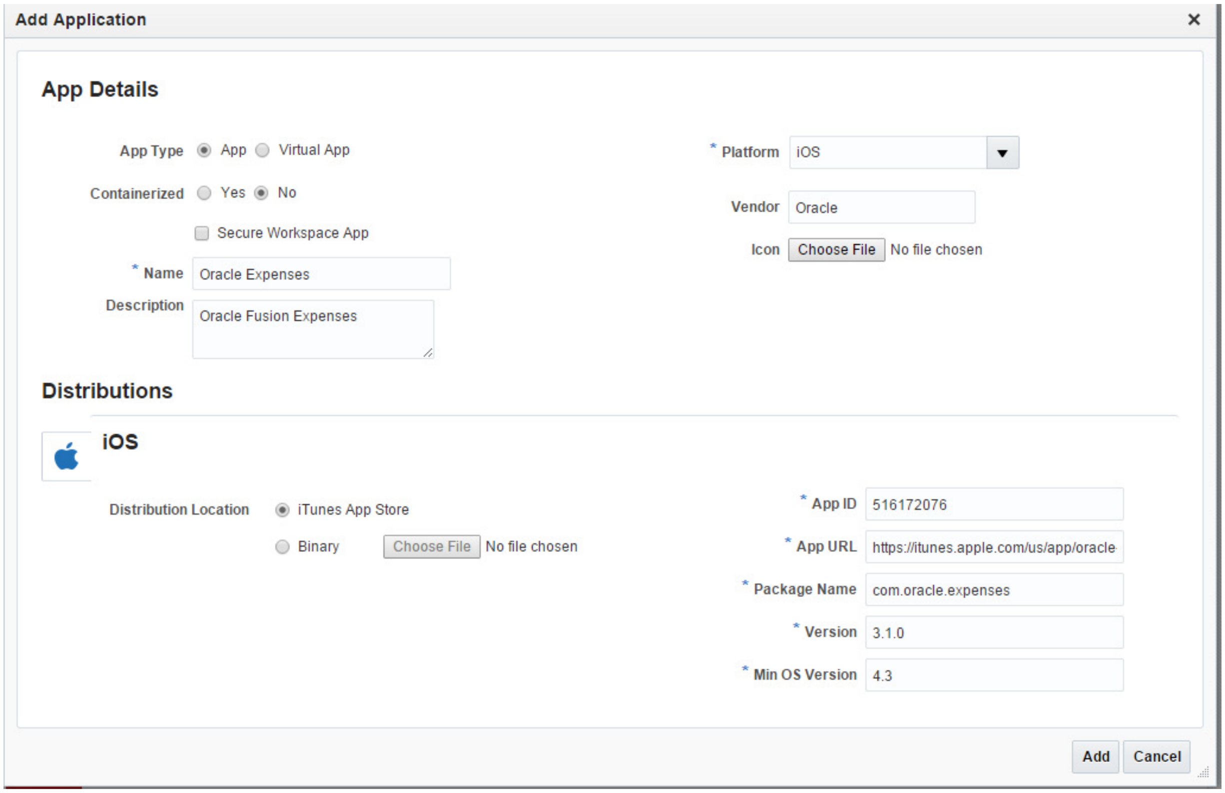
   - **App Type**: App
   - **Containerized**: No (App stores do not host containerized apps.)
   - **Name**: The name of the app. The name is referenced when assigning the app to policies.
   - **Description**: Optional app description.
   - **Platform**: Choose the platforms for which app distributions are available. You can add additional platforms later on.
   - **Vendor**: The name of the app vendor.
   - **Icon**: Click **Choose File** to upload an icon file if you want to use your icon. Otherwise the default App icon in MSM will be used.

4. Upon selecting Platform, the Distributions section opens.

   (Optional) Click Browse to upload the app binary.

5. Enter the app properties (**Package Name**, **Version**, **Containerization Version**, **Min OS Version**) and click **Add**.

Figure 6–7 An app store app that has been added to the Catalog



Add Application ×

App Details

App Type ● App ○ Virtual App

Containerized ○ Yes ● No

□ Secure Workspace App

* Name: Oracle Expenses

Description: Oracle Fusion Expenses

* Platform: iOS ▼

Vendor: Oracle

Icon: Choose File No file chosen

Distributions

iOS

Distribution Location ● iTunes App Store

○ Binary Choose File No file chosen

* App ID: 516172076

* App URL: https://itunes.apple.com/us/app/oracle-

* Package Name: com.oracle.expenses

* Version: 3.1.0

* Min OS Version: 4.3

Add Cancel

6.3.5 Adding Virtual Apps

A virtual app can be a web app that displays in a web browser, or a Shared Folder app that connects to a network file share. Virtual apps are subject to Mobile Security Policies because users access them using either the Secure Web browser or the Mobile File Manager.

Unlike device-native apps that run on the device, web apps are hosted on remote servers and use the web browser installed on the device to display the app. Web apps are popular because they can support multiple devices and platforms, and they can be updated at the server without having to distribute and install binaries across thousands of devices. When you add a web app to a device, you add a shortcut or alias to the Workspace or device home screen. When the user opens the app, the browser launches and the app displays.

A Shared Folder virtual app is a network file share that users access using a file manager app. (Mobile file manager apps are discussed in Section 6.2.2, "About the Mobile File Manager.") Virtual apps can be added to policies, so by adding Shared Folder apps to policies, you can assign network file shares by role assignment.

This section describes how to add virtual apps to the catalog.

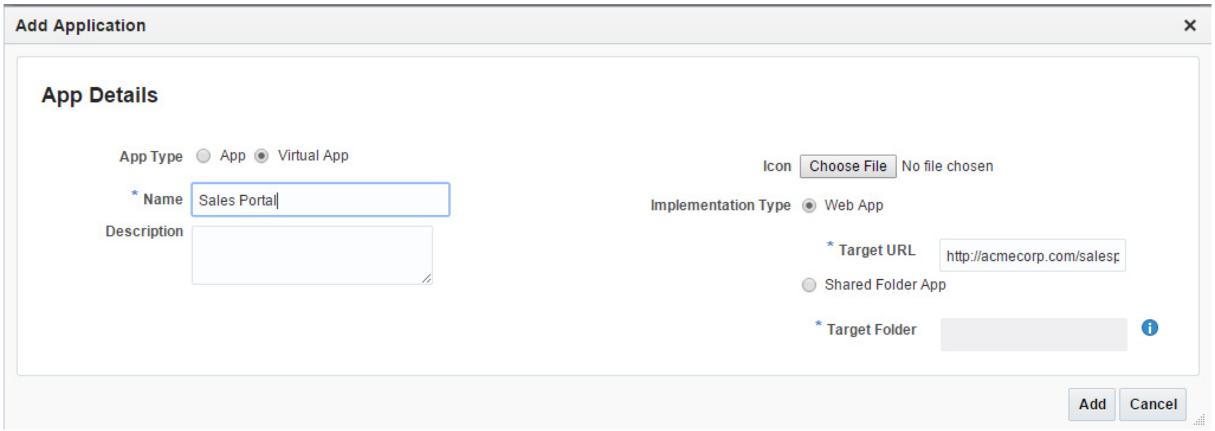
Adding a Web App

1. Open the Mobile App Catalog page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Click **Add**.

The Add Application dialog opens.

3. Complete the form as follows:
   - **App Type**: Virtual App
   - **Name**: The name of the virtual app. The name is referenced when assigning the app to policies.
   - **Description**: Optional app description.
   - **Icon**: Click **Choose File** to upload an icon file if you want to use your icon. Otherwise the default App icon in MSM will be used.
   - **Implementation Type**: Choose **Web App**.
   - **Target URL**: Enter the URL for the Web app.
4. Click **Add**.

Figure 6–8 Adding a Web app virtual app



App Type ○ App ● Virtual App

* Name Sales Portal

Implementation Type ● Web App

* Target URL http://acmecorp.com/salesp

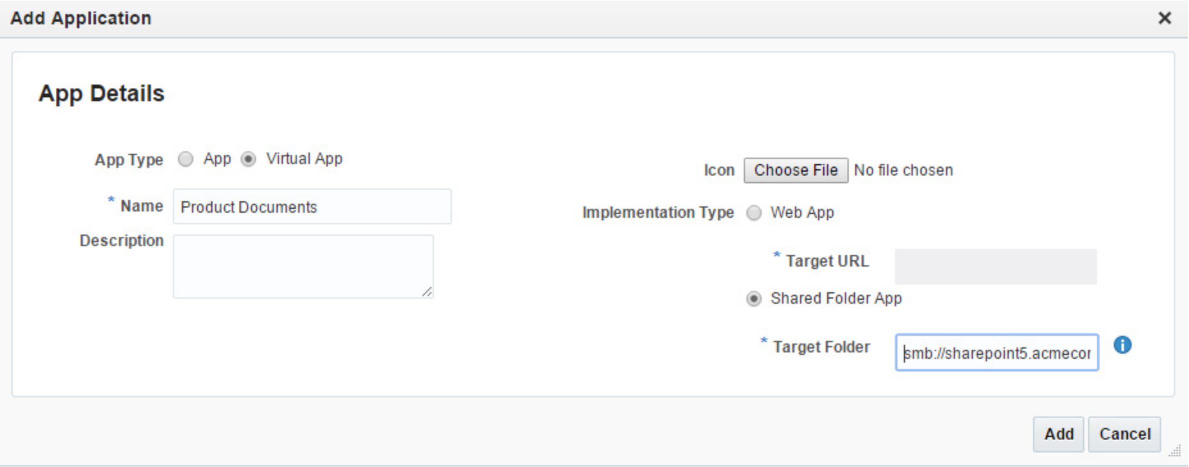
○ Shared Folder App

Adding a Shared Folder App

1. Open the Mobile App Catalog page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Click **Add**.

   The Add Application dialog opens.
3. Complete the form as follows:
   - **App Type**: Virtual App
   - **Name**: The name of the virtual app. The name is referenced when assigning the app to policies.
   - **Description**: Optional app description.
   - **Icon**: Click **Choose File** to upload an icon file if you want to use your icon. Otherwise the default App icon in MSM will be used.
   - **Implementation Type**: Choose **Shared Folder**.
   - **Target URL**: Enter the URL for the shared folder.
4. Click **Add**.

Figure 6–9 Adding a Shared Folder virtual app



Add Application ×

App Details

App Type ○ App ● Virtual App

* Name: Product Documents

Description:

Icon: Choose File No file chosen

Implementation Type: ○ Web App

* Target URL:

● Shared Folder App

* Target Folder: smb://sharepoint5.acmecor

6.3.6 Understanding the Dynamic App Catalog

The Dynamic App Catalog is the custom list of apps available for each individual user. The list is determined by the Effective Policy for the user. The user can browse the dynamic catalog from the Workspace by tapping the catalog icon on the home page.

Figure 6–10 Tap the Catalog icon from the Workspace to view the dynamic catalog

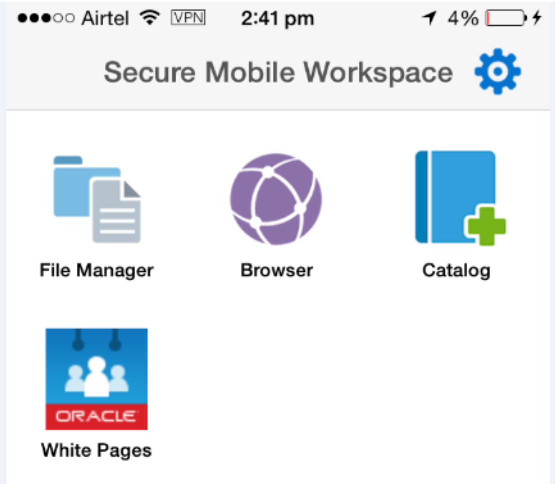
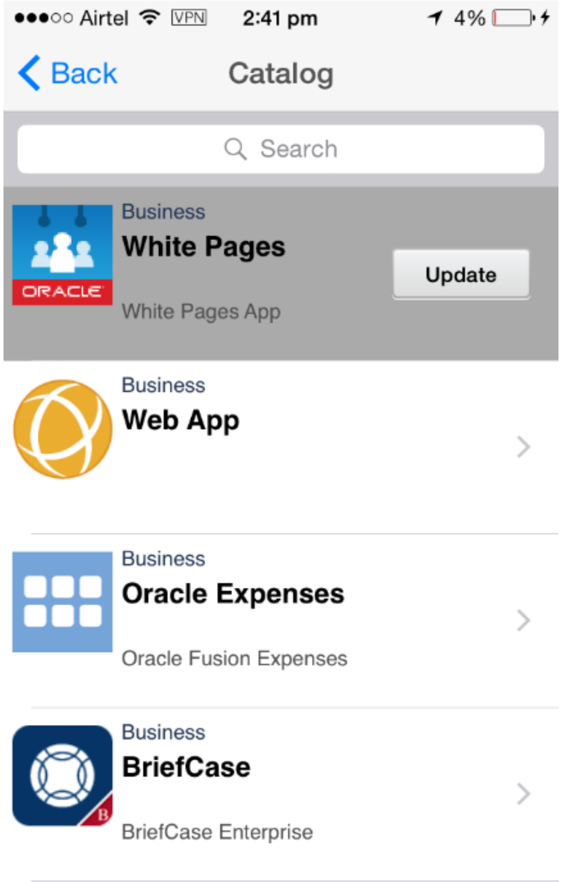


Figure 6–11 The list of apps in this user's Dynamic App Catalog



●●●○○ Airtel VPN 2:41 pm 4%

Back Catalog

Search

Business
White Pages
White Pages App
Update

Business
Web App

Business
Oracle Expenses
Oracle Fusion Expenses

Business
BriefCase
BriefCase Enterprise

6.4 Recovering Secure Workspace and Containerized App Data

Workspace and containerized apps have a feature that allows containerized apps and their saved data to remain accessible after the Workspace has been deleted and then re-installed. This feature allows users who accidentally delete the Workspace and containerized apps to recover their data. This only occurs if the user re-registers the Workspace with the same credentials that were in place when the containerized apps were being used. If the user re-registers with new credentials, the containerized apps will be wiped. Once the Workspace and containerized apps have been wiped, they are no longer recoverable.

6.5 How to Perform Common Mobile App Catalog Tasks

This section describes how to complete common tasks in the Mobile App Catalog. It includes the following topics:

- How to Update or Remove an App in the Catalog
- How to Alert the User That App Updates are Available
- How to Blacklist an App

- How to Include the Mobile App Catalog on the Workspace Home Screen

Note: To learn how to assign apps to users by policy, see Chapter 8, "Managing Mobile Security Policies."

6.5.1 How to Update or Remove an App in the Catalog

1. Open the Mobile App Catalog page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the app. To learn how, see Section 6.3.2, "How to Search for an App in the Mobile App Catalog."
3. To *update* an app in the catalog:
   a. Click the app name to show the App Details form.
   b. Use the form to make your changes and click **Apply**.

      For help understanding the App Details form click Help, or see "Mobile Applications Catalog Page Help" in the *Help Reference for Oracle Mobile Security Suite Consoles*.
   c. To update one or both app binaries, click the iOS or Android tab, click **Choose File,** and navigate to the binary to be uploaded. After the binary uploads, click **Apply**. Repeat if updating the binaries for both mobile operating systems.

      To remove an iOS binary and retain the Android version (or vice versa), click the App Details tab and update the **Platform** settings accordingly.

      Similarly, to expand app support to cover both iOS and Android, open the **Platform** menu and select both **iOS** and **Android**.

   To *remove* an app from the catalog:
   a. Click the **x** icon on the right side of the app catalog record.
   b. Click **OK** in the Remove App dialog.

   Note that removing an app from the catalog will not delete the app from users' devices.

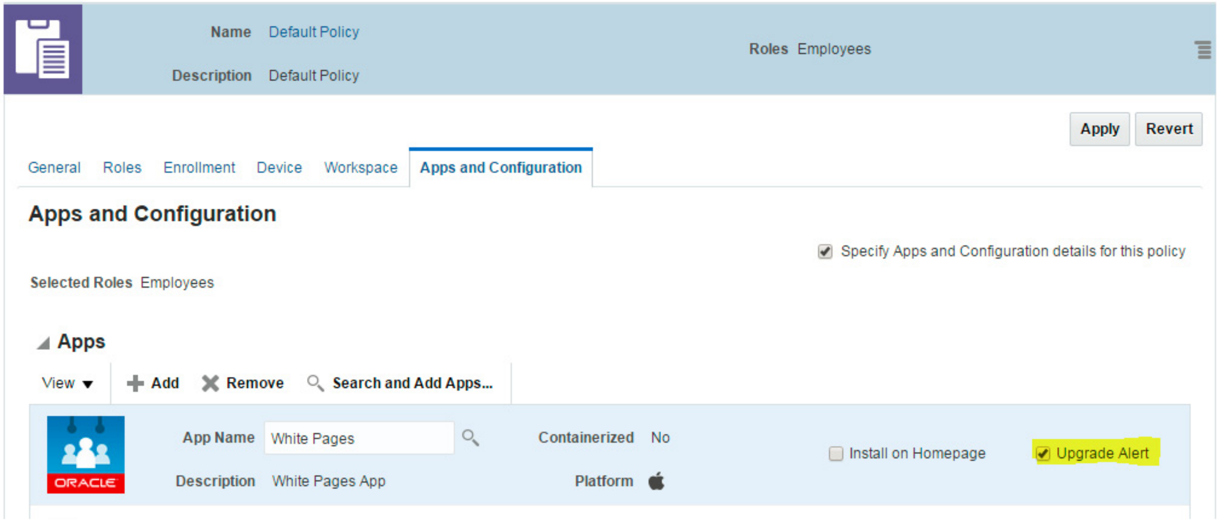
6.5.2 How to Alert the User That App Updates are Available

The **Upgrade Alert** setting is configured at the policy level. If this option is enabled, the app is highlighted in the Workspace to alert the user that an updated version of an installed app is available from the Mobile App Catalog. If the option is not selected, a badge on the catalog app indicates that an update is available, but the system does not alert the user otherwise.

To enable the **Upgrade Alert** setting, follow these steps:

1. Open the Mobile Security Policies page. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the policy to edit and open the Policy Management page.
3. Click **Apps and Configuration** > **Apps**.
4. Select **Upgrade Alert** to alert the user when an app is launched that an upgrade is available. If the option is not selected, a badge on the catalog app indicates that an update is available, but the system does not alert the user otherwise.

Figure 6–12 Select Upgrade Alert to alert the user that an app upgrade is available



Name Default Policy
Description Default Policy
Roles Employees

☑ Specify Apps and Configuration details for this policy

Selected Roles Employees

App Name White Pages
Description White Pages App
Containerized No
Platform
□ Install on Homepage
☑ Upgrade Alert

6.5.3 How to Blacklist an App

Administrators can blacklist specific apps on managed devices only. The blacklist feature checks for device compliance and, depending on how the policy is enforced, the device is locked or wiped if blacklisted apps are found.

1. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."
2. Click **Blacklisted Apps** on the menu bar. (If Blacklisted Apps is not visible, use the arrow buttons to scroll the menu bar to the right. Or, click ▼ to view additional menu items.)
3. Click **Add** to add a row to the Blacklisted Apps table.

   To remove an App from the Blacklisted Apps table, select the app and click **Remove**.
4. Type the name of the application package name that you are blacklisting, then click **Apply**.

6.5.4 How to Include the Mobile App Catalog on the Workspace Home Screen

The Mobile App Catalog app can be added to the Workspace home screen.

1. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."
2. Click the **Client Settings** tab.
3. Select the **Enable add App button** option.

7

Managing Device Configurations

This chapter documents the Device Configurations feature that lets you create pre-configured E-mail, VPN, calendar, and Wi-Fi settings profiles that you can add to policies. It includes the following sections.

- About Device Configurations
- How to Perform Common Device Configuration Tasks

7.1 About Device Configurations

Use Device Configurations to create pre-configured E-mail, VPN, calendar, and Wi-Fi settings profiles that you add to mobile security policies. Because Mobile Security Policies are assigned to roles, users receive the appropriate configuration settings based on their role assignments. Device Configuration updates are pushed to managed devices shortly after the updates are saved. See Section 8.4, "Understanding How Policies are Enforced" for details.

Note: Only *managed* iOS devices support Device Configurations. Device configurations are ignored on Android devices and unmanaged iOS devices.

7.2 How to Perform Common Device Configuration Tasks

This section describes how to complete common Device Configurations tasks. It includes the following topics:

- About the Mobile Device Configurations Page in the Mobile Security Manager Console
- How to Search for a Device Configuration
- How to Create an E-mail, VPN, Calendar, or Wi-Fi Device Configuration
- How to Update or Remove a Device Configuration
- How to Associate Device Configurations with a Policy

7.2.1 About the Mobile Device Configurations Page in the Mobile Security Manager Console

To learn how to open the Mobile Device Configurations page, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages" and choose **Mobile Device Configurations** from the page menu.

Use the Device Configurations Search page to:

- Search for one or more device configurations.
- View and modify device configuration information.
- Add a new E-mail, VPN, calendar, or Wi-Fi configuration, or delete an existing configuration.

7.2.2 How to Search for a Device Configuration

Follow these steps to find a device configuration record.

1. Open the Mobile Device Configurations page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the device configuration by name, description, or type.

   Device configuration records that meet the search criteria are listed in the Search Results section of the page.
3. Use the **Sort** menu to sort search results. The following options are provided:
   - **Last Updated** - Sort search results chronologically so that the most recently updated records display at the top.
   - **Name** - Sort search results alphabetically by configuration name.
4. Click the device configuration record to open and close it.

7.2.3 How to Create an E-mail, VPN, Calendar, or Wi-Fi Device Configuration

1. Open the Mobile Device Configurations page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Click **Add** and choose E-mail, VPN, Calendar, or Wi-Fi from the menu.
3. Complete the configuration form and click **Add** to save it. Refer to Help for detailed information about completing the form, or see the *Help Reference for Oracle Mobile Security Suite Consoles*.

7.2.4 How to Update or Remove a Device Configuration

To *update* a device configuration record, do the following:

1. Open the Mobile Device Configurations page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. At the top of the page, click your user name and choose **Help** from the menu.

   Help for the Device Configurations page opens.
3. Use Search to locate the Device Configuration record that you want to modify. See Section 7.2.2, "How to Search for a Device Configuration" for more information.
4. In the Search results, click the record to select it and expand its properties information.
5. Edit the record. Refer to Help for detailed information about completing the form.

Click **Apply** to save your changes, or click **Revert** to remove your changes and return to the last saved version of the form.

To *delete* a device configuration record, do the following:

1. Open the Mobile Device Configurations page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Use Search to locate the Device Configuration record. See Section 7.2.2, "How to Search for a Device Configuration" for more information.
3. In the Search results, locate the Device Configuration record that you want to delete and click the x (Remove) icon on the right side of the record.

   Click **Yes** in the Remove Device Configuration confirmation prompt, or click **No** to cancel the operation.

7.2.5 How to Associate Device Configurations with a Policy

You provision Device Configurations by adding them to device policies. (A device policy is a policy that has the **Device** tab configured.) Upon completing the following steps, users belonging to groups governed by the policy will get the Device Configurations on their device after the device is enrolled. Device Configurations are provisioned to devices during device enrollment and policy updates to add/remove Device Configurations.

1. Open the Mobile Security Policies page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the policy that you want to use to provision the Device Configuration(s). To learn how, see Section 8.7.1, "How to Search for a Policy Record in Mobile Security Manager."
3. Expand the policy record by clicking it, then click the **Apps and Configuration** tab.

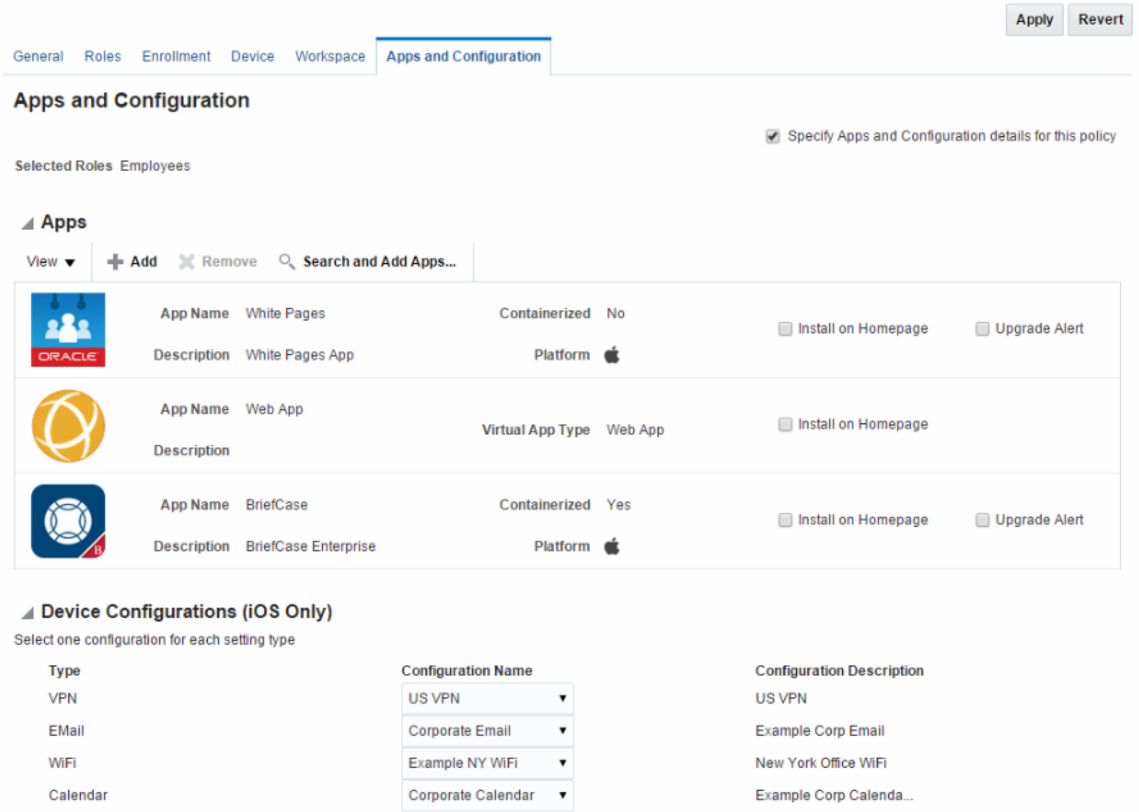
   Note: To configure this page, **Specify Apps and Configuration details for this policy** must be selected.

   To configure the **Device Configurations** section, the **Device** policy tab must be configured. (**Specify Device details for this policy** must be selected.)

   ---

4. Scroll down to the **Device Configurations** (iOS Only) section.
5. Select a Device Configuration from the **Configuration Name** menu.
6. Click **Apply** to save your changes.

Figure 7–1 Selecting Device Configurations from the Edit Policy page



Apply Revert

General Roles Enrollment Device Workspace **Apps and Configuration**

Apps and Configuration

☑ Specify Apps and Configuration details for this policy

Selected Roles Employees

Apps

View ▾ ✚ Add ✖ Remove 🔍 Search and Add Apps...

App Name White Pages Description White Pages App	Containerized No Platform	□ Install on Homepage	□ Upgrade Alert
App Name Web App Description	Virtual App Type Web App	□ Install on Homepage	
App Name BriefCase Description BriefCase Enterprise	Containerized Yes Platform	□ Install on Homepage	□ Upgrade Alert

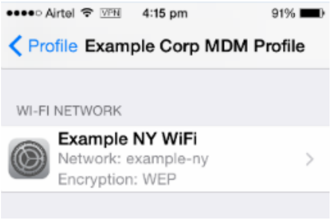
Device Configurations (iOS Only)

Select one configuration for each setting type

Type	Configuration Name	Configuration Description
VPN	US VPN ▾	US VPN
EMail	Corporate Email ▾	Example Corp Email
WiFi	Example NY WiFi ▾	New York Office WiFi
Calendar	Corporate Calendar ▾	Example Corp Calenda...

Once provisioned, the Wi-Fi Device Configuration is included in the MDM profile on the device. If auto-join is enabled, this option will show up in the list of available Wi-Fi networks once the Wi-Fi network is available.

Figure 7–2 Wi-Fi Device Configuration on the Device



●●●●○ Airtel VPN 4:15 pm 91%

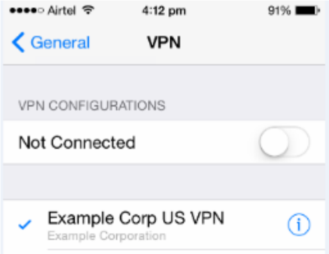
< Profile **Example Corp MDM Profile**

WI-FI NETWORK

Example NY WiFi
Network: example-ny
Encryption: WEP

The provisioned VPN connection appears on the device under **Settings** > **VPN**.

Figure 7–3 VPN Configuration on the Device



●●●●○ Airtel 4:12 pm 91%

< General VPN

VPN CONFIGURATIONS

Not Connected ○

✓ Example Corp US VPN ⓘ
Example Corporation

8 Managing Mobile Security Policies

This chapter documents mobile security policies. It includes the following sections.

- About Mobile Security Policies
- Understanding the Mobile Security Policy Page
- Understanding Workspace Policies and Device Policies
- Understanding How Policies are Enforced
- Understanding How the System Resolves Policy Conflicts
- Creating Manageable Policies
- How to Perform Common Mobile Security Policy Tasks

8.1 About Mobile Security Policies

Organizations use mobile security policies to empower users by provisioning apps to mobile devices and enabling mobile access to corporate file shares. Policies also protect sensitive data by restricting users' actions and access based on role assignments. System Administrators assign policies to roles, not directly to individual users.

Mobile security policies allow organizations to enforce rules around the following security concerns:

- *Device enrollment* - Policies can verify that the device is policy compliant when it enrolls with Mobile Security Manager. Policies can require a minimum version of iOS or Android, and limit the number of devices that a user can enroll.
- *Compliance* - Following enrollment, policies can verify that a device is policy compliant. Policies can look for blacklisted apps, check for extended periods of inactivity, detect if a device is jailbroken, and enforce passcode compliance. Policies also continue to enforce enrollment requirements. If a device is flagged as non-compliant, the policy can lock the Workspace, wipe the device, or take no action depending on configuration.
- *Provisioning* - Policies can define which apps and device configuration settings are available for users belonging to a specific role or group. Note: Device configuration settings (Wi-Fi, VPN, e-mail, calendar) are available for iOS devices only.
- *Time/Geo access* - Policies can restrict employee access to apps in the Secure Workspace container. Access policies include time-based policies (called Time Fence in Mobile Security Manager) that define specific access windows during which apps in the Secure Workspace are available, and location-based policies

(called Geo-fence) that define specific cities, states, or countries from which Secure Workspace access is allowed.

- *Data leakage protection* - Policies can restrict access to mobile device features such as copy/paste, e-mail, instant messaging, video chat, and so on to prevent information from leaving the Secure Workspace.
- *Device restrictions* - MDM (mobile device management) restrictions can be used to restrict access to various device functions, such as the camera and the browser. Refer to the table of device Restrictions in the *Help Reference for Oracle Mobile Security Suite Consoles* for details of supported functions that can be controlled on iOS and Android.
- *Workspace restrictions* - Policy Workspace restrictions affect the Secure Workspace on both managed (MDM+MAM) and unmanaged (MAM-only) devices. Workspace restrictions define access to the built-in Workspace Apps and other Workspace-specific settings.
- *Authentication settings* - Policies can enforce password requirements for device (MDM+MAM) and Workspace (MAM-only) users.

After installing Oracle Mobile Security Suite, one policy named *Default Policy* is present in the system. If no other policies are created after installation, the Default Policy is applied. Be sure to review the Default Policy and modify it to match the controls that are appropriate for your environment.

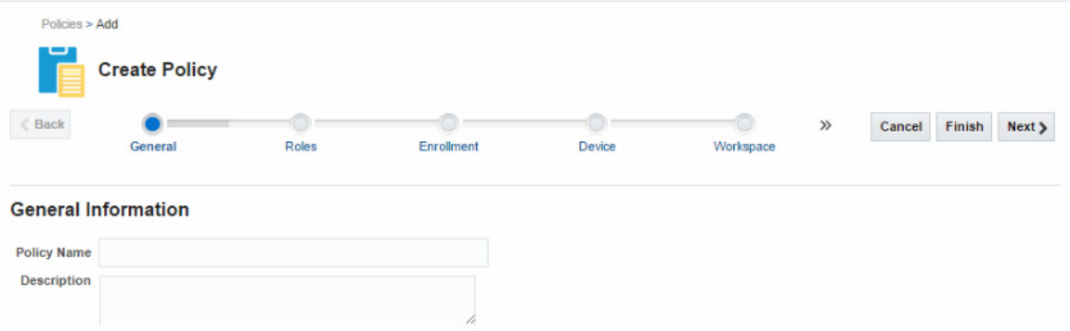
> **Note:** Following installation, you must update the *WorkspaceUsers* placeholder role in the Default Policy with the appropriate role in your identity store. Users belonging to this role/group and its subroles/subgroups will be allowed to enroll Workspaces.

8.2 Understanding the Mobile Security Policy Page

The Mobile Security Policy page consists of the following six configuration tabs:

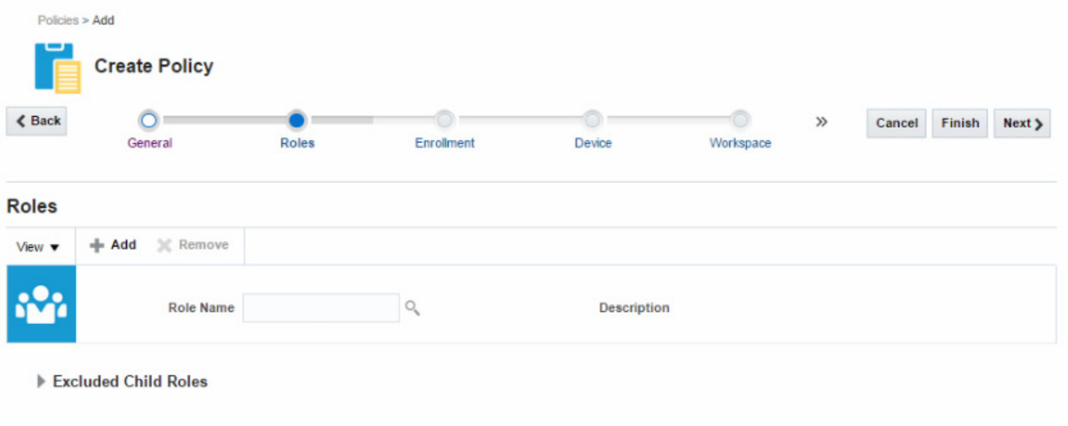
- **General** - This tab includes fields for policy name and description.

Figure 8–1 General tab, page one in the Create Policy flow



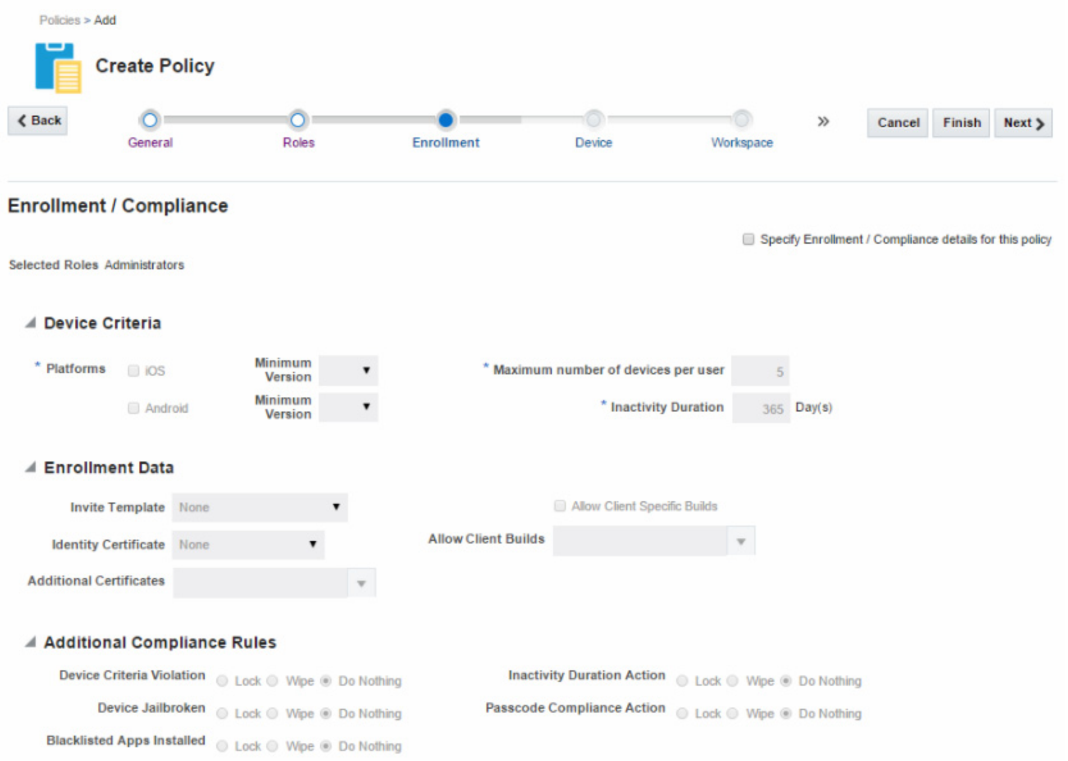
- **Roles** - Used to assign the policy to one or more roles. You can also exempt specific child roles from the policy using this tab.

Figure 8–2 Roles tab, page two in the Create Policy flow



- **Enrollment** - Used to specify enrollment requirements, enrollment details, and compliance rules. Specifying a minimum version of iOS or Android is an example of an *enrollment requirement*, specifying an Invite Template is an example of an *enrollment detail*, and a policy rule that also specifies the security action that should be undertaken if the device is out of compliance is a *compliance rule*.

Figure 8–3 Enrollment tab, page three in the Create Policy flow



□ Specify Enrollment / Compliance details for this policy

Selected Roles Administrators

* Platforms □ iOS Minimum Version ▾

□ Android Minimum Version ▾

* Maximum number of devices per user 5

* Inactivity Duration 365 Day(s)

Invite Template None ▾

Identity Certificate None ▾

Additional Certificates ▾

□ Allow Client Specific Builds

Allow Client Builds ▾

Device Criteria Violation ○ Lock ○ Wipe ● Do Nothing

Device Jailbroken ○ Lock ○ Wipe ● Do Nothing

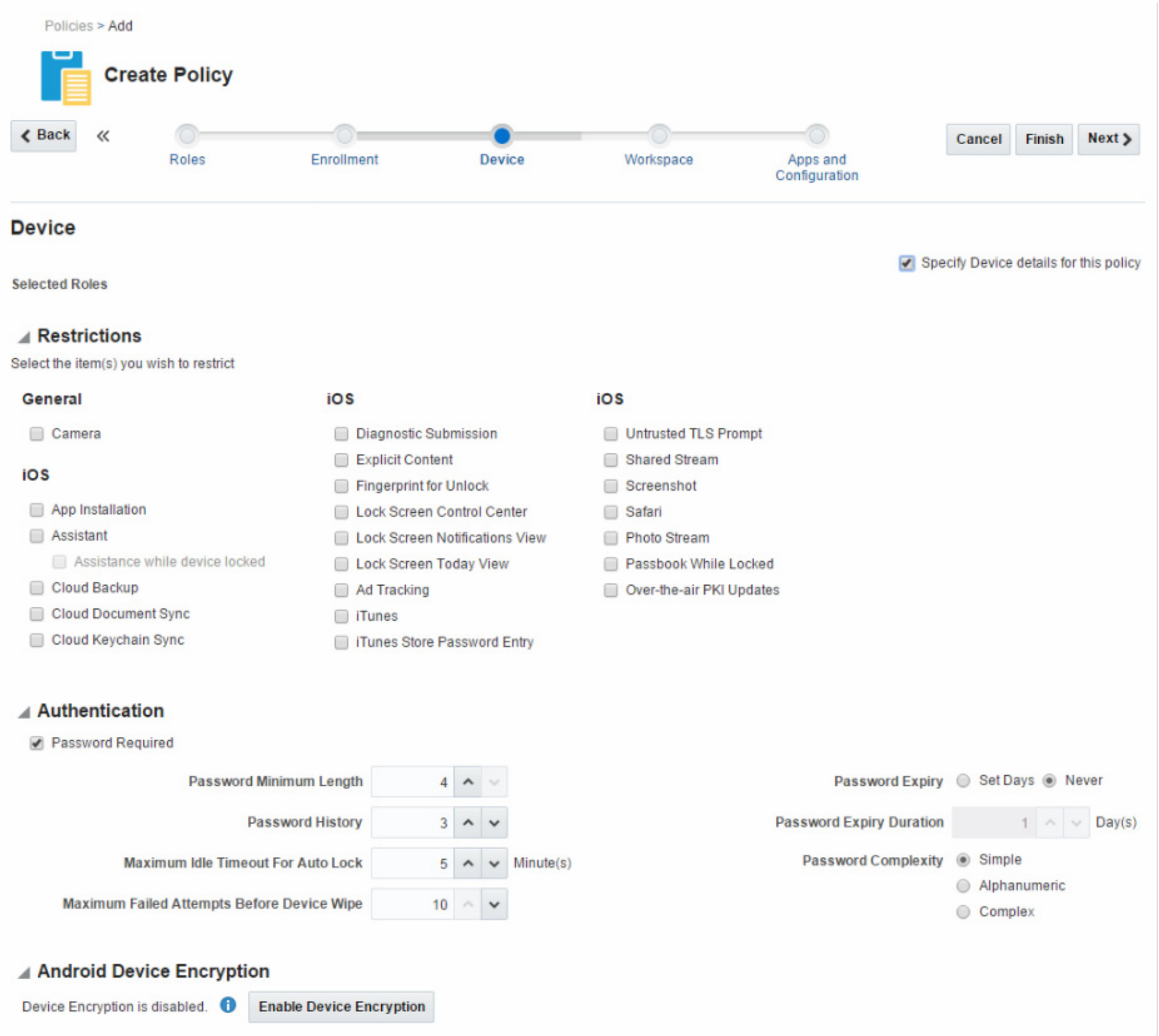
Blacklisted Apps Installed ○ Lock ○ Wipe ● Do Nothing

Inactivity Duration Action ○ Lock ○ Wipe ● Do Nothing

Passcode Compliance Action ○ Lock ○ Wipe ● Do Nothing

- **Device** - Used to restrict device features, specify device authentication details, and enable Android device encryption. Configuring the **Device** tab in a policy activates the MDM deployment mode for the roles associated with the policy. (See Section 8.3, "Understanding Workspace Policies and Device Policies" for details.)

Figure 8–4 Device tab, page four in the Create Policy flow



Device

☑ Specify Device details for this policy

Selected Roles

Restrictions

Select the item(s) you wish to restrict

General

□ Camera

iOS

□ App Installation
□ Assistant
□ Assistance while device locked
□ Cloud Backup
□ Cloud Document Sync
□ Cloud Keychain Sync

iOS

□ Diagnostic Submission
□ Explicit Content
□ Fingerprint for Unlock
□ Lock Screen Control Center
□ Lock Screen Notifications View
□ Lock Screen Today View
□ Ad Tracking
□ iTunes
□ iTunes Store Password Entry

iOS

□ Untrusted TLS Prompt
□ Shared Stream
□ Screenshot
□ Safari
□ Photo Stream
□ Passbook While Locked
□ Over-the-air PKI Updates

Authentication

☑ Password Required

Password Minimum Length 4

Password History 3

Maximum Idle Timeout For Auto Lock 5 Minute(s)

Maximum Failed Attempts Before Device Wipe 10

Password Expiry ○ Set Days ● Never

Password Expiry Duration 1 Day(s)

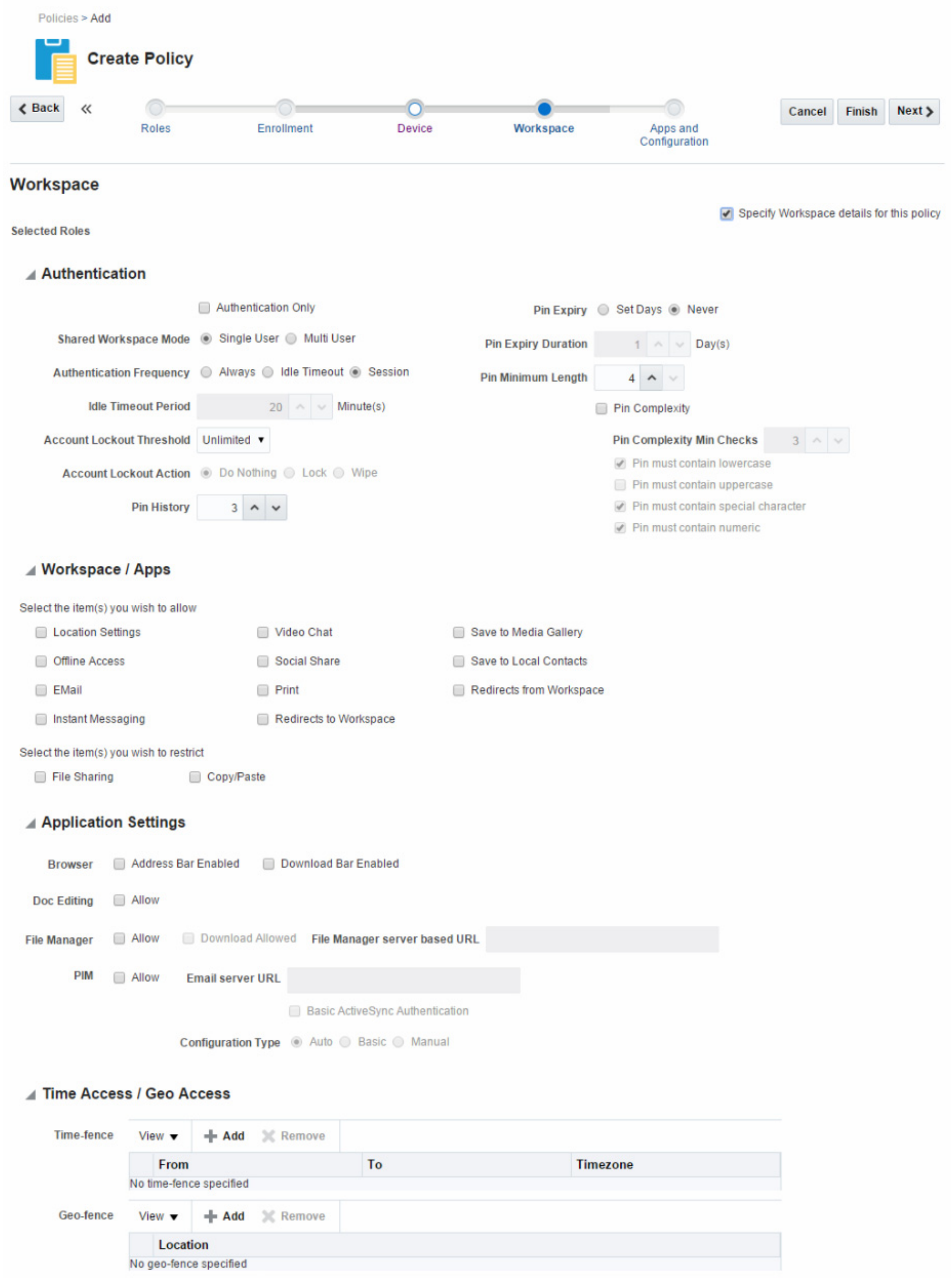
Password Complexity ● Simple ○ Alphanumeric ○ Complex

Android Device Encryption

Device Encryption is disabled. Enable Device Encryption

- **Workspace** - Used to specify Workspace authentication details, allow (or restrict) Workspace privileges, enable and configure application settings, and set Time Access and Geo Access settings. Configure the Workspace tab to create a MAM policy that you will assign to either a managed (MDM+MAM) or an unmanaged (MAM-only) enrolled device.

Figure 8–5 Workspace tab, page five in the Create Policy flow



Policies > Add

Create Policy

< Back « Roles Enrollment Device Workspace Apps and Configuration Cancel Finish Next >

Workspace

☑ Specify Workspace details for this policy

Selected Roles

Authentication

☐ Authentication Only

Shared Workspace Mode ● Single User ○ Multi User

Authentication Frequency ○ Always ○ Idle Timeout ● Session

Idle Timeout Period 20 Minute(s)

Account Lockout Threshold Unlimited ▾

Account Lockout Action ● Do Nothing ○ Lock ○ Wipe

Pin History 3

Pin Expiry ○ Set Days ● Never

Pin Expiry Duration 1 Day(s)

Pin Minimum Length 4

☐ Pin Complexity

Pin Complexity Min Checks 3

☑ Pin must contain lowercase

☐ Pin must contain uppercase

☑ Pin must contain special character

☑ Pin must contain numeric

Workspace / Apps

Select the item(s) you wish to allow

☐ Location Settings

☐ Offline Access

☐ EMail

☐ Instant Messaging

☐ Video Chat

☐ Social Share

☐ Print

☐ Redirects to Workspace

☐ Save to Media Gallery

☐ Save to Local Contacts

☐ Redirects from Workspace

Select the item(s) you wish to restrict

☐ File Sharing ☐ Copy/Paste

Application Settings

Browser ☐ Address Bar Enabled ☐ Download Bar Enabled

Doc Editing ☐ Allow

File Manager ☐ Allow ☐ Download Allowed File Manager server based URL

PIM ☐ Allow Email server URL

☐ Basic ActiveSync Authentication

Configuration Type ● Auto ○ Basic ○ Manual

Time Access / Geo Access

Time-fence View ▾ + Add ✕ Remove

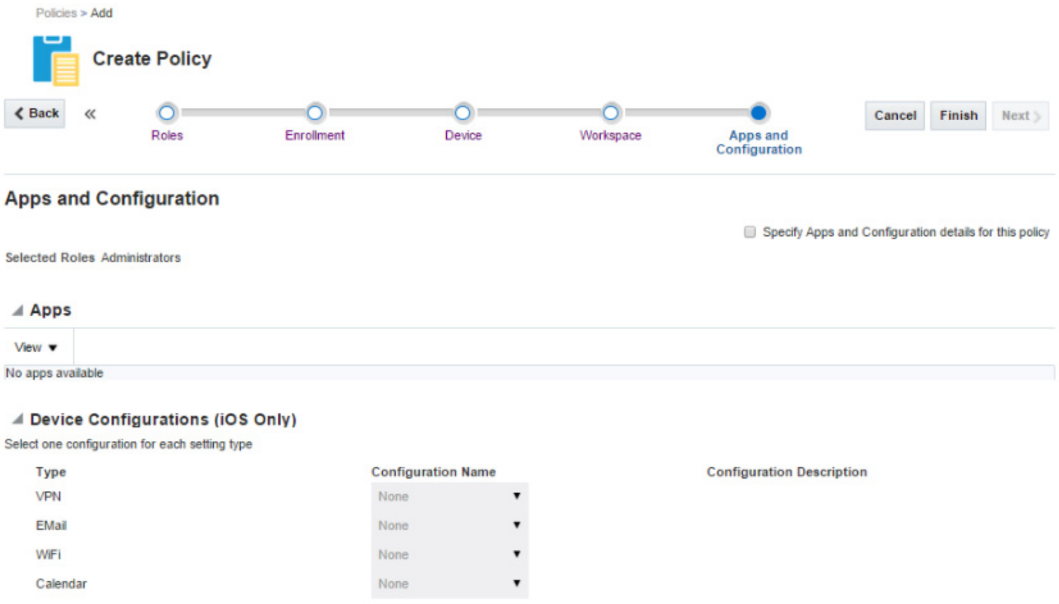
From	To	Timezone
No time-fence specified		

Geo-fence View ▾ + Add ✕ Remove

Location
No geo-fence specified

- **Apps and Configuration** - Used to specify policy-based app assignments and device configurations. A user can only install an app in the Secure Workspace if the app is included in the policy assigned to the user’s role. There are also four device configurations that you can pre-configure and assign to a policy: e-mail, calendar, Wi-Fi, and VPN.

Figure 8–6 Apps and Configuration tab, page six in the Create Policy flow



Apps and Configuration

□ Specify Apps and Configuration details for this policy

Selected Roles Administrators

No apps available

Select one configuration for each setting type

Type	Configuration Name	Configuration Description
VPN	None	
EMail	None	
WiFi	None	
Calendar	None	

Note: Also see Section 4.4, "Managing Devices and Workspaces With Policies" for an overview of how System Administrators can use policies to monitor and restrict mobile devices and Workspaces.

8.3 Understanding Workspace Policies and Device Policies

Mobile security policies are either Workspace policies (MAM policies) or device policies (MDM policies). To review the differences between MAM-only and MDM+MAM enrolled devices, see "MAM and MDM" in the "Understanding Oracle Mobile Security Suite" chapter. A device policy is a policy that has the **Device** tab configured, and a workspace policy is a policy that has any tab *other than* the **Device** tab configured. Remember, the **Device** tab is used to restrict device features and specify device authentication details.

If an invite template is attached to a policy that has the Device tab configured, the managed device enrollment must succeed before the Workspace enrollment can take place. If you do not intend to enroll a device as a managed device, then be sure to attach the invite template to a policy that does not have the Device tab configured.

If your environment includes a mix of managed and unmanaged devices, avoid assigning a device policy to a role that is associated with users who have enrolled unmanaged devices. An unmanaged device cannot enforce a device policy.

> **Tip:** If your environment includes a mix of managed and unmanaged devices, consider making device polices readily identifiable by using a naming convention, such as adding a `MDM_` prefix or `_MDM` suffix to the policy names. For example: `MDM_FieldSupervisor` or `DispatcherIV_MDM`.

To simplify the management of managed and unmanaged devices, consider restricting one mode or the other to only a few mobile roles, for example limit unmanaged devices to employees and require managed devices for contractors.

8.4 Understanding How Policies are Enforced

Policies are assigned to roles, not directly to individual users. For each individual user, Mobile Security Manager merges the user's policy assignments (the *applicable policies*) and arrives at the *Effective Policy*, which is the merge of policy elements across the applicable policies. When conflicting policies apply to a user, Mobile Security Manager typically enforces the policy rules that are the most restrictive. (See Section 8.5, "Understanding How the System Resolves Policy Conflicts" for details.) The Effective Policy is the policy that is enforced on the device or Workspace. To view a user's Effective Policy, use the Mobile Devices page to search for and view details about the devices or Workspaces registered to the user.

All policies are managed on the server but enforced on the client. Therefore policies are enforced even when the client is offline. Policy updates take effect on the client as follows:

- If a *Workspace policy* is updated on the server, the user sees the policy update at the next "heartbeat" interval when the Workspace checks the server for updates. The Secure Workspace checks for policy updates and app updates based on the Workspace **Poll Interval** setting in **Client Settings**. The poll interval is set to 60 seconds by default.
- If a *device policy* is updated on the server, a scheduled task processes it within five minutes and sends a notification to the device by way of the APNS (Apple Push Notification Service) or GCM (Google Cloud Messaging) servers. When the device receives its notification, it downloads the updated policy from the server.

The client enforces Workspace and device policies at all times.

> **Note:** When a new Password policy is pushed to the device by the server, the Android MDM agent only sets the policy parameter if the new value is different than the older value. This ensures that counters for parameters like "Password History Length," "Password Expiration Timeout," and "Maximum Failed Passwords For Wipe" are not unintentionally reset.

To enforce *compliance policies*, the client runs rule checks intermittently. For managed devices, compliance rule checks run (1) during the enrollment process, (2) whenever the device **Sync** command is issued from the MSM server, and (3) every night when the `ComplianceCheckTrigger` scheduled job runs on the device and evaluates all enrolled devices for policy compliance. For unmanaged devices, compliance rule checks happen (1) during the enrollment process, (2) whenever a policy is updated, and (3) every night when the `ComplianceCheckTrigger` scheduled job runs on the device and evaluates all enrolled devices for policy compliance.

System Administrators can specify the security actions that should be carried out when compliance violations occur. When configuring these actions, remember that the *Wipe* action differs for unmanaged and managed devices:

- On unmanaged devices the *Wipe* action only wipes data from the Workspace and apps associated with the Workspace.
- On managed devices the *Wipe* action de-registers the device and removes apps, certificates, restrictions, and other content that was provisioned by the Mobile Security Manager. Following this action the device is no longer controlled by the server.

Compliance rules are configured on the Enrollment/Compliance page in the course of creating a Mobile Security Policy. For details, see Section 4.4, "Managing Devices and Workspaces With Policies."

8.5 Understanding How the System Resolves Policy Conflicts

Mobile Security Manager has a system that resolves overlapping or conflicting policies. Policy merging and conflict resolution are necessary because multiple policies can be assigned to a role, and multiple roles can be assigned to a user. While System Administrators can take steps to reduce the number of policy conflicts (see Section 8.6, "Creating Manageable Policies"), conflicts can still occur. The preliminary policies that users inherit through role assignments are called the *applicable policies*, and the merged policy that the system enforces is called the *Effective Policy*. Mobile Security Manager calculates the Effective Policy for every user enrolled in the mobility program twice daily, at 11 PM when the Compliance Check Trigger job runs, and at 10 PM when the Device Sync Trigger job runs. See Section 11.1, "Understanding Scheduled Jobs" for information about these jobs.

If security-related policies are in conflict, the system enforces the most restrictive policy. For example, policy attributes such as *Account Lockout Action* and *Authentication Frequency*, use precedence rules to determine the most restrictive policy. In the case of *Account Lockout Action*, "Wipe" takes precedence over "Lock" takes precedence over "Do Nothing"; whereas with *Authentication Frequency*, "Always" takes precedence over "Idle Timeout" takes precedence over "Session."

If access-related policies are in conflict, the system uses one of the following merge rules: MAX, MIN, AND, OR, or UNION. For example, for *Idle Timeout Period* (merge rule: MIN) the system will enforce the lowest (minimum) value in the Effective Policy. For *Print* (merge rule: AND) the result is YES (the user is allowed to print) only if all values in the aggregate policy are YES. For *PIN Expiry* (merge rule: OR), the result is YES (the user credential should expire after a set number of days) if any of the values in the aggregate policy are YES.

> **Note:** *File Sharing* and *Copy/Paste* are restriction policies that use the OR conflict resolution rule. File sharing is restricted if any of the values in the *File Sharing* aggregate policy are YES, and copy/paste is restricted if any of the values in the *Copy/Paste* aggregate policy are YES.

String attributes such as URLs cannot be merged, so the system selects the first instance it finds using the FIRST_OCCURANCE merge rule. This rule only applies to Application Settings policies, and only if you specify a server URL for the File Manager app, an e-mail server URL for the PIM app, or both. To avoid unexpected results, take special care to avoid overlapping Application Settings policies with different URL values.

The following cases result in a UNION in which all values are added to the Effective Policy:

- In the case of multiple Device Configurations, all configurations (e-mail, calendar, Wi-Fi, VPN) are added to the Effective Policy.
- In the case of assigned apps, all app assignments are added to the Effective Policy.

Table 8–1 Policy Merge Rules

Merge Rule	Description
MAX	System will use the highest (maximum) value in the aggregate policy
MIN	System will use the lowest (minimum) value in the aggregate policy
AND	The result is YES only if all values are YES
OR	The result is YES if any of the values are YES
UNION	All values are added into a list
FIRST_OCCURENCE	String attribute that cannot be merged. The system will use the first occurrence found
MOST_RESTRICTIVE_AUTH_FREQUENCY	“Always” takes precedence over “Idle Timeout” takes precedence over “Session"
MOST_RESTRICTIVE_COMP_ACTION	“Wipe” takes precedence over “Lock” takes precedence over “Do Nothing"
SHARED_WORKSPACE_MERGE	“Multi User” takes precedence over “SingleUser”
MOST_RESTRICTIVE_PIM_CONFIG_TYPE	“Auto” takes precedence over “Basic” takes precedence over “Manual"

Table 8–2 Merge Rules for Workspace Policy Attributes

UI Display Element	JSON Attribute Name	Merge Rule
Configuration Type	config-type	MOST_RESTRICTIVE_PIM_CONFIG_TYPE
Shared Workspace Mode	shared-workspace-mode	SHARED_WORKSPACE_MERGE
Account Lockout Action	auth-failure	MOST_RESTRICTIVE_COMP_ACTION
Enrollment Page: Compromised Device Action	compromised	MOST_RESTRICTIVE_COMP_ACTION
Authentication Frequency	online-access	MOST_RESTRICTIVE_AUTH_FREQUENCY
Offline Access	offline-access	AND
Idle Timeout Period	idle-timeout-min	MIN
Enrollment Page: Inactivity Duration	inactivity-duration-days	MIN
Enrollment Page: Inactivity Action	inactivity-duration-action	MOST_RESTRICTIVE_COMP_ACTION
Account Lockout Threshold	pin-retries	MIN
Account Lockout Action	pin-retries-failure	MOST_RESTRICTIVE_COMP_ACTION

Table 8–2 (Cont.) Merge Rules for Workspace Policy Attributes

UI Display Element	JSON Attribute Name	Merge Rule
Authentication Only	auth_only	OR
Location Settings	enable-location-service	OR
E-Mail	allow-email	AND
Instant Messaging	allow-sms	AND
Video Chat	allow-videochat	AND
Social Share	allow-socialshare	AND
Print	allow-print	AND
File Sharing	restrict-openin	OR
Copy/Paste	restrict-copypaste	OR
Save to Media Gallery	allow-save-media-gallery	AND
Save to Local Contacts	allow-save-localcontact	AND
Redirects From Workspace	allow-redirects-from-container	AND
Redirects To Workspace	allow-redirects-to-container	AND
PIN Expiry	n/a	OR
Browser: Address Bar Enabled	addressbar-enabled	AND
Browser: Download Bar Enabled	download-enabled	AND
n/a	share-enabled	AND
Doc Editing Allow	Document allow	AND
File Manager: Allow	Mfm allow	AND
File Manager: Download Allowed	allow-download	AND
File Manager: Server-based URL	server-baseurl	FIRST_OCCURANCE
PIM Allow	Pim allow	AND
n/a	allow-save-attachment	AND
Basic ActiveSync Authentication	enable-basic-auth	AND
PIM E-mail Server URL	email-server-url	FIRST_OCCURANCE
PIN History	password-history	MAX
PIN Expiry Duration	password-max-age-day	MIN
PIN Minimum Length	password-min-length	MAX
PIN Complexity	password-complexity	OR
Geo-fence	geo-fence	UNION
Limit Access	time-fence	UNION
PIN Complexity Min Checks	password-should-pass-minimum-checks	MAX

Table 8–3 Merge Rules for Enrollment Policy Attributes

UI Display Element	JSON Attribute Name	Merge Rule
Enrollment Screen: Maximum Number of Devices per User	MaxDevice Allowed	MIN
Enrollment Screen: Allow Client Specific Builds	allowSpecificClientBuild	AND
Enrollment Screen: Allow Client Builds	clientbuild	UNION

Table 8–4 Merge Rules for Device Settings and Configuration Policy Attributes

UI Display Element	Settings	Merge Rule
Device Encryption	deviceEncryptionEnabled	OR
Camera	allowCamera	AND
App Installation	allowAppInstallation	AND
No UI Element	allowAppRemoval	AND
Assistant	allowAssistant	AND
Assistance While Device Locked	allowAssistantWhileLocked	AND
Cloud Backup	allowCloudBackup	AND
Cloud Document Sync	allowCloudDocumentSync	AND
Cloud Keychain sync	allowCloudKeychainSync	AND
Diagnostic Submission	allowDiagnosticSubmission	AND
Explicit Content	allowExplicitContent	AND
Fingerprint for Unlock	allowFingerprintForUnLock	AND
Lockscreen Control Center	allowLockScreenControlCenter	AND
Lockscreen Notifications View	allowLockScreenNotificationsView	AND
Lockscreen Today view	allowLockScreenTodayView	AND
n/a	allowOpenFromManagedToUnmanaged	AND
n/a	allowOpenFromUnmanagedToManaged	AND
OTAPKI Updates	allowOTAPKIUpdates	AND
Passbook While Locked	allowPassbookWhileLocked	AND
Photo Stream	allowPhotoStream	AND
Safari	allowSafari	AND
Screenshot	allowScreenShot	AND
Shared Stream	allowSharedStream	AND
Untrusted TLS Prompt	allowUntrustedTLSPrompt	AND
YouTube	allowYouTube	AND
iTunes	allowiTunes	AND
iTunes Store Password Entry	forceITunesStorePasswordEntry	OR
AdTracking	forceLimitAdTracking	OR

Table 8–5 Merge Rules for Compliance Policy Attributes

UI Display Element	Compliance	Merge Rule
Enrollment Screen: Inactivity Duration Action	Inactivity Action	MOST_RESTRICTIVE_COMP_ACTION
Enrollment Screen: Blacklisted Apps Installed	Blacklisted Apps	MOST_RESTRICTIVE_COMP_ACTION
Enrollment Screen: Device Criteria Violation	Enrollment Criteria Violation	MOST_RESTRICTIVE_COMP_ACTION
Enrollment Screen: Passcode Compliance Action	Passcode Compliance Action	MOST_RESTRICTIVE_COMP_ACTION
Enrollment Screen:Device JailBroken	Compromised Device Action	MOST_RESTRICTIVE_COMP_ACTION

8.6 Creating Manageable Policies

How System Administrators approach the job of designing policies will depend in part on how many roles can be covered by one or more generic policies, and how many roles will require their own custom policies. This section describes some strategies that can be used to create policies that are easy to manage and that will adapt as your organization’s needs evolve.

The Restrictive Default Policy With Child-Role Exemptions Approach

This approach calls for a highly restrictive policy that is applied generically to all users in the organization. Additional policies are then created that grant certain niche privileges to specific child roles. Depending on your organization’s needs, this approach can be efficient in terms of the number of policies that need to be created. Administrators should assess whether this approach will entail more or less policy maintenance over time than alternate approaches. This approach may also require more initial planning than the other options.

The All-in-one Policy Approach

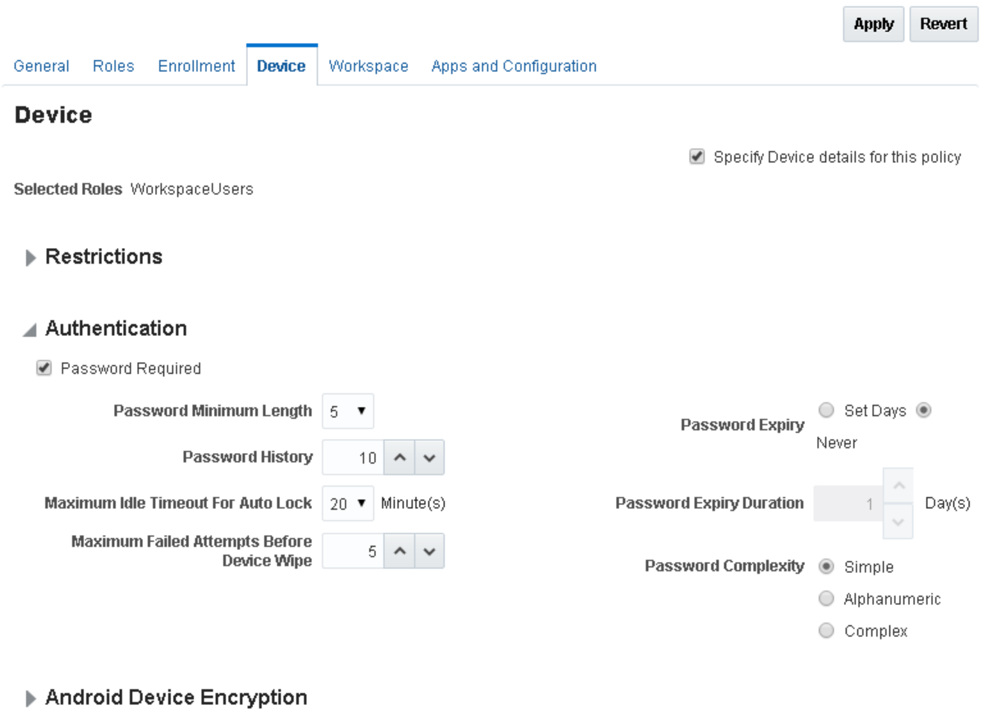
Small organizations with a limited number of roles may be able to get by with policies that specify a combination of Enrollment, Workspace, and Apps and Configuration settings all in the same policy. "All-in-one" policies can be hard to manage, however, because it is difficult to achieve predictable results when users with multiple roles have multiple overlapping policies.

The Narrowly-Defined Policy Approach

Large organizations with a large number of roles may prefer to create small, narrowly targeted policies that contain little-to-no overlap that you can then assign in various combinations to roles as needed. For example, suppose your organization has three groups: Contractors, Employees, and Vendors. Contractors and Employees share the same enrollment requirements, while Vendors have more stringent requirements. By creating two enrollment policies, one for Contractors and Employees, and the other for Vendors, you can cover your enrollment needs for three groups with two policies. Now suppose you have three departments—Support, Sales, and Engineering—and each department requires their own apps. By creating a narrowly-defined App policy for each department and assigning each policy to its department role, you have covered your app needs without having to worry that an unexpected policy overlap may cause a subset of users to inherit an unintended policy restriction.

To help you limit policies to certain tabs, the Enrollment, Device, Workspace, and Apps and Configuration tabs have a **Specify...details for this policy** check box at the top-right corner of the form. Clear the check box to deactivate the tab and prevent the tab's properties from being included in the policy, or select the check box to add the selected properties to the policy.

Figure 8–7 The Devices tab of the Mobile Security Policies page



☑ Specify Device details for this policy

Selected Roles WorkspaceUsers

☑ Password Required

Password Minimum Length 5 ▼

Password History 10

Maximum Idle Timeout For Auto Lock 20 ▼ Minute(s)

Maximum Failed Attempts Before Device Wipe 5

Password Expiry ○ Set Days ● Never

Password Expiry Duration 1 Day(s)

Password Complexity ● Simple ○ Alphanumeric ○ Complex

The Narrowly-Defined Policy Approach provides flexibility and minimizes the risk of unintended policy results, however, it can result in a large number of policies that over time may be less efficient to maintain.

8.7 How to Perform Common Mobile Security Policy Tasks

This section includes the following topics:

- How to Search for a Policy Record in Mobile Security Manager
- How to Assign a Mobile Policy to a Role
- How to Create or Edit a Mobile Security Policy
- How to Duplicate or Delete a Policy

8.7.1 How to Search for a Policy Record in Mobile Security Manager

Follow these steps to find a policy record in Mobile Security Manager.

1. Open the Mobile Security Policies page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the policy using any of the following criteria:
    - Policy name

- Description
- Role (Note: Role name search is case sensitive. If you are searching by role name, enter the whole name using the exact sequence of upper and lowercase characters.)

Policies that meet the search criteria are listed in the Search Results section of the page.

3. Click a policy record to view (expand) additional policy details; click again to hide the details.

8.7.2 How to Assign a Mobile Policy to a Role

System Administrators can add a policy to a role from the **Mobile Security Policies** tab, or from the **Mobile Roles** tab. The following procedure describe the steps starting from the Mobile Security Policies tab. To start from the Mobile Roles tab, see the steps in Section 5.4.2, "How to Assign a Mobile Policy to a Role."

1. Open the Mobile Security Policies page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."
2. Search for the policy that you want to assign to a role. To learn how, see Section 8.7.1, "How to Search for a Policy Record in Mobile Security Manager."
3. Expand the policy record by clicking it.
4. To assign a policy to a role:

   a. In the **Roles** section, click **Add**.

   A new Role Name row is added to the table.

   b. In the **Role Name** field, type the name of the role that you are adding to the policy. (You can also type part of the name and let the auto-complete feature suggests matching role names.)

   c. Click **Apply** to save your changes, or click **Revert** to return the page to the last saved version.

   To remove a policy from a role:

   a. Click to highlight the role that you are removing from the policy.

   b. Click **Remove**.

   c. Click **Apply** to save your changes, or click **Revert** to return the page to the last saved version.

8.7.3 How to Create or Edit a Mobile Security Policy

Before you start, see Section 8.6, "Creating Manageable Policies" for general policy advice. For help completing the form fields, refer to online help or see the *Help Reference for Oracle Mobile Security Suite Consoles*.

> **Note:** When editing a configuration tab, if you clear a **Specify...policy** check box and save your policy changes, the configuration settings for that tab are permanently deleted from Mobile Security Manager.

1. Open the Mobile Security Policies page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."

2. If you are *creating* a new policy, click **Add**.

   The Create Policy page opens.

   If you are *editing* a policy, search for the policy, then click the policy record to expand the policy details section.

3. On the General Information page, type a name for the policy and click **Next**.

4. On the Roles page, type the name of a role to add to the policy. (You can also type part of the name and let the auto-complete feature suggests matching role names.)

   If you don't want the policy to apply to a child role you can exclude it. Expand the **Excluded Child Roles** section, click **Add** to add a row to the table, type the name of the role to exclude, and click **Next**.

5. *The Enrollment/Compliance page opens*.

   This policy page lets you verify that a device is policy compliant when it enrolls with Mobile Security Manager, as well as on an ongoing basis following enrollment. Use the Enrollment/Compliance policy page to specify enrollment requirements, enrollment details, and/or compliance rules.

   To skip creating an Enrollment/Compliance policy, click **Next** to go to the **Device** policy page. Otherwise, select **Specify Enrollment/Compliance details for this policy** to activate the form. See the *Help Reference for Oracle Mobile Security Suite Consoles* for help completing the Enrollment/Compliance form, then click **Next**.

6. *The Device page opens*.

   Use the Device policy page to specify an MDM security policy. Complete the page if you plan to add this policy to a role that you will assign only to MDM+MAM users.

   To skip creating a Device policy and instead create a policy that you can assign to MAM-only users, click Next to go to the **Workspace** form. Otherwise, select **Specify Device details for this policy** to activate the Device policy form. See the *Help Reference for Oracle Mobile Security Suite Consoles* for help completing the form, then click **Next**.

7. *The Workspace page opens*.

   Use the Workspace policy page to specify a MAM security policy, including setting Workspace authentication details, allowing (or restricting) Workspace privileges, enabling and configuring application settings, and setting Time Access and Geo Access settings. PIN settings only apply for PKI authentication and clients configured for PKINIT authentication.

   To skip creating a Workspace policy, click **Next** to go to the **Apps and Configuration** page. Otherwise, select **Specify Workspace details for this policy** to activate the Workspace policy form. See the *Help Reference for Oracle Mobile Security Suite Consoles* for help completing the form, then click **Next**.

8. *The Apps and Configuration page opens.*

   Use the Apps and Configuration policy page to specify policy-based app assignments and device configurations. A user can only install an app from the Mobile Application Catalog if you include the app in a policy assigned to the user's role.

Select **Specify Apps and Configuration details for this policy** to activate the form. See the *Help Reference for Oracle Mobile Security Suite Consoles* for help completing the form, then click **Finish**.

8.7.4 How to Duplicate or Delete a Policy

1. Open the Mobile Security Policies page in the Mobile Security Manager console. To learn how, see Section 2.2.2, "Opening the Mobile Security Manager Console Pages."

2. Search for the policy that you want to duplicate or delete. To learn how, see Section 8.7.1, "How to Search for a Policy Record in Mobile Security Manager."

3. Click ☰ (Actions) on the right side of the policy record, then choose ☰ **Duplicate** or ✖ **Remove** from the pop-up menu.

Part III

Preparing Apps for use With Oracle Mobile Security Suite

This part contains documentation that describes how to customize and brand Oracle Workspace apps and use the containerization tool.

This part contains the following chapters:

- Chapter 9, "Using the Oracle Mobile Security Suite Application Containerization Tool"
- Chapter 10, "Customizing the Oracle Secure Workspace App"

9

Using the Oracle Mobile Security Suite Application Containerization Tool

Oracle Mobile Security Suite enhances employee productivity by allowing secure access to corporate apps and data from mobile devices while preserving rich user experience. Its Mobile Security Container creates the enterprise Workspace on any iOS or Android device, corporate-owned or personal. Employees get seamless access to corporate data and apps with enterprise-grade security and deep integration with Windows Authentication for true single sign-on.

The Oracle Mobile Security Container is an important feature of Oracle Mobile Security Suite. The container isolates your enterprise access on personal devices, enabling corporate Bring Your Own Device (BYOD). This chapter explains the tool that containerizes apps so that they can be deployed in the Secure Workspace.

This document contains the following sections:

- Containerization Tool Features
- Containerization Process
- Preparing an App for Containerization
- Running the Containerization Tool
- App Signing Requirements
- Logging
- Dev Mode for iOS

9.1 Containerization Tool Features

With Mobile Security App Containerization Tool, customers can add a standardized security layer to native mobile apps. The containerization process is simple and injects the following security services into your app.

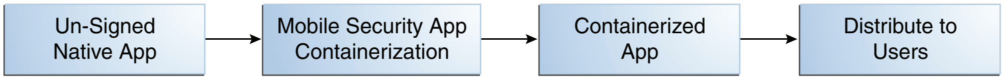
- Secure data transport: An encrypted AppTunnel through Mobile Security Access Server to application back-end resources behind an enterprise firewall.
- Authentication: Windows Integrated Authentication/SSO (Kerberos or NTLM) to application back end servers.
- Secure data storage: Encrypted storage of app data, including files, database, app cache and user preferences.
- Data leakage controls: The ability to restrict file sharing and copy paste to only other trusted apps. This enables you to control the sharing of data, including e-mail, messaging, printing and saving.

- Dynamic policy engine: More than 50 detailed app controls, including authentication frequency, geo and time fencing as well as remote lock and wipe.

Customers use the Mobile Security App Containerization Tool to add enterprise security services to apps including advanced features such as multi-factor authentication and Windows integrated Authentication (Kerberos or NTLM).

9.2 Containerization Process

The containerization process is simple and developers do not need to change a line of code. The first step is to obtain an unsigned version of the app. The unsigned version is a fully compiled app that is intended to be signed with an enterprise distribution certificate for distribution as an internal app to company employees through an enterprise appstore. The unsigned app can be an Android `apk` file or for iOS either `ipa` or static library.



Administrators then import the unsigned app into the Mobile Security App Containerization Tool, which inspects target apps and then injects containerization code and signs with the target enterprise distribution certificate. The result of this process is a fully signed app ready for enterprise distribution that can be uploaded to the Mobile Security Access Server or any other Enterprise AppStore.

The injected code looks for specified system calls to the target OS and injects Mobile Security frameworks in between to handle security tasks such as data encryption, networking and authentication. The beauty of this process is that developers don't have to change code. Rather, they make the same app that is on public appstores available in a different distribution package for enterprise customers.

A containerized app does not require its own login screen. Once the app is launched, it is redirected to the Mobile Security Container, which performs Single Sign-On and hands the session back to the app. Also, the app does not require VPN to connect to an internal website or services. Instead a secure AppTunnel is established between the app and Mobile Security Access Server, which provides secure transport for accessing internal sites and services.

9.3 Preparing an App for Containerization

To prepare an app for containerization, proceed as described in the following subsections:

- Preparing an iOS App for Containerization
- Preparing an Android App for Containerization

9.3.1 Preparing an iOS App for Containerization

In preparation for containerization, an iOS app for containerization must either be unsigned or signed by the same cert that will be used for signing after the app is containerized. A signed iOS app cannot be signed again with a different cert.

To create an unsigned iOS app follow these instructions:

1. Open `BitzerSecureContainer.xcodeproj` in Xcode

2. In Xcode 6.2, go to Build Settings > Code Signing and set **Provisioning Profile** to **Automatic** and **Code Signing Identity** to **Don't Code Sign**.

   If using a version of Xcode older than 6.2, Go to Build Settings and set **Provisioning Profile** to **None** and **Code Signing Identity** to **Don't Code Sign**.

3. Save the project and exit.
4. From the command line, go to the folder where you extracted enterprise distribution static library project.
5. Run the following command:

   ```
   xcodebuild clean build -project BitzerSecureContainer.xcodeproj -target Bitzer
   CODE_SIGN_IDENTITY="" CODE_SIGNING_REQUIRED=NO -configuration Release
   ```

This generates an unsigned application bundle with the extension `.app` under the `build/Release-iphoneos` folder. You can create an IPA for the application bundle or pass the application bundle `.app` file to the containerization tool as an input file.

To create an IPA from application bundle, follow these steps:

1. Create a folder called `Payload`.
2. Copy the app bundle (`.app` file) to the `Payload` folder.
3. Zip the `Payload` folder and rename it, changing the file extension from `.zip` to `.ipa`

9.3.2 Preparing an Android App for Containerization

In preparation for containerization, an Android app can be signed or unsigned. An already signed Android app must not be signed with a different cert.

To un-sign an existing signed app, run the following command:

1. `build-apk.sh extract` *`YourSigned.apk`*
2. `build-apk.sh package` *`YourSignedApkFolder Unsigned.apk YourSigned.apk`*

9.4 Running the Containerization Tool

The containerization tool is a command-line utility with the file name `c14n`. The tool is included in the binary distributed from Oracle eDelivery, along with the iOS and Android Workspace apps. You can run `c14n` from anywhere because the Oracle Mobile Security Suite installer sets the containerization tool system path during installation.

The `c14n` command takes a number of parameters. You use different parameters for iOS apps than for Android apps. The tool can determine which type of app you are using as input.

Most of the parameters can be omitted, either because they are optional or because their value can be passed through environment variable instead of on the command line. This makes it easier to containerize multiple apps without having to type common parameters every time.

9.4.1 Syntax for c14n

The containerization tool command `c14n` has following syntax:

```
c14n [-c command]
     [-i input_file]
     [-o output_file]
```

```
     [-conf conf_file]
     [-cert cert_name -p provisioning_profile_file]
     [-keystore keystore_file -storepass keystore_password -storealias keystore_
alias]
     [-x custom_parameters]
     [-xc]
     [-v ]
     [-version ]
     [-log {log_file | off} ]
```

9.4.2 Parameters and Their Options for c14n

-c *command*

The command to pass to containerization tool. The command determines what action will be performed on the input IPA. Command can be one of following:

- `inject`: Injects and signs an unsigned app in a single step. The output app is ready for enterprise distribution. This is the default value if **-c** is not specified.
- `injectonly`: Injects an unsigned app without signing it. The output app is unsigned and will need to be signed using `signonly` option.

> **Note:** For iOS, codesign should not be directly used on an uninjected IPA.

- `signonly`: Sign an injected or uninjected app. The app must be unsigned or previously signed by the same certificate.
- `info`: Display information about an injected or an uninjected app.

-i *input_file*

The input file to run containerization tool on. It can be an iOS IPA (`.ipa`), iOS app bundle (`.app`), iOS Xcode Archive (`.xcarhive`) or Android APK (`.apk`). You can specify *`input_file`* as the full path to the file or just as the filename. If you specify just the filename, the command looks for the input file in the directory that the containerization tool is being run from.

You can also specify the input file by setting the environment variable *`C14N_INPUT_FILE`*.

-o *output_file*

The output app file generated by the command. If no path is specified then the output file is generated in the same location as input file. The output file is has the extension `.ipa` for iOS and `.apk` for Android.

If you do not specify an output file, the command generates a file with the default name *`input_file`*`-c14n-`*`command-MMDDYYYY`*`.ipa` or *`input_file`*`-c14n-`*`command-MMDDYYYY`*`.apk` in the same folder as the input file.

You can also specify the output file by setting the environment variable *`C14N_OUTPUT_FILE`*.

-conf *conf_file*

The conf file to use for iOS. This parameter is optional. If you do not specify this parameter, the default configuration file is used.

You can also specify the conf file by setting the environment variable *`C14N_CONF_FILE`*.

-cert *cert_name*

The certificate to use for code signing an iOS app. It must be a valid iOS development or distribution certificate present in Keychain.

You can also specify the certificate by setting the environment variable *`C14N_CERT_NAME`*.

This parameter is only needed for the containerization of iOS apps.

-p *provisioning_profile_file*

The provisioning profile file to use for an iOS app. You can specify the path to the file or just the profile name. If you specify just the profile name, the containerization tool looks for it in the folder that the containerization tool is being run from.

You can also specify the provisioning profile by setting the environment variable *`C14N_PROVISIONING_PROFILE`*.

This parameter is only needed for the containerization of iOS apps.

-keystore *keystore_file*

The path of the keystore file to use for signing an Android APK.

If you do not specify the keystore file, the command uses the default keystore. You can only use the default keystore for testing. To do so, you must also have the workspace app signed by the default keystore.

You can also specify the keystore file by setting the environment variable *`C14N_KEYSTORE_FILE`*.

This parameter is only needed for the containerization of Android apps.

-storepass *password*

The password for the keystore file that you specified with `-keystore`.

You can also specify the keystore password by setting the environment variable *`C14N_KEYSTORE_PASSWORD`*.

This parameter is only needed for the containerization of Android apps.

-storealias *alias*

An alias for the keystore entry to use for the keystore specified with `-keystore`.

You can also specify the keystore alias by setting the environment variable *`C14N_KEYSTORE_ALIAS`*.

This parameter is only needed for the containerization of Android apps.

-x *custom_parameters*

Custom parameters to pass to the containerization tool.

You can also specify the custom parameters by setting the environment variable *`C14N_CUSTOM_PARAMETERS`*.

This parameter is only needed for the containerization Android apps.

-xc

Disables validation that the input iOS app is signed by same cert as the one specified by `-cert`. This validation is enabled by default.

This parameter is only needed for the containerization iOS apps.

-v

Verbose mode. If not specified, the containerization tool is run in silent mode.

-version

Prints the version of containerization tool

-log *log_file* | off

Generates log file with the path specified by `log_file`, or disables log file generation if off is specified.

By default, a log file is generated under `/opt/BitzerC14N directory` each time the containerization tool is run. By default, the log file name is *`output_ipa`*`-c14n-`*`command-MMDDYYYY`*`.log`.

The following table gives snapshot of the parameters required for iOS and Android

Table 9–1 Containerization Tool (c14n) Parameters

Parameter	iOS	Android	Required	Environment Variable	Default
`-c`	Yes	Yes	No		`inject`
`-i`	Yes	Yes	Yes	`C14N_INPUT_FILE`	
`-o`	Yes	Yes	No	`C14N_OUTPUT_FILE`	*`input_file`*`-c14n-`*`command-MMDDYYYY`*`.ipa` or `.apk`
`-conf`	Yes	No	No	`C14N_CONF_FILE`	Set by `c14n`
`-cert`	Yes	No	Yes	`C14N_CERT_NAME`	
`-p`	Yes	No	Yes	`C14N_PROVISIONING_PROFILE`	
`-keystore`	No	Yes	Yes	`C14N_KEYSTORE_FILE`	Default provided by `c14n`
`-storepass`	No	Yes	Yes	`C14N_KEYSTORE_PASSWORD`	Default provided by `c14n`
`-storealias`	No	Yes	Yes	`C14N_KEYSTORE_ALIAS`	Default provided by `c14n` if `-keystore` is not specified
`-x`	No	Yes	No	`C14N_CUSTOM_PARAMETERS`	
`-xc`	Yes	No	No		
`-v`	Yes	Yes	No		
`-version`	Yes	Yes	No		
`-log`	Yes	Yes	No		*`output_ipa`*`-c14n-`*`command-MMDDYYYY`*`.log`

9.4.3 Examples for c14n

Example 1 Inject and Sign an Unsigned iOS App

```
c14n -c inject -i Candidate.ipa -o injected.ipa -conf c14n.conf -cert
'iPhone Distribution: Acme Corp Inc.' -p dist.mobileprovision -v
```

Example 2 Inject Unsigned iOS app Only and Run in Silent Mode

```
c14n -c injectonly -i Candidate.ipa -o injected.ipa -conf c14n.conf
```

Example 3 Sign an iOS App Only Without Injection

```
c14n -c signonly -i ./Candidate.ipa -o ./injected.ipa -cert 'iPhone
Distribution: Acme Corp Inc.' -p dist.mobileprovision -v
```

Example 4 Inject and Sign an Android apk

```
c14n -c inject -i candidate.apk -o containerized.apk -keystore
prod-key.keystore -storepass mypass -storealias mykey -v
```

Example 5 Inject Android apk Only and Run in Silent Mode

```
c14n -c injectonly -i candidate.apk -o containerized.apk
```

Example 6 Sign Android apk Only Without Injection

```
c14n -c signonly -i candidate.apk -o containerized.apk -keystore
prod-key.keystore -storepass mypass -storealias mykey -v
```

9.5 App Signing Requirements

The following requirements apply when signing apps. A containerized app will not work with the Workspace app if these conditions are not met.

9.5.1 App Signing Requirements for iOS

Workspace app and containerized apps must be signed using provisioning profiles that have the same App ID Prefix. Containerized apps will not work with Workspace if they have a different App ID Prefix.

9.5.2 App Signing Requirements for Android

Workspace app and containerized apps must be signed using the same certificate. Containerized apps will not work with Workspace if they are signed with a different certificate.

For more information on signing Android apps, see "Signing Your Applications" at `http://developer.android.com`.

9.6 Logging

Containerization tool creates a log file every time an app is containerized. By default log files are generated with name `*output ipa*-c14n-*command-MMDDYYYY*.log`. The file name and location can be specified using `-log` *`log_file parameter`*. Logging can also be disabled using `-log off` *`parameter`*.

9.7 Dev Mode for iOS

Dev mode enables developers to containerize and run an iOS app on a device by using Xcode. Dev mode provides developers the following benefits:

- They can leverage single sign-on and AppTunnel for the app during development.
- They can debug any issues with the containerized app through Xcode.
- They can switch file system encryption or SQLite encryption on or off during development to test and debug the application.

Pre-requisites for running dev mode are:

- iOS Device must be selected as the destination. Dev mode is not supported on Simulator
- A specific Code Signing Identity must be selected in build settings. Dev mode does not work with automatic selection for Code Signing Identity.
- A Workspace app signed by the same Code Signing Identity as the one selected in build settings must already be installed on the device.

To enable Dev Mode perform the following steps in your Xcode project.

1. Open your Xcode project and edit the scheme (**Product > Scheme > Edit Scheme...**).
2. In the Edit Scheme window, expand **Build steps** and go to **Pre-actions**.
3. Add a new Run Script.
4. In the Run Script window, set the following field values and click **OK**:
   - **Shell**: `/bin/bash`
   - **Provide build settings from**: Select your target from the list
   - **Script**: `/opt/BitzerC14N/bin/c14n -c devmode_clean -v`
5. In the Edit Scheme window, expand **Build step** and go to **Post-actions**.
6. Add a new Run Script.
7. In the Run Script window, set the following field values and click **OK**:
   - **Shell**: `/bin/bash`
   - **Provide build settings from**: Select your target from drop down.
   - **Script**: `/opt/BitzerC14N/bin/c14n -c devmode -v`

   At the top of the Edit Scheme window, select the Scheme to set iOS device as the destination.

Verify that **iOS Device** is selected as the destination in Xcode and then run the app. Xcode will containerize the app before deploying and running it on the device. Once the app runs on the device, you can attach it to the debugger and take full advantage of Xcode debugging features for the containerized app.

After the containerized app is run for the first time, the following two Boolean keys are added to the app's Info plist:

- C14NEncryptFilesystem
- C14NEncryptSQLite

Setting the Boolean to YES or NO will allow you to enable or disable encryption to test your app.

When running an app in Dev mode through Xcode, the containerization tool generates a log file under `/opt/BitzerC14N/logs` folder with the name `c14n-devmode-`*`MM-DD-YYYY`*. You can monitor this file for any errors generated during containerization.

10

Customizing the Oracle Secure Workspace App

Download the Oracle Secure Workspace app from eDelivery and unzip the package to a local directory. Patches are published on ARU. This app is *not* available from the Apple or Google app stores.

This chapter contains the following sections:

- Oracle Secure Workspace Customization for iOS
- Oracle Secure Workspace Customization for Android

10.1 Oracle Secure Workspace Customization for iOS

To customize and brand the Oracle Workspace app for your company, use the Enterprise Distribution static library framework Xcode project, which is part of Oracle Mobile Security Suite. The project file name is `SecureWorkspace.FIPS.zip` or `SecureWorkspace.NONFIPS.zip`.

You can perform the following customizations on the Oracle Workspace app:

- Bundle identifier
- App name
- App icon
- Company logo
- EULA text file
- Custom config URLs for workspace app
- Remove support for various document types
- Enable Apple Data Protection

This section contains the following topics:

- Change Bundle Identifier
- Change App Name
- Change App Icon, Company Logo and Default Splash Screen
- Create a EULA File
- Customize Config URLs
- Customize Password Management

- Enable Apple Data Protection
- Remove Document Types

10.1.1 Change Bundle Identifier

The bundle identifier (bundle ID) needs to match your provisioning profile. To change the bundle identifier follow these steps:

> **Note:** To learn about bundle IDs, see "Configuring Your Xcode Project for Distribution," in the *App Distribution Guide*, which is available from the iOS Developer Library:
>
> ```
> https://developer.apple.com/library/ios/documentation/IDEs/Conceptu
> al/AppDistributionGuide/ConfiguringYourApp/ConfiguringYourApp.html
> ```

1. Open `BitzerSecureContainer.xcodeproj` in Xcode.
2. Choose **BitzerSecureContainer > Targets > General > Bundle Identifier**.

   In versions of Xcode older than Xcode version 6.2, choose **Build Settings > General**.
3. Under the Identity section, change the domain part of **Bundle Identifier** to the domain you want to use.

   The default bundle identifier value is: `com.oracle.OracleSecureWorkspace`.

   Change this to: *`com.example`*`.OracleSecureWorkspace`, where `com.example` is your domain.

10.1.2 Change App Name

The default name of the app is `Workspace`. This is what you will see on iOS springboard. This section describes how to change the default name.

Note: The following steps require the Plistbuddy application to be installed on your Mac.

For English Language Devices

To change the app name on devices that use the English (en) language setting, do the following:

1. Open the folder where you unzipped the `BitzerSecureContainer` and `BitzerSecureContainer.xcodeproj` folders.
2. Type the following command:

   ```
   /usr/libexec/PlistBuddy -c "Set :CFBundleDisplayName YourNewAppName"
   BitzerSecureContainer/BitzerSecureContainer.embeddedframework/BitzerSecureConta
   iner.framework/Resources/en.lproj/InfoPlist.strings
   ```

For All Other Languages

To change the app name for other languages, do the following:

1. Replace the `en.lproj` subfolder with the appropriate language subfolder. For example, for French use `fr.lproj`, for German use `de.lproj`, for Spanish use `es.lproj`, and so on.

2. Type the following command to see all of the language files:

```
ls
BitzerSecureContainer/BitzerSecureContainer.embeddedframework/BitzerSecureConta
iner.framework/Resources/*.lproj/InfoPlist.strings
```

3. To change the name in all of the language files, type this command:

```
for d in `ls
BitzerSecureContainer/BitzerSecureContainer.embeddedframework/BitzerSecureConta
iner.framework/Resources/*.lproj/InfoPlist.strings`; do
/usr/libexec/PlistBuddy -c "Set :CFBundleDisplayName YourNewAppName" $d ; done
```

10.1.3 Change App Icon, Company Logo and Default Splash Screen

To change the app icon, the company logo, and the default splash screen, replace the icons under the folder:

`BitzerSecureContainer/BitzerSecureContainer.embeddedframework/Resources`

The icons must have following dimensions:

Table 10–1 Icon Dimensions

Image Name	Dimension	Description
Icon.png	57 x 57	App Icon
Icon@2x.png	114 x 114	App Icon
Icon-72.png	72 x 72	App Icon
Icon-144.png	144 x 144	App Icon
Icon-Small.png	29 x 29	App Icon
Icon-Small@2x.png	58 x 58	App Icon
Icon-Small-50.png	50 x 50	App Icon
Default.png	320 x 480	iPhone splash screen
Default@2x.png	640 x 960	iPhone splash screen for retina
Default-568h@2x.png	640 x 1136	iPhone 5 splash screen for retina
Default-Portrait~ipad.png	768 x 1004	iPad portrait splash screen
Default-Portriat@x~ipad.png	1536 x 2008	iPad portrait splash screen for retina
Default-Landscape~ipad.png	1024 x 748	iPad landscape splash screen
Default-Landscape@2x~ipad.png	2048 x 1496	iPad landscape splash screen for retina
company-logo.png	200 x 55	Company logo image for iPhone
company-logo@2x.png	400 x 110	Company logo image for iPhone with retina
company-logo~ipad.png	600 x 165	Company logo image for iPad
company-logo@2x~ipad.png	1200 x 330	Company logo image for iPad with retina

10.1.4 Create a EULA File

This section describes how to create and display the end-user license agreement (EULA) on a device.

> **Note:** Because the EULA applies to the device and not the end-user, it is only displayed once per device the first time a user logs in to the Secure Workspace. This holds true even if Multi-User mode (Shared Workspace mode) is enabled on the device.

To create a file with the end-user license agreement content, name it `EULA.txt` and add it to this location:

```
BitzerSecureContainer/BitzerSecureContainer.embeddedframework/BitzerSecureContainer.framework/Resou
rces/en.lproj/EULA.txt
```

In addition, the `SHOW_EULA_SCREEN` value should be set to true in the `CompanySettings.plist` file:

```
BitzerSecureContainer/BitzerSecureContainer.embeddedframework/BitzerSecureContainer.framework/Resou
rces/Targets/Bitzer/CompanySettings.plist file
```

```
        <key>SHOW_EULA_SCREEN</key>
        <true/>
```

10.1.5 Customize Config URLs

> **Note:** By default, the Secure Workspace app is not configured to connect to any specific MSAS server. The Secure Workspace app can be customized to automatically configure itself to a single MSAS server or present a list of available MSAS servers for the user to select from. Please contact your Oracle Mobile Security Suite server team for Config URL details.

The URLs and help text on the Workspace config screen are customizable. This feature makes it possible to link to custom help files that you develop and host. Follow these steps to customize the URLs and help text on the config screen:

1. Open `BitzerSecureWorkspace.xcodeproj` in Xcode.
2. Open the file: `BitzerSecureContainer/BitzerSecureContainer.embeddedframework/Resources/CustomizableSettings.plist`
3. The Config URL settings are under the **Root** node in an Array type item called **`CONFIG_URL_SETTINGS`**. Each item in this array represents a Config URL that will be available for selection on the Config screen. If no items are present for **`CONFIG_URL_SETTINGS`** array, a default demo URL and help text is displayed. If one or more items are present under the **`CONFIG_URL_SETTINGS`** array, the user is presented with custom URL selection options.
4. Modify **`Item 0`** under **`CONFIG_URL_SETTINGS`** and enter your customized Config URL. The information to enter is described in the following table.

Table 10–2 CONFIG_URL_SETTINGS

Property (key)	Value	Example

`autoconfigure`	Optional setting that is used if there is only one configuration URL.
	`true` - Automatically select the configuration option and proceed to the login display. The user is not prompted for the configuration URL. This option only works if there is only one option to select. Otherwise, it behaves like the false setting.
	`false` - Prompt the user to select a configuration option from the list or enter one manually. The user is presented with the Config URL screen and a link appears under the Config URL field.
		Example 1:
		`<key>autoconfigure</key>`
		`<true/>`
		Example 2:
		`<key>autoconfigure</key>`
		`<false/>`

`label`	String value that will be displayed instead of the URL. This should be user friendly text that describes your Mobile Security Access Server site. For example, if you have more than one Mobile Security Access Server site you can label them as Houston Site, or BYOD Site.	`<key>label</key>`
		`<string>Your MSAS Server</string>`

`value`	The Mobile Security Access Server configuration URL.	`<key>value</key>`
		`<string>http://example.com/bmax/bmconfig_kinit_kinit.json</string>`

5. If more than one MSAS server is available, you can configure the Secure Workspace so that end users can pick which MSAS server to connect to.

   To add more than one Config URL, close **`Item 0`**, copy and paste it to create a duplicate item, and edit it. Multiple Config URLs are presented as an action sheet with site labels for each button.

CONFIG_URL_SETTINGS Example

A `CONFIG_URL_SETTINGS` example is given here.

```
<key>CONFIG_URL_SETTINGS</key>
<array>
    <dict>
    <key>autoconfigure</key>
    <false/>
    <key>label</key>
    <string>Your MSAS Server name alias</string>
    <key>value</key>
    <string>http://example.com/bmax/bmconfig_kinit_kinit.json</string>
    </dict>
</array>
```

10.1.6 Customize Password Management

Use this feature to add one or more password management links to the Secure Workspace login dialog, for example *Forgot Password* and *Forgot User ID*. Oracle Mobile Security Suite does not provide advanced password management functionality. Rather, this feature provides a placeholder to launch password management links from the Secure Workspace app. Using these links, the user can open a web page where they can create, reset, or recover their password. Each link is opened outside of the Secure

Workspace and does not use any of the Secure Workspace security or networking features. Be sure to use unauthenticated links that do not require a login session. Users should be able to use the URLs from Safari (the default browser) without having to authenticate.

Customize the Secure Workspace app by modifying the `CustomizableSettings.plist` file located here:

```
BitzerSecureContainer/BitzerSecureContainer.embeddedframework/Resources/CustomizableSettings.plist
```

Add the `PASSWORD_MANAGEMENT` key, which specifies the link to the password management site that should be included in the login dialog.

The following notes apply to this property:

- You can add unlimited entries, however, three entries fills the display in the Secure Workspace login screen. More than three entries is not recommended.
- Descriptions in multiple languages can be provided using standard language codes.

A `PASSWORD_MANAGEMENT` example with two entries is shown here.

```
<key>PASSWORD_MANAGEMENT</key>
<array>
    <dict>
    <key>URL</key>
    <string>http://example.com/NewUser.html </string>
    <key>de</key>
    <string>Passwort vergessen URL</string>
    <key>en</key>
    <string>Forgot Password URL</string>
    <key>fr</key>
    <string>Mot de passe oublié URL</string>
    </dict>
    <dict>
    <key>URL</key>
    <string>http://example.com/PasswordReset.html</string>
    <key>de</key>
    <string>Benutzer-ID vergessen URL</string>
    <key>en</key>
    <string>Forgot User ID URL</string>
    <key>fr</key>
    <string>ID utilisateur Mot URL</string>
    </dict>
</array>
```

10.1.7 Enable Apple Data Protection

Tip: Apple Support defines Apple Data Protection as follows:

Apple Data protection is available for devices that offer hardware encryption, including iPhone 3GS and later, all iPad models, and iPod touch (3rd generation and later). Data protection enhances the built-in hardware encryption by protecting the hardware encryption keys with your passcode. This provides an additional layer of protection for your e-mail messages attachments, and third-party applications.

For more information, refer to this Apple Support page:

`https://support.apple.com/en-us/HT202064`

To enable Apple Data Protection, follow these steps:

1. Log into the Apple iOS Dev Center under `https://developer.apple.com/` and go to Certificate, Identifiers and Profiles.
2. Go to **Identities** and select the App ID you want to use with Workspace app.
3. Edit the App ID and, select **Data Protection**, then select **Protected Unless open** under Sharing and Permissions.
4. Regenerate your provisioning profile and use that profile for signing the workspace app.

10.1.8 Remove Document Types

The Workspace app enables you to open the following documents types:

- Word: `docx` and `doc`
- Powerpoint: `pptx` and `ppt`
- Excel: `xlsx` and `xls`
- Text: `txt`, `rtf`
- Adobe `pdf`
- Images: `jpg`, `jpeg`, `png`, `tif`, `tiff`, `bmp`, `gif`
- Videos: `mov`

To remove support for any or all of these file types (that is, to prevent the Workspace app from opening them) you must remove the document types from `BitzerSecureWorkspace.xcodeproj` and rebuild the Workspace app. Follow these steps:

10.1.8.1 Removing the Document Types from BitzerSecureWorkspace.xcodeproj

1. Open `BitzerSecureContainer.xcodeproj` in Xcode.
2. Select the **Info** tab and scroll down to Document Types section.
3. Remove the document types: Bitzer MS Reader, Bitzer Document Editor, Bitzer PDF Reader and Bitzer Image Viewer. Do not remove any other documents types.

10.1.8.2 Rebuilding the Workspace

To rebuild the Workspace app after making changes, follow these steps:

1. Make sure the bundle identifier has been updated to match your provisioning profile
2. Ensure the correct provisioning profile has been selected under **Build Settings > Code Signing**.
3. Ensure the correct code signing identity has been selected under **Build Settings > Code Signing**.
4. Select **iOS Device** from **Product > Destination**. The Workspace app can only be built for **iOS Device**.
5. Select **Product** from the menu and then select **Archive**.
6. In the **Organizer - Archives** window click **Export**.
7. In the **Select the method for export** window, select **Save for Enterprise or Ad Hoc Deployment** and click **Next**.

8. Select **Provisioning Profile** and click **Choose**.
9. Select **Export** from the Summary window.
10. Save the app without selecting **Save for Enterprise Distribution**. This generates a signed IPA for the Workspace app.

Once the Workspace app `ipa` file is generated, you can upload it to your Catalog on the Mobile Security Manager console or to your enterprise app store.

10.1.8.3 Certificate and Provisioning Profile Requirements

Workspace app and containerized apps must be signed using provisioning profiles that have same App ID Prefix. Containerized apps will not work in the Workspace if they have a different App ID Prefix.

10.2 Oracle Secure Workspace Customization for Android

The Secure Workspace unsigned APK that is part of Oracle Mobile Security Suite can be used for customization and branding. You can perform the following customizations:

- App package name
- App name
- App icon
- Splash screen
- EULA text file
- Custom config URLs for workspace app
- Remove support for various document types

This section contains the following topics:

- Extracting APK for Customization
- Change App Package Name
- Change App Name
- Change App Icon
- Change Splash Screen Image
- Change MDM Agent Settings
- Create a EULA File
- Customize Config URLs
- Customize Password Management
- Sample prop.txt File
- Enable Kiosk Mode
- Remove Document Types
- Packaging Customized APK
- Signing APK
- Signing Certificate Requirements

10.2.1 Extracting APK for Customization

Before you can customize or make any changes for branding, you must extract the APK. Ensure that the Oracle Mobile Security App Containerization Tool has been installed, then run the following command to extract the APK.

```
build-apk.sh extract SecureWorkspace-unsigned-xxxx.apk
```

10.2.2 Change App Package Name

The package name serves as a unique identifier for the application. You may want to customize this value to gain additional control of the app upgrade process or for security reasons. Because the package name defines your application's identity, if you change it after it has been deployed, the new app will be considered to be a different application and users of the previous version will not be able to update to the new version. For more information see:

http://developer.android.com/guide/topics/manifest/manifest-element.html

To change the package name for a Secure Workspace app, follow these steps:

1. Go to the folder where the APK was extracted.
2. Edit file `AndroidManifest.xml`.
3. On the second line of the file, modify the attribute package to the package name you want. For example:

   ```
   <manifest android:versionCode="1" android:versionName="@string/version_name"
   package="com.acme.secureworkspace"
   xmlns:android="http://schemas.android.com/apk/res/android">
   ```

4. Save the file.

10.2.3 Change App Name

To change name of a Secure Workspace app follow these steps:

1. Go to the folder where the APK was extracted.
2. Edit the file `res/values/strings.xml`.
3. Modify the value for the string name `app_name`. For example:

   ```
   <string name="app_name">My Workspace</string>
   ```

4. Save the file.

10.2.4 Change App Icon

To change the icon of a Secure Workspace app, follow these steps:

1. Go to the folder where the APK was extracted.
2. Under the `res` folder, there are several folders with names like `drawable-xxxx`. Each of these folder contains a file, `icon.png`, that is sized at 128 x 128 px. Replace each `icon.png` file with your own icon. Make sure the resolutions match.

   The icons must go in the folders listed in Table 10–3:

Table 10–3 Icon Locations

`icon.png` (App icon)	`splashscreen.png`	Folder (under `res/`)
128x128	N/A	`drawable`
128x128	480x800	`drawable-hdpi`
128x128	800x480	`drawable-land-hdpi`
128x128	800x480	`drawable-land-ldpi`
128x128	800x480	`drawable-land-mdpi`
128x128	1024x768	`drawable-land-xhdpi`
128x128	480x800	`drawable-ldpi`
128x128	480x800	`drawable-mdpi`
128x128	768x1024	`drawable-xhdpi`
128x128	N/A	`drawable-xlarge-hdpi`
128x128	N/A	`drawable-xxhdpi`

For more information about resolution and size of images to be placed in the respective drawable folders, see "Supporting Multiple Screens" at `http://developer.android.com`

10.2.5 Change Splash Screen Image

To Change the splash screen image, follow these steps:

1. Go to folder where the APK was extracted.
2. Under the `res` folder, there are several folders with names like `drawable-`*`xxxx`*. Each of these folder contains a file, `splashscreen.png`, with a specific resolution. Replace each `splashscreen.png` file with your own icon. Make sure the resolutions match.

10.2.6 Change MDM Agent Settings

To change the MDM Agent settings, which include the display-name label value and description value, follow these steps:

1. Go to the folder where the APK was extracted.
2. Open the file `res/values/strings.xml` for editing.
3. Locate `msm_device_admin_label` and `msm_device_admin_description` and modify the values as needed. For example:

   ```
   <string name="msm_device_admin_label">My Device Administrator</string>
   <string name="msm_device_admin_description">Lets you securely manage your
   device</string>
   ```

   If translated versions of `msm_device_admin_label` and `msm_device_admin_description` exist, update them in the corresponding `strings.xml` file under `res/values-<`*`language_code`*`>-<`*`region_code`*`>/strings.xml`. If translated versions are *not* available, then the translated versions of the default values should be removed from those files.
4. Save the file.

10.2.7 Create a EULA File

This section describes how to create and display the end-user license agreement (EULA) on a device.

> **Note:** Because the EULA applies to the device and not the end-user, it is only displayed once per device the first time a user logs in to the Secure Workspace. This holds true even if Multi-User mode (Shared Workspace mode) is enabled on the device.

To create a file with the end-user license agreement content, follow these steps:

1. Go to the folder where the APK was extracted.
2. Create the file `assets/EULA-en` with the EULA text contents.
3. Create the file `assets/EULA-`*`<locale>`* with EULA text using language specific to that locale. For example, for French the file would be assets`/EULA-fr`.
4. Save the EULA files.

If the device locale does not match the locale specified in a EULA file, the device will default to the `EULA-en` version.

10.2.8 Customize Config URLs

This feature makes it possible to link to custom help files that you develop and host. Follow these steps to customize the default config URLs and help text:

1. Go to the folder where the APK was extracted.
2. Edit the file `assets/prop.txt`. See Section 10.2.10 for a sample `prop.txt` file.
3. Add config URLs under `"--- Choose config urls from List ---"`. URLs must be comma separated and each URL must be enclosed within double quotes. For example:

```
"properties":
   {
      "autoConfigure": "false",
      "configURLs":
      [
         "--- Choose config urls from List ---",
                 "https://omss1.acme.com/bmax/bmax_config.json"
                ,"https://omss2.acme.com/bmax/bmax_config.json"
      ]
   }
```

4. When a single config URL is specified, then `autoConfigure` can be used. If `autoConfigure` is set to true, you are not prompted to select a config URL. Instead, the specified URL is selected for auto configuration.

   The following table describes the `config_url` configuration properties and values:

Table 10–4 Settings and values for prop.txt

Property	Value	Example

`autoConfigure`	**`false`** - Prompt the user to select a configuration option from the list or enter one manually.
	`true` - Automatically select the configuration option and proceed to the login display. This option only works if there is only one option to select. Otherwise, it behaves like the false setting.
		Example 1:
		`"autoConfigure": "false"`
		Example 2:
		`"autoConfigure": "true"`

`config_urls`	**`url`** - URL for configuring the Secure Workspace client.
	`en` - English language description of the above URL.
	Additional descriptions in other languages can be included using standard language codes.
		`"en":"Your Config URL ",`
		`"url":"https://bmax6.example.com/bmax/bmconfig_kinit_kinit.json"`

5. Save the file `prop.txt`.

10.2.9 Customize Password Management

Use this feature to add one or more password management links to the Secure Workspace login dialog, for example *Forgot Password* and *Forgot User ID*. Oracle Mobile Security Suite does not provide advanced password management functionality. Rather, this feature provides a placeholder to launch password management links from the Secure Workspace app. Using these links, the user can open a web page where they can create, reset, or recover their password. Each link is opened outside of the Secure Workspace and does not use any of the Secure Workspace security or networking features. Be sure to use unauthenticated links that do not require a login session. Users should be able to use the URLs from the default browser without having to authenticate.

Customize the Secure Workspace app by modifying the `prop.txt` file located in the `AndroidContainer/assets` folder. See Section 10.2.10 for a sample `prop.txt` file.

The `prop.txt` file supports localization. For `password_management`, supply localized, non-English strings using standard two character language codes. Strings are displayed based on the device language setting the language identifier. If the device language is not present for the desired option, the English language string is displayed.

The following table describes the `password_management` configuration properties and values:

Table 10–5 Settings and values for prop.txt

Property	Value	Example
password_managment	**url** - URL that opens the password management web site. **en** - English language description of the above URL. Additional descriptions in other languages can be included using standard language codes. **Note** - You can add unlimited entries, however, three entries fills the display in the Secure Workspace login screen. More than three entries is not recommended.	"en":"Your Password Reset site ", "url":"https://example.com/password_reset.html "

10.2.10 Sample prop.txt File

A sample prop.txt file is shown here:

```
{
  "properties":
    {
    "autoConfigure": "false",
    "config_urls":
    [
      {
      "en":"Your KINIT Server Config",
      "fr":"Votre KINIT Server Config",
      "de":"Ihr KINIT Server Config",
      "url":”https://example.com/bmax/bmconfig_kinit_kinit.json”
      },
      {
      "en":"Your PKINIT Server Config",
      "fr":"Votre PKINIT Server Config",
      "de":"Ihr PKINIT Server Config",
      "url":”https://example.com/bmax/bmconfig_pkinit_tlp.json”
      }
    ],
    "password_managment":
    [
      {
      "en":"Your Registration URL",
      "fr":"Votre Registration URL",
      "de":"Ihr Registration URL",
      "url":"http://example.com/NewUser.html"
      },
      {
      "en":"Your Password Reset URL",
      "fr":"Votre Password Reset URL",
      "de":"Ihr Password Reset URL",
      "url":"http://example.com/PasswordReset.html"
      }
    ],
    "exitOnLogin": "false",
        }
}
```

10.2.11 Enable Kiosk Mode

When Kiosk Mode is enabled, users only see the Workspace and the apps within the Workspace. Interaction with the operating system outside the Workspace is minimized. The user cannot close the Secure Workspace app, making this mode suitable for public environments where supervision is minimal, such as lobbies, exhibit spaces, and show rooms.

1. To enable Kiosk Mode, customize the Secure Workspace app by modifying the `prop.txt` file located in the `AndroidContainer/assets` folder. See Section 10.2.10 for a sample `prop.txt` file. Modify the file as follows:

   If the `"appMode":"workspace",` line is present, change it to: `"appMode":"launcher",`

   Otherwise, after

   ```
   "exitOnLogin": "false",
   ```

   add the following line:

   ```
   "appMode":"launcher",
   ```

   This change enables a proprietary Oracle launcher, not the Google Now launcher.

2. Save your changes.

3. Create a Kiosk Mode mobile security policy.

   Edit the mobile security policy's Workspace policy and set the **Shared Workspace Mode** setting to **Multi-User**. This is a required setting that wipes away Workspace data every time a user logs out of the Workspace. Assign the policy to a role in LDAP that has users who can only access corporate data using the Kiosk mode.

10.2.12 Remove Document Types

To prevent the Workspace app from opening certain file types follow these steps:

1. Go to the folder where the APK was extracted.
2. Edit file `AndroidManifest.xml`.
3. Search for the data element in intent-filter that matches the mimeType or path pattern for the file type you want to prevent from opening in Workspace app.
4. Delete the data element.
5. Save `AndroidManifest.xml`.

10.2.13 Packaging Customized APK

To package the customized APK content into an APK, follow these steps:

1. Verify that the Oracle Mobile Security App Containerization Tool has been installed.
2. Go to the folder where the APK was extracted.
3. Run the following command:

```
build-apk.sh package <Secure_Workspace_apk_folder> <new_output_apk> <original_apk>
```

Secure_Workspace_apk_folder is where the APK was extracted using the `extract` command.

10.2.14 Signing APK

To sign an APK, use the `c14n -c signonly` command. For example:

```
c14n -c signonly -i input.apk -o output.apk -keystore release-key.keystore
-storepass password -storealias release -v
```

For more information on signing Android apps, see "Signing Your Applications" at: `http://developer.android.com`

10.2.15 Signing Certificate Requirements

Workspace app and containerized apps must be signed using the same certificate. Containerized apps will not work with Workspace if they are signed with a different certificate.

For more information on signing Android apps, see "Signing Your Applications" at `http://developer.android.com`

Part IV

Managing Oracle Mobile Security Suite Settings and Configuration

This part contains documentation to help you configure Mobile Security Manager to work in your environment.

This part contains the following chapters:

- Chapter 11, "Configuring Mobile Security Manager"
- Chapter 12, "Configuring Your Environment to Work With Mobile Security Manager"
- Chapter 13, "Troubleshooting Oracle Mobile Security Suite"

11

Configuring Mobile Security Manager

This chapter documents advanced administration topics and Mobile Security Manager configuration settings. It includes the following topics.

- Understanding Scheduled Jobs
- Configuring Mobile Security Manager Settings

11.1 Understanding Scheduled Jobs

Mobile Security Manager runs the scheduled jobs in Table 11–1, which keep the server and the mobile clients up to date. Scheduled jobs are not configurable.

Table 11–1 Scheduled jobs in Mobile Security Manager

Job Name	Description	Run Frequency
Post-Process Task Trigger	Executes the post-process tasks from the queue.	Every 5 minutes
Identity Changelog Sync Trigger	Syncs the back-end LDAP directory change log.	Every 5 minutes
User Group Membership Sync Trigger	Performs the identity Users and Groups membership check on all the registered endpoints.	Every 4 hours
Device Sync Trigger	The Device Sync Trigger job performs the following tasks: ■ Syncs device information such as device attributes and info about installed apps with the Mobile Security Manager server. ■ Evaluates MDM device policies on the Mobile Security Manager server and pushes device policies to all MDM-enrolled devices.	Every day at 10 PM
Compliance Check Trigger	Evaluates all enrolled devices for policy compliance.	Every day at 11 PM

When the Compliance Check Trigger or Device Sync Trigger jobs run, Mobile Security Manager resolves policy conflicts and calculates the Effective Policy for every user enrolled in the mobility program.

11.2 Configuring Mobile Security Manager Settings

This section includes the following topics:

- About the Mobile Security Manager Settings Page
- How to Open the Mobile Security Settings Page
- Configuring Client Settings
- Configuring Server Settings

- Configuring Identity Store Settings
- Configuring CA Settings
- Configuring User Notification Settings
- Configuring Exchange Server Settings
- Configuring Device Notification Settings
- Configuring the APNS Certificate
- Configuring the GCM Entry
- Configuring Notification Templates
- Configuring MDM Agent Settings
- Configuring Blacklisted Apps

11.2.1 About the Mobile Security Manager Settings Page

The Mobile Security Manager Settings page is organized into twelve tabs that let you configure options such as client and server settings, user notification settings, settings that affect interactions with third-party systems such as Microsoft Exchange, Apple Push Notification Service, Google Cloud Messaging, and so on.

The Mobile Security Manager Settings page is located in the **Settings** section of the Oracle Access Management console.

> **Note:** Use online help to view field-level descriptions of the Mobile Security Manager Settings page, or see "Mobile Security Manager Settings Help" in the *Help Reference for Oracle Mobile Security Suite Consoles*.

11.2.2 How to Open the Mobile Security Settings Page

Use these steps to open the Mobile Security Settings console pages in the Oracle Access Management console. You must have System Administrator privileges to view this page.

1. In a browser window, open the Oracle Access Management console using the appropriate protocol (HTTP or HTTPS). For example:

   ```
   https://hostname:port/access
   ```

   For details, see "Working with the Oracle Access Management Console" in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.

2. Log in with your System Administrator user name and password.

3. Click the main Oracle Access Management **Launch Pad** page, then click **Configuration**.

   The Configuration Launch Pad opens.

   Under **Settings**, click **View** and choose **Mobile Security Manager Settings** from the menu.

   The Mobile Security Settings page opens.

4. The Mobile Security Settings page contains the following tabs that you can click to open:

- **Client Settings** - Click to change options and configuration settings that affect the Secure Workspace.
- **Server Settings** - Click to configure properties that control how Mobile Security Manager functions at the server level
- **Identity Store Settings** - Click to configure properties that control how Mobile Security Manager interacts with the directory server.
- **CA (Certificate Authority) Settings** - Click to create PKI certificate profiles and CA connections.
- **User Notification Settings** - Click to enter your mail server settings. (Mobile Security Manager uses e-mail to send users notifications.)
- **Exchange Server Settings** - Click to configure mail server settings if your organization uses Microsoft Exchange.
- **Device Notification Settings** - Click to configure notifications that the Mobile Security Manager sends to users.
- **APNS Settings** - Click to manage and upload the required APNS certificates that are used to securely communicate with the Apple Push Notification service.
- **GCM Settings** - Click to configure the values needed to communicate with the Google Cloud Messaging service.
- **Notification Templates** - Click to manage the Invite templates that the system uses to provide notification to users.
- **MDM Agent Settings** - Click to edit Android and iOS Mobile Device Management (MDM) settings.
- **Blacklisted Apps** - Click to manage prohibited apps on the device.

11.2.3 Configuring Client Settings

For descriptions of the Client Settings form fields, see "Client Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Client Settings provide the ability to configure some aspects of the Secure Workspace behavior in an OMSS deployment.

11.2.4 Configuring Server Settings

This section includes the following topics:

- Configuring General Server Settings
- Configuring Proxy Settings
- Configuring Mobile File Manager Authentication Settings

11.2.4.1 Configuring General Server Settings

For descriptions of the Server Settings form fields, see "Server Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

11.2.4.2 Configuring Proxy Settings

For descriptions of the Proxy Settings form fields, see "Proxy Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

11.2.4.3 Configuring Mobile File Manager Authentication Settings

For descriptions of the File Manager Settings form fields, see "File Manager Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

You can configure the following Mobile File Manager authentication settings:

- Whether the File Manager server should accept HTTP Basic authentication.
- Whether the server should reject unauthenticated requests without extending an authentication offer. If HTTP Basic is enabled and the **Authentication Challenge** option is selected, then the user is asked to provide a user name and password; if the **Authentication Challenge** option is not selected, the user is not asked for a user name and password and the server rejects the unauthorized request.
- Whether the server should accept HTTP Basic authentication over a non-SSL connection.
- Whether the server should offer Kerberos or NTLM authentication to the client.

Note: The Mobile File Manager will fail to connect if a Windows file share on any of the following Windows versions is being referenced by a DNS alias instead of the native system host name:

- Windows Server 2012
- Windows Server 2008
- Windows 8
- Windows 7
- Windows Vista

To fix the issue, either access the file share using the native host name, or complete the following steps to modify the registry on the file share server:

1. Locate and click the following key in the registry of the server:

   ```
   HKEY_LOCAL_
   MACHINE\System\CurrentControlSet\Services\LanmanServer\Paramete
   rs
   ```

2. On the **Edit** menu, click **Add Value**, and then add the following registry value:

   Value name: `DisableStrictNameChecking`

   Data type: `REG_DWORD`

   Radix: `Decimal`

   Value: `1`

3. Restart the Windows Server service on the file share server.

11.2.5 Configuring Identity Store Settings

For descriptions of the Identity Store Settings form fields, see "Identity Store Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Use the Identity Store Settings configuration page to:

- Edit the configuration values that Mobile Security Manager uses to import LDAP directory records from your directory server on a scheduled basis.

- Edit the System Administrator and Help Desk Administrator LDAP group mappings.
- Choose the default action (Lock, Wipe, Do Nothing) that the system should carry out when a user account is deleted or disabled in the directory server.
- Add extra LDAP user attributes to Mobile Security Manager to facilitate mapping a user's Home drive in Mobile Security File Manager.

11.2.6 Configuring CA Settings

For descriptions of the CA Settings form fields, see "CA Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles*.

> **Note:** Choose a CA provider that uses Microsoft CA servers. Only Microsoft CA servers are supported.

Use the CA (Certificate Authority) Settings tab to create PKI certificate profiles and CA connections. These settings are used for device enrollment. For successful processing, you must trust the NDES certificate authority and the certificates issued by the MSM server and MSAS server. Use the following steps:

1. Create a certificate profile for the NDES SCEP server by configuring the NDES server as described in Section 12.3, "Configuring NDES and the Active Directory Certificate Authority."
2. The Mobile Security Manager server verifies the certificate issued by the SCEP server. Consequently, you need to import the Active Directory (AD) Certificate Manager root CA and issuing CA (if different from the root CA) into the Mobile Security Manager server’s `wlstrust.jks` file. Use the following steps to import the certificates into the MSM Server trust store:

   a. Export the AD Certificate Manager root CA and issuing CA and import them into the Mobile Security Manager server’s trust store. The Mobile Security Manager server trust store file is located here:

   ```
   <DOMAIN_HOME>/config/fmwconfig/wlstrust.jks
   ```

   b. Run the following keytool commands to import the trusted certificates into the Mobile Security Manager trust store:

   ```
   keytool -importcert -keystore wlstrust.jks -file <rootca_filename> -alias
   ndesrootca -storepass <password>

   keytool -importcert -keystore wlstrust.jks -file <issuerca_filename> -alias
   ndesissuerca -storepass <password>
   ```

   c. Restart the Mobile Security Manager server.

   You can create a new CA certificate profile for the above Active Directory (AD) NDES server.

For information about certificate revocation, see Section 12.4, "Configuring Automatic Certificate Revocation with the Active Directory Certificate Authority."

11.2.6.1 Configure CA Settings for Internal CA Server

Use the following cURL commands to configure the validity period for certificates issued by an internal certificate authority.

Note: cURL is free software that you can download from the cURL website at http://curl.haxx.se/

1. Enter the following command, which retrieves the MSM server settings and saves them to a file in JSON format:

   ```
   curl -v -H "Content-Type:application/json"
   -u <adminusername>:<adminpass> --request
   -k GET https://<msmhost>:<msmport>/msm-mgmt/systemSettings/server >
   serversetting.json
   ```

2. Modify the `serversetting.json` file and update the `scepCACertValidity` parameter with a new value.

3. Enter the following command, which modifies the MSM server settings as provided in the JSON input file:

   ```
   curl -v -H "Content-Type:application/json"
   -u <adminusername>:<adminpass> --request
   -k PUT https://<msmhost>:<msmport>/msm-mgmt/systemSettings/server
   -d serversetting.json
   ```

11.2.7 Configuring User Notification Settings

For descriptions of the User Notification Settings form fields, see "User Notification Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Use this tab to enter your mail server settings. Mobile Security Manager uses e-mail to send users notifications.

Note: When using SSL to connect to the SMTP server, import the certificate into the WebLogic keystore.

11.2.8 Configuring Exchange Server Settings

For descriptions of the Exchange Server Settings form fields, see "Exchange Server Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Use this tab to configure mail server settings if your organization uses Microsoft Exchange.

Note: Also see Section 12.5, "Configuring Microsoft Exchange (Secure Mail) to Work With Mobile Security Manager."

11.2.9 Configuring Device Notification Settings

For descriptions of the Device Notification Settings form fields, see "Device Notification Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Use this tab to configure notifications that the Mobile Security Manager sends to users.

11.2.10 Configuring the APNS Certificate

For descriptions of the APNS Certificate Settings form fields, see "Apple Push Notification Service (APNS) Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Mobile Security Manager requires an Apple MDM certificate to manage iOS devices. This certificate enables secure communication using Apple Push Notification Services (APNS). (If you are only supporting unmanaged iOS devices, Mobile Security Manager does *not* need an MDM certificate.)

Before you begin - These steps require a computer running Mac OS X.

1. Create a Certificate Signing Request (CSR) to obtain an APNS certificate from Apple.

   a. Open the Keychain Access application by opening the Finder and opening **Applications** > **Utilities** > **Keychain Access**.

   b. From the menu choose **Keychain Access** > **Certificate Assistant** > **Request a Certificate From a Certificate Authority...**

      The Certificate Assistant opens.

   c. Complete the form by providing an e-mail address and a common name, then select **Saved to Disk**.

      Click **Continue**.

      The Save-as dialog opens.

   d. Save the CSR to a convenient location.

2. Send the unsigned CSR to Oracle to obtain a Signed CSR. The unsigned CSR (generated above) should be sent to Oracle Support.

   Oracle Support will sign the CSR and send it back to you.

3. Upload the signed CSR to the Apple Push Certificates Portal.

   a. Using an Apple ID and password, sign in to the Apple Push Certificate Portal located here:

      `https://identity.apple.com/pushcert/`

      The Apple ID does not need to be associated with an Apple Developer / Enterprise Account. It can be any Apple ID.

   b. Accept the EULA and continue.

   c. Click **Create a Certificate**, then click **Browse**.

      Select the Oracle-signed CSR and click **Upload**.

      A new certificate for "Oracle" Mobile Device Management opens.

   d. Click **Download** and download the Apple signed certificate.

4. Export the APNS certificate.

   a. Double-click the downloaded file to upload it using the Keychain Access application.

   b. Expand the left arrow and verify that it contains `APSP:`*`<UUID>`* `(Apple Production Services)` and that it has an associated private key. *`UUID`* is a randomly generated number.

   c. Right-click the certificate and click **Export**.

Save the certificate in .p12 format.

Enter a password to protect the exported .p12 file. Record the password because you will need it in the next step.

5. Upload the APNS certificate to OMSS.

   a. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."

   b. Click **APNS Settings** on the menu bar. (If APNS Settings is not visible, use the arrow buttons to scroll the menu bar to the right. Or, click ▼ to view additional menu items.)

   c. Click **Add** to create a new row in the settings table.

   d. For **Certificate Name**, type `MDM` and for **Certificate Password** enter the password you used to protect the exported .p12 file.

      Click **Choose File** to select the .p12 file and click **Apply** to upload the file and save the APNS settings to Mobile Security Manager.

11.2.11 Configuring the GCM Entry

For descriptions of the GCM Settings form fields, see "Google Cloud Messaging (GCM) Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Mobile Security Manager requires GCM (Google Cloud Messaging) credentials to connect to GCM and send push notifications to Android devices. Follow these steps to create a GCM key.

1. Create a Google API project and enable the GCM service.

   a. Sign in with Google credentials to the Google Developers Console:

      `https://cloud.google.com/console`

   b. If you have an API Project, click it to open the Project Dashboard.

      If you do not have an API project yet, click **Create Project**. Specify a Project Name and click **Create**.

      A page opens and displays your project number—for example, Project Number: 670330094152.

      Copy the project number. You will need it when you upload the API key to Mobile Security Manager.

   c. Choose **APIs & auth** > **APIs** from the sidebar, then, under **Mobile APIs**, click **Cloud Messaging for Android**.

      Click **Enable API**.

      Google Cloud Messaging is enabled.

2. Obtain an API key.

   a. Choose **APIs & auth** > **Credentials** from the sidebar.

   b. In the **Public API access** section, click **Create new Key**, then click **Server key** in the **Create a new key** dialog.

      The ***Create a server key and configure allowed IPs*** *dialog box opens.*

   c. Enter your server’s IP address and click **Create**.

The API key is created.

d. Copy the API key.

3. Upload the API key to Mobile Security Manager.

   a. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."

   b. Click **GCM Settings** on the menu bar. (If GCM Settings is not visible, use the arrow buttons to scroll the menu bar to the right. Or, click ▼ to view additional menu items.)

   c. Click **Add** to create a new row in the settings table.

   d. For **Application ID**, type `MDM`.

      For **Sender ID**, enter the project number from step 1.

      For **API key**, paste the API key from step 2.

   e. Click **Apply** to save the GCM settings to Mobile Security Manager.

4. After modifying the values, click **Apply** to save changes or **Revert** to discard the changes.

   Click **Refresh** to view the updated changes on the back-end server.

11.2.12 Configuring Notification Templates

For descriptions of the Notification Templates form fields, see "Notification Templates" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Use this tab to manage the Invite templates that the system uses to provide notification to users.

- Multiple instances of a template can be created in different languages. First select a template, then click **Add New Language**. Choose a language from the menu. A new tab shows the name of the selected language. Use the editor to format the message content as needed.
- You can also delete templates: you can delete just a specific locale, or you can delete a template and all of its locales.

To learn how to create or edit a notification template, see Section 3.2.1.1, "How to Create and Edit Notification Templates."

11.2.13 Configuring MDM Agent Settings

For descriptions of the MDM Agent Settings form fields, see "MDM Agent Settings" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Use this tab to edit iOS Mobile Device Management (MDM) settings. The Android client does not accept the settings entered on the MDM Agent Settings tab. For Android, see Section 10.2.6, "Change MDM Agent Settings" to configure MDM agent values on the Secure Workspace app.

11.2.14 Configuring Blacklisted Apps

For descriptions of the Blacklisted Apps form fields, see "Blacklisted Apps" in the *Oracle Fusion Middleware Help Reference for Oracle Mobile Security Suite Consoles.*

Use this tab to manage prohibited apps on the device. Apps can only be blacklisted on managed devices. Mobile security policies can check for blacklisted apps during enrollment and take action if a blacklisted app is found on the device. Following device enrollment, mobile security policies can check for blacklisted apps and, if one is found, take appropriate action as defined in the policy.

12 Configuring Your Environment to Work With Mobile Security Manager

This chapter documents configuration steps that may be required to get Mobile Security Manager working in your environment. It is organized into the following sections:

- Tuning Oracle Mobile Security Suite
- Configuring the Identity Store Configuration
- Configuring NDES and the Active Directory Certificate Authority
- Configuring Automatic Certificate Revocation with the Active Directory Certificate Authority
- Configuring Microsoft Exchange (Secure Mail) to Work With Mobile Security Manager
- Configuring Oracle Mobile Security Suite to use Oracle Access Management 11gR2 PS2 for Authentication and SSO

> **Note:** See the *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite* to learn how to integrate Oracle Identity Manager Oracle Mobile Security Suite with Oracle Mobile Security Suite.

12.1 Tuning Oracle Mobile Security Suite

See "Tuning Oracle Mobile Security Suite" in the *Oracle Fusion Middleware Performance and Tuning Guide* for information about tuning the heap size, tuning the datasource connection pool settings, and other tuning recommendations.

12.2 Configuring the Identity Store Configuration

To configure the identity store connection, create an Identity Directory Service Profile in the Oracle Access Management console. Then in Mobile Security Manager, set the IDS Profile Name in the Identity Store Settings tab.

- To learn how to configure the identity store connection, see "Managing User Identity Store Registrations" in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.
- To set the IDS Profile Name in the Identity Store settings tab, see Section 11.2.5, "Configuring Identity Store Settings."

12.3 Configuring NDES and the Active Directory Certificate Authority

This section describes how to configure Windows Enterprise 2008 R2 machines so that the Simple Certificate Enrollment Protocol (SCEP) works with the Mobile Security Manager server. This configuration is required for Secure Workspace enrollment.

Before you begin, install the Network Device Enrollment Service (NDES) on either the Windows Enterprise 2012 R2 or Windows Enterprise 2008 R2 machine.

> **Note:** Ensure that the user password for the user account specified for the NDES configuration never expires.

1. Apply the required hotfixes. The following hotfixes must be applied to Windows 2008 R2 machines before you configure NDES:

   - KB 959193 http://support.microsoft.com/kb/959193
   - KB 2633200 http://support.microsoft.com/kb/2633200
   - KB 2483564 http://support.microsoft.com/kb/2483564

2. Configure the Network Device Enrollment Services (NDES) on your Active Directory server by completing the Setup instructions found in the Microsoft TechNet article "Network Device Enrollment Service (NDES) in Active Directory Certificate Services (AD CS):

   http://social.technet.microsoft.com/wiki/contents/articles/9063.network-device-enrollment-service-ndes-in-active-directory-certificate-services-ad-cs.aspx

3. Increase the MAX URL limit. Because SCEP requests use the HTTP GET method, the length of the URL can exceed the limit due to the certificate signing request present in the request URL.

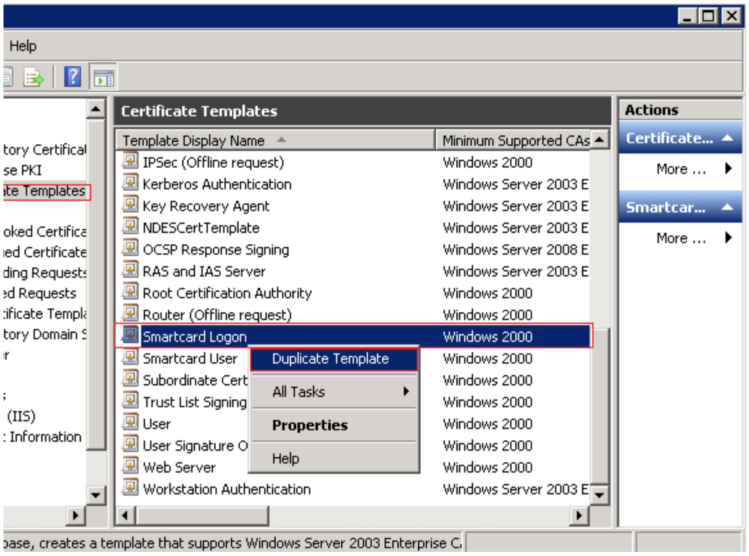
   ```
   %windir%\system32\inetsrv\appcmd set config /section:requestfiltering
   /requestlimits.maxurl:4096

    %windir%\system32\inetsrv\appcmd set config /section:requestfiltering
   /requestlimits.maxquerystring:4096
   ```

4. Extend the Smart Card Logon Certificate Template to create a new template to be used while issuing certificates.

   a. Open Server Manager and right-click the **Smartcard Logon** template.

      Select **Duplicate Template**.

      The Duplicate Template dialog opens.



Certificate Templates

Template Display Name	Minimum Supported CAs
IPSec (Offline request)	Windows 2000
Kerberos Authentication	Windows Server 2003 E
Key Recovery Agent	Windows Server 2003 E
NDESCertTemplate	Windows Server 2003 E
OCSP Response Signing	Windows Server 2008 E
RAS and IAS Server	Windows Server 2003 E
Root Certification Authority	Windows 2000
Router (Offline request)	Windows 2000
Smartcard Logon	Windows 2000
Smartcard User	Windows 2000
Subordinate Cert	Windows 2000
Trust List Signing	Windows 2000
User	Windows 2000
User Signature O	Windows 2000
Web Server	Windows 2000
Workstation Authentication	Windows Server 2003 E

pase, creates a template that supports Windows Server 2003 Enterprise C.

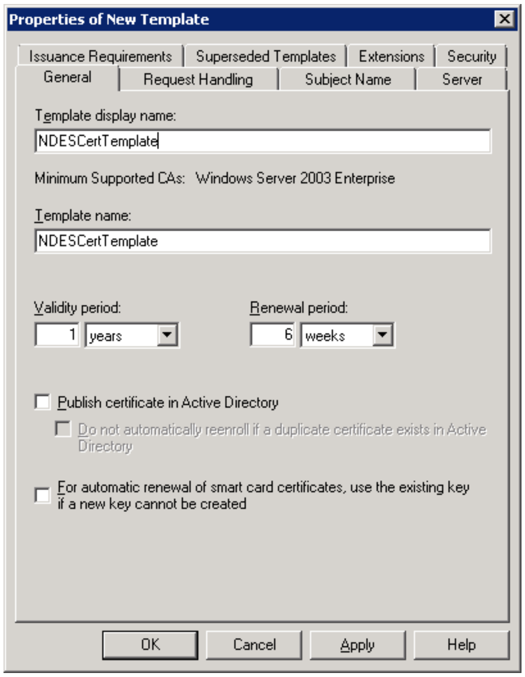
Screen capture shows the Server Manager configuration screen. "**Smartcard Logon**" is highlighted in the "**Certificate Templates**" pane, and "**Duplicate Template**" is selected.

****

b. Select **Windows Server 2003 Enterprise** and click OK.

*The **Properties of New Template** form opens.*

c. On the **General** tab of the Properties of New Template form, type a template name. You will use this name when configuring NDES.



Template display name: NDESCertTemplate

Minimum Supported CAs: Windows Server 2003 Enterprise

Template name: NDESCertTemplate

Validity period: 1 years

Renewal period: 6 weeks

☐ Publish certificate in Active Directory

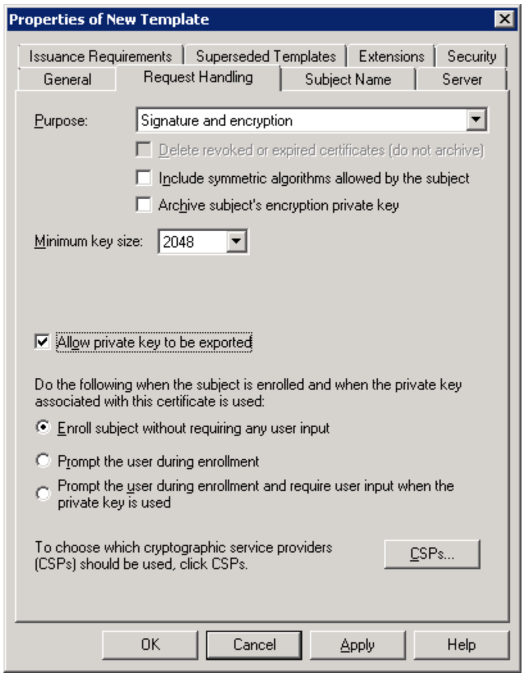
☐ Do not automatically reenroll if a duplicate certificate exists in Active Directory

☐ For automatic renewal of smart card certificates, use the existing key if a new key cannot be created

Screen capture shows the "**General**" tab portion of the "Properties of New Template" form.

****

d. On the **Request Handling** tab of the Properties of New Template form, select **Allow private key to be exported**. Do not change the other values.



Purpose: Signature and encryption

□ Delete revoked or expired certificates (do not archive)

□ Include symmetric algorithms allowed by the subject

□ Archive subject's encryption private key

Minimum key size: 2048

☑ Allow private key to be exported

Do the following when the subject is enrolled and when the private key associated with this certificate is used:

● Enroll subject without requiring any user input

○ Prompt the user during enrollment

○ Prompt the user during enrollment and require user input when the private key is used

To choose which cryptographic service providers (CSPs) should be used, click CSPs.

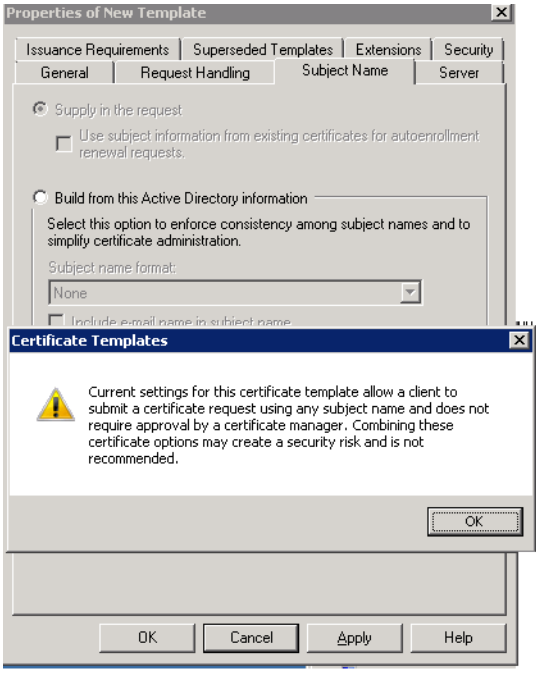
Screen capture shows the "Request Handling" tab portion of the Properties of New Template" form, which is configured as follows: The "**Purpose**" setting is set to "**Signature and encryption**"; The **minimum key size** is **2048**; The "**Allow private key to be exported**" option is selected.

****

e. On the **Subject Name** tab of the Properties of New Template form, select **Supply in the request**.

A warning message opens.

Click OK to close the warning, then click **Apply**, and click OK to complete the template creation.



● Supply in the request

□ Use subject information from existing certificates for autoenrollment renewal requests.

○ Build from this Active Directory information

Select this option to enforce consistency among subject names and to simplify certificate administration.

Subject name format: None

□ Include e-mail name in subject name

Certificate Templates

Current settings for this certificate template allow a client to submit a certificate request using any subject name and does not require approval by a certificate manager. Combining these certificate options may create a security risk and is not recommended.

Screen capture shows the warning message, which reads "Current settings for this certificate template allow a client to submit a certificate request using any subject name and does not require approval by a certificate manager. Combining these certificate options may create a security risk and is not recommended."

****

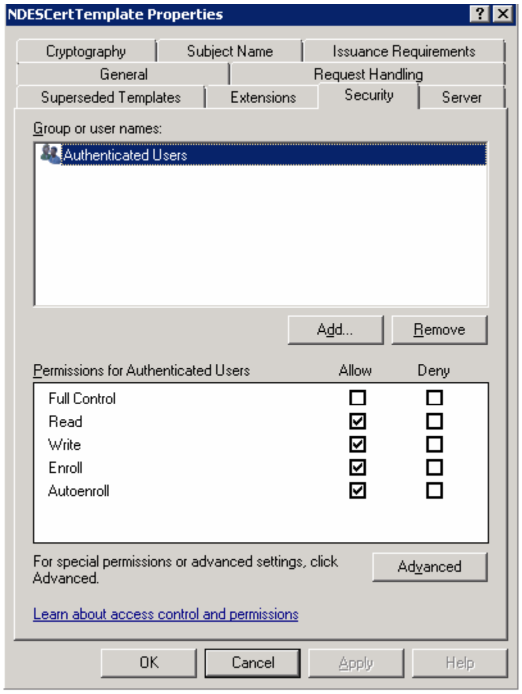
f. In the Server Manager window, expand the certificate authority node and choose the **Certificate Templates** entry under it.

   Select the recently created certificate.

g. From the menu, choose **Action** > **New** > **Certificate Template to Issue**.

   Select the newly created certificate template.

5. Set permissions for the certificate template by selecting **Allow** for **Read**, **Write**, **Enroll**, and **Autoenroll**.



Permissions for Authenticated Users	Allow	Deny
Full Control	□	□
Read	☑	□
Write	☑	□
Enroll	☑	□
Autoenroll	☑	□

Screen capture shows the "**Security**" tab portion of the "NDESCertTemplate Properties" form. The "**Authenticated Users**" group name is selected, and in the "**Permissions for Authenticated Users**" section, "**Read**, **Write**, **Enroll**, and **Autoenroll**" are marked as **Allow**.

****

6. Configure the following values so that Certification Authority includes SAN in the certificate

```
certutil -setreg policy\EditFlags +EDITF_ATTRIBUTESUBJECTALTNAME2

net stop certsvc
net start certsvc
```

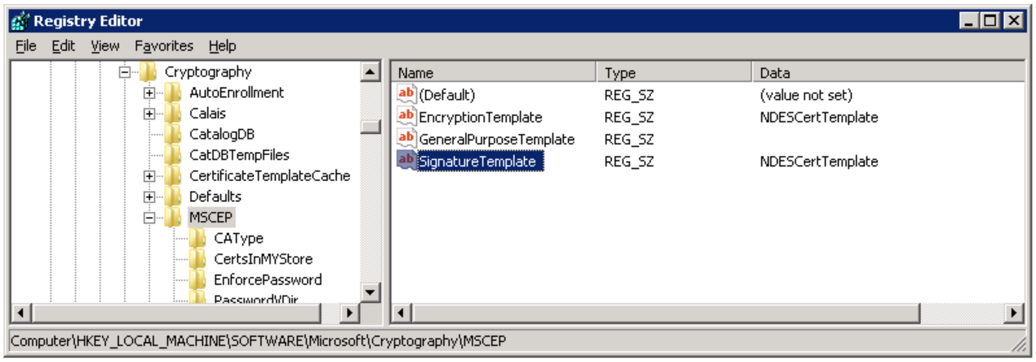
7. Set the NDES configuration to disable the password policy and to use the Certificate Template for certificate issuance.

   Update the registry settings for SCEP as follows:

   a. Open the following registry key:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\MSCEP
   ```

   b. Add the new key and value as shown in the screen capture.



Name	Type	Data
(Default)	REG_SZ	(value not set)
EncryptionTemplate	REG_SZ	NDESCertTemplate
GeneralPurposeTemplate	REG_SZ	
SignatureTemplate	REG_SZ	NDESCertTemplate

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\MSCEP

Screen capture shows the MSCEP registry settings in the Registry Editor. The navigation tree shows that the Registry Editor is open to the **Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography > MSCEP** folder. The main pane contains three columns (**Name**, **Type**, and **Data**) and four registry entries: Row one shows: `(Default)`, `REG_SZ`, `(value not set)`; Row two shows: `EncryptionTemplate`, `REG_SZ`, `NDESCertTemplate`; Row three shows: `GeneralPurposeTemplate`, `REG_SZ`, and column 3 has a blank value; Row four shows: `SignatureTemplate`, `REG_SZ`, `NDESCertTemplate`.

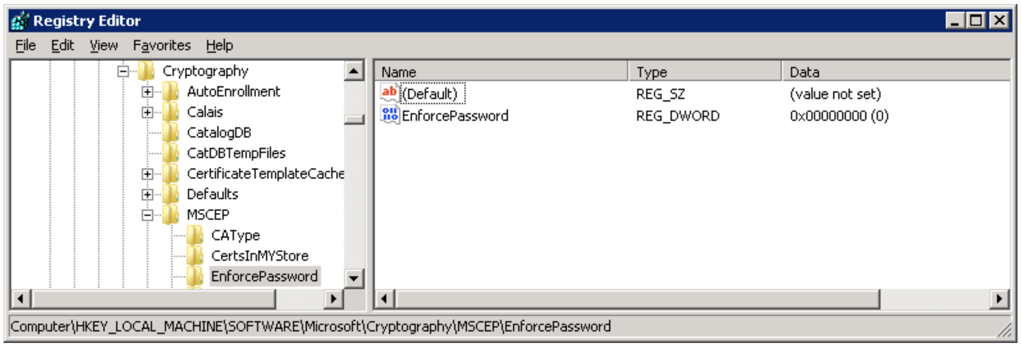
****

8. Disable the password policy for NDES.

   a. Open the following registry key:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\MSCEP\
   EnforcePassword
   ```

   b. Change the value for **EnforcePassword** to `0x00000000`.



Name	Type	Data
(Default)	REG_SZ	(value not set)
EnforcePassword	REG_DWORD	0x00000000 (0)

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\MSCEP\EnforcePassword

Screen capture shows the "EnforcePassword" registry setting in the Registry Editor set to `0x00000000`. The navigation tree shows that the Registry Editor is open to the **Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography > MSCEP\EnforcePassword** folder. The main pane contains three columns (**Name**, **Type**, and **Data**) and two registry entries: Row one shows: `(Default)`, `REG_SZ`, `(value not set)`; Row two shows: `EnforcePassword`, `REG_DWORD`, `0x00000000 (0)`.

****

9. Restart the NDES SCEP server or the Internet Information Server (IIS).

12.4 Configuring Automatic Certificate Revocation with the Active Directory Certificate Authority

When the Mobile Security Manager component of the Oracle Mobile Security Suite is configured to provision certificates from a Microsoft CA, by default those certificates will not be automatically revoked when mobile devices and Workspace apps are wiped or deregistered. Mobile Security Manager uses the Network Device Enrollment Service (NDES) for provisioning certificates. NDES, however, does not provide a network interface for certificate revocation.

This section describes how to configure a Certificate Revocation Application in Internet Information Server (IIS) on the NDES server. Mobile Security Manager will invoke this application so that mobile device and user certificates are automatically revoked at the appropriate points in the lifecycle of mobile devices and workspace apps.

The URL that Mobile Security Manager calls to revoke certificates is computed based on the NDES URL provided in the Mobile Security Manager **CA Settings** tab. The URL for certificate revocation will be constructed like the following:

```
http(s)://<ca_host>:<ca_port>/CertService/revoke.php
```

The Mobile Security Manager server sends POST requests to revoke the certificates, indicating “Cease of Operation” as the reason for the revocation.

This section includes the following topics:

- Steps to Configure Certificate Revocation Application
- Deploy the Certificate Revocation Application in IIS
- Deploy the Certificate Revocation PHP Scripts
- Import the Mobile Security Manager Certificate in IIS
- Configuring IIS to run PHP applications

12.4.1 Steps to Configure Certificate Revocation Application

Before you begin: The Certification Revocation Application is a PHP application. It requires that Internet Information Server (IIS) on your NDES server support PHP applications. If it does not, then follow the Steps in Section 12.4.5, "Configuring IIS to run PHP applications."

Configuring the Certificate Revocation Application involves the following steps:

1. Deploy Certificate Revocation Application in IIS – In this step you create and configure the PHP application in IIS.
2. Deploy Certificate Revocation PHP Scripts – In this step you deploy the PHP scripts provided by Oracle to the physical path of the Certificate Revocation Application.
3. Import Mobile Security Manager Certificate – In this step you import the Mobile Security Manager certificate to the physical path of the Certificate Revocation Application so that it can authenticate certificate revocation requests as originating from Mobile Security Manager.

12.4.2 Deploy the Certificate Revocation Application in IIS

This section describes how to deploy the Certificate Revocation Application in IIS.

1. Add an Application Pool

a. Open the IIS Manager application for the NDES IIS instance. Select **Application Pools**, and in the **Actions** pane, select **Add Application Pool**.

The Add Application Pool dialog opens.

b. Complete the form as follows and click **OK**:

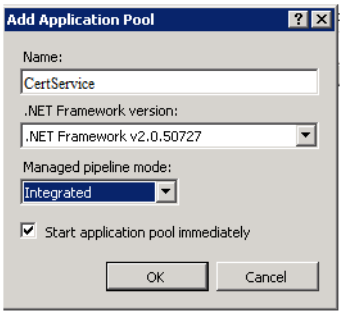
Name – Type *CertService*.

.NET Framework version – Use the default value.

Managed pipeline mode – Choose *Integrated*.

Start application pool immediately – Select this option.

Figure 12–1 Complete the Add Application Pool form



Add Application Pool

Name: CertService

.NET Framework version: .NET Framework v2.0.50727

Managed pipeline mode: Integrated

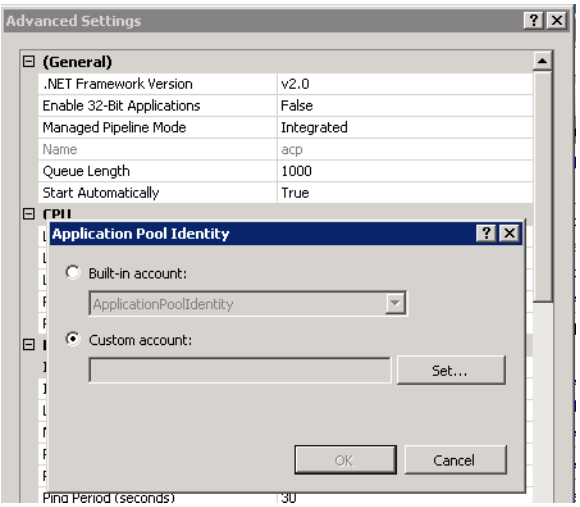
☑ Start application pool immediately

c. Edit the newly created **CertService** application pool.

In the **Actions** pane, select **Advanced** and click **ApplicationPoolIdentity** to set Identity as a privileged service account.

The Application Pool Identity dialog opens.

Figure 12–2 The Application Pool Identity dialog



Advanced Settings

(General)	
.NET Framework Version	v2.0
Enable 32-Bit Applications	False
Managed Pipeline Mode	Integrated
Name	acp
Queue Length	1000
Start Automatically	True

CPU

Ping Period (seconds) 30

Application Pool Identity

○ Built-in account: ApplicationPoolIdentity

● Custom account:

d. Select **Custom account** and click **Set**.

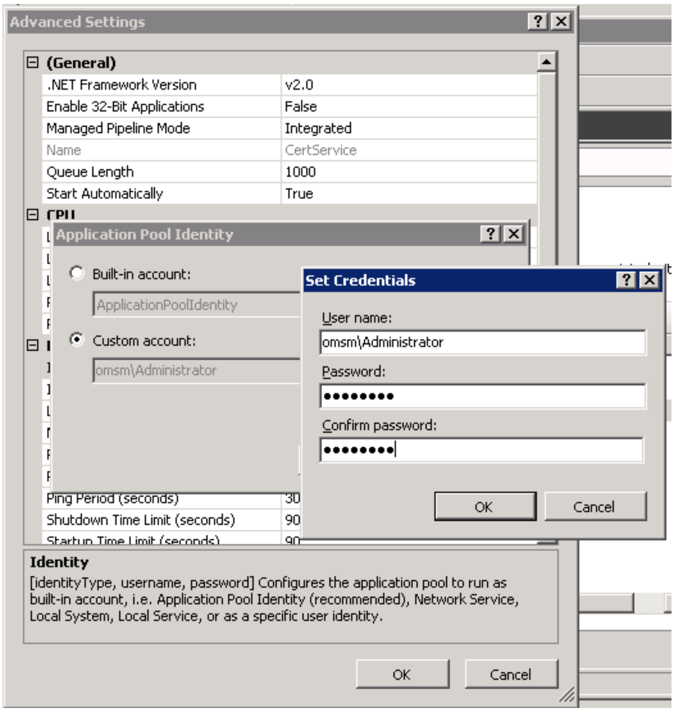
The Set Credentials dialog opens.

e. In the **User name** field, enter the user name of a Windows account that has permission to manage certificates at the Certificate Authority level (assigned on the security properties of the CA).

Enter the password in the **Password** and **Confirm password** fields and click **OK**.

Click OK to close the other open dialogs.

Figure 12–3 Enter CA administrator credentials in the Set Credentials dialog



Advanced Settings

(General)

.NET Framework Version	v2.0
Enable 32-Bit Applications	False
Managed Pipeline Mode	Integrated
Name	CertService
Queue Length	1000
Start Automatically	True
Ping Period (seconds)	30
Shutdown Time Limit (seconds)	90
Startup Time Limit (seconds)	90

Identity

[identityType, username, password] Configures the application pool to run as built-in account, i.e. Application Pool Identity (recommended), Network Service, Local System, Local Service, or as a specific user identity.

Application Pool Identity

○ Built-in account: ApplicationPoolIdentity

● Custom account: omsm\Administrator

Set Credentials

User name: omsm\Administrator

Password: ••••••••

Confirm password: ••••••••

The service account is shown as the identity for the application pool.

2. Right-click the Default Web Site in the IIS NDES instance and select **Add New Application**.

The Add Application dialog opens.

In the **Alias** field, enter *CertService*, then click **Select** and choose CertService as the Application Pool to use.

In **Physical path**, click the **...** button and select a physical path for the application.

Click **OK** to finish creating the new Certificate Revocation Application.

3. Update the `php.ini` file.

Add the following line to the *<PHP_INSTALLATION_HOME>*`/php.ini` file to enable the OpenSSL extension for PHP:

```
extension=php_openssl.dll
```

12.4.3 Deploy the Certificate Revocation PHP Scripts

This section describes how to deploy two PHP scripts for certificate revocation to the physical path of the Certificate Revocation Application.

1. Create a text file named `revoke.php` and add the contents of the following code snippet.

   Save the file to the root of the physical path that you selected for the Certificate Revocation Application in step 2 of Section 12.4.2, "Deploy the Certificate Revocation Application in IIS."

Contents of the `revoke.php` file

```
<?php
function checkNull($var){
    return (!isset($var) || is_null($var));
}
function checkNullOrEmpty($var){
    return (!isset($var) || trim($var)==='');
}

include_once "JWT.php";
$postdata = file_get_contents("php://input");
$result = 1;
$reasonCode ="";
try{
    $ar = json_decode($postdata);
    if( ! checkNull($ar) ){
        if(!(checkNullOrEmpty($ar->serialnumber)
              || checkNullOrEmpty($ar->authtoken)
              || checkNullOrEmpty($ar->caauthority)
              || checkNullOrEmpty($ar->reason))) {

              $check = $ar->caauthority . ":" . $ar->serialnumber;
              $res_pubkey = openssl_pkey_get_public(file_get_
contents("certs/SignerCertificate.pem"));
              $payload = JWT::decode($ar->authtoken, $res_pubkey, true);
              if(strcmp($check, $payload->sub) == 0){
                  $cmd="certutil -config \"$ar->caauthority\"  -revoke
$ar->serialnumber $ar->reason" ;
                  exec($cmd);
                  $status = "0";
                  $message="Certificate Revoked Successfully" ;
                  $result = 0;
              } else {
                  $status = "337";
                 $message = "Revoke request does not match with Authentication
subject";
                  $result = 4;
              }
        } else {
                  $result = 2;
                  $message = "Missing request parameters";
        }
    } else {
            $result = 3;
            $status = "335";
            $message = "Http Get is not supported";
    }
} catch (Exception $ex){
   $result = 1;
   $reasonCode = "Authentication Token verification failed";
}
if($result == 1){
    $status = "333" ;
    $message = "Certificate revoke failed.  Reason $reasonCode";
} else if($result == 2){
    $status = "333";
}
$data="{ \"message\": \"$message\", \"code\": \"$status\"}";
header("Content-Type: application/json");
print ($data);
?>
```

2. Create a text file named `JWT.php` and add the contents of the following code snippet.

   Save the file to the root of the physical path that you selected for the Certificate Revocation Application in step 2 of Section 12.4.2, "Deploy the Certificate Revocation Application in IIS."

Contents of the `JWT.php` file

```
<?php
class JWT
{
    public static function encode($payload, $key, $algo = 'HS256')
    {
       $header = array('typ' => 'JWT', 'alg' => $algo);
       $segments = array(
           JWT::urlsafeB64Encode(json_encode($header)),
           JWT::urlsafeB64Encode(json_encode($payload))
       );
       $signing_input = implode('.', $segments);
       $signature = JWT::sign($signing_input, $key, $algo);
       $segments[] = JWT::urlsafeB64Encode($signature);
       return implode('.', $segments);
    }

    public static function decode($jwt, $key = null, $verify = true)
    {
       $tks = explode('.', $jwt);
       if (count($tks) != 3) {
           throw new Exception('Wrong number of segments');
       }
       list($headb64, $payloadb64, $cryptob64) = $tks;
       if (null === ($header = json_decode(JWT::urlsafeB64Decode($headb64))))
       {
           throw new Exception('Invalid segment encoding');
       }
      if (null === $payload = json_decode(JWT::urlsafeB64Decode($payloadb64)))
      {
           throw new Exception('Invalid segment encoding');
       }
       $sig = JWT::urlsafeB64Decode($cryptob64);
       if ($verify) {
           if (empty($header->alg)) {
               throw new DomainException('Empty algorithm');
           }
           if (!JWT::verifySignature($sig, "$headb64.$payloadb64", $key,
$header->alg)) {
               throw new UnexpectedValueException('Signature verification
failed');
           }
      }
      return $payload;
```

```
   }

    public static function getSignature($jwt, $key = null, $verify = true)
    {
       $tks = explode('.', $jwt);
       if (count($tks) != 3) {
           throw new Exception('Wrong number of segments');
       }
       list($headb64, $payloadb64, $cryptob64) = $tks;
       if (null === ($header = json_decode(JWT::urlsafeB64Decode($headb64)))) {
           throw new Exception('Invalid segment encoding');
       }
       if (null === $payload = json_decode(JWT::urlsafeB64Decode($payloadb64)))
      {
           throw new Exception('Invalid segment encoding');
       }
       $sig = JWT::urlsafeB64Decode($cryptob64);
       return $sig;
    }

    private static function verifySignature($signature, $input, $key, $algo =
'HS256')
    {
       switch ($algo) {
           case'HS256':
           case'HS384':
           case'HS512':
               return JWT::sign($input, $key, $algo) === $signature;
           case 'RS256':
               return (boolean) openssl_verify($input, $signature, $key,
OPENSSL_ALGO_SHA256);
           case 'RS384':
               return (boolean) openssl_verify($input, $signature, $key,
OPENSSL_ALGO_SHA384);
           case 'RS512':
               return (boolean) openssl_verify($input, $signature, $key,
OPENSSL_ALGO_SHA512);
           default:
               throw new Exception("Unsupported or invalid signing
algorithm.");
       }
    }
    private static function sign($input, $key, $algo = 'HS256')
    {
       switch ($algo) {
           case 'HS256':
               return hash_hmac('sha256', $input, $key, true);
           case 'HS384':
               return hash_hmac('sha384', $input, $key, true);
           case 'HS512':
               return hash_hmac('sha512', $input, $key, true);
           case 'RS256':
               return JWT::generateRSASignature($input, $key, OPENSSL_ALGO_
SHA256);
           case 'RS384':
               return JWT::generateRSASignature($input, $key, OPENSSL_ALGO_
SHA384);
           case 'RS512':
               return JWT::generateRSASignature($input, $key, OPENSSL_ALGO_
SHA512);
```

```
           default:
               throw new Exception("Unsupported or invalid signing
algorithm.");
       }
    }
    private static function generateRSASignature($input, $key, $algo)
    {
       if (!openssl_sign($input, $signature, $key, $algo)) {
           throw new Exception("Unable to sign data.");
       }
       return $signature;
    }
    private static function urlSafeB64Encode($data)
    {
       $b64 = base64_encode($data);
       $b64 = str_replace(array('+', '/', '\r', '\n', '='),
               array('-', '_'),
               $b64);
       return $b64;
    }
    private static function urlSafeB64Decode($b64)
    {
       $b64 = str_replace(array('-', '_'),
               array('+', '/'),
               $b64);
       return base64_decode($b64);
    }
}
?>
```

12.4.4 Import the Mobile Security Manager Certificate in IIS

This section describes how to import the Mobile Security Manager certificate to the physical path of the Certificate Revocation Application. This step is necessary so that the Certificate Revocation Application can authenticate certificate revocation requests as originating from Mobile Security Manager.

1. Export the Mobile Security Manager certificate from the Mobile Security Manager server:

   a. Go to the Mobile Security Manager installation directory:

   ```
   <DOMAIN_HOME>/config/fmwconfig
   ```

   b. Export the certificate using the following command:

   ```
   keytool -alias oraclemsm -exportcert -file oraclemsm.crt -keystore
   OracleMSMCertificates.p12 -storepass <KeyStorePassword> -storetype
   pkcs12
   ```

   Tip: The keystore password can be obtained from CSF using map name `msm` and key `serverKeystoreKey`.

2. On the NDES server, use the following command to convert the resulting DER encoded certificate to PEM format.

   ```
   openssl x509 -inform der -in oraclemsm.crt -out SignerCertificate.pem
   ```

3. Copy the `SignerCertificate.pem` file to the `/certs` directory under the physical path of the Certificate Revocation Application.

   ```
   <PHYSICAL_PATH>/certs/SignerCertificate.pem
   ```

12.4.5 Configuring IIS to run PHP applications

This section describes how to configure IIS on your NDES server to support PHP applications.

Note: If your IIS instance is already configured to run PHP applications, then skip the steps in this section. If the CGI Role Service is configured, but PHP is not enabled, then skip step 1.

---

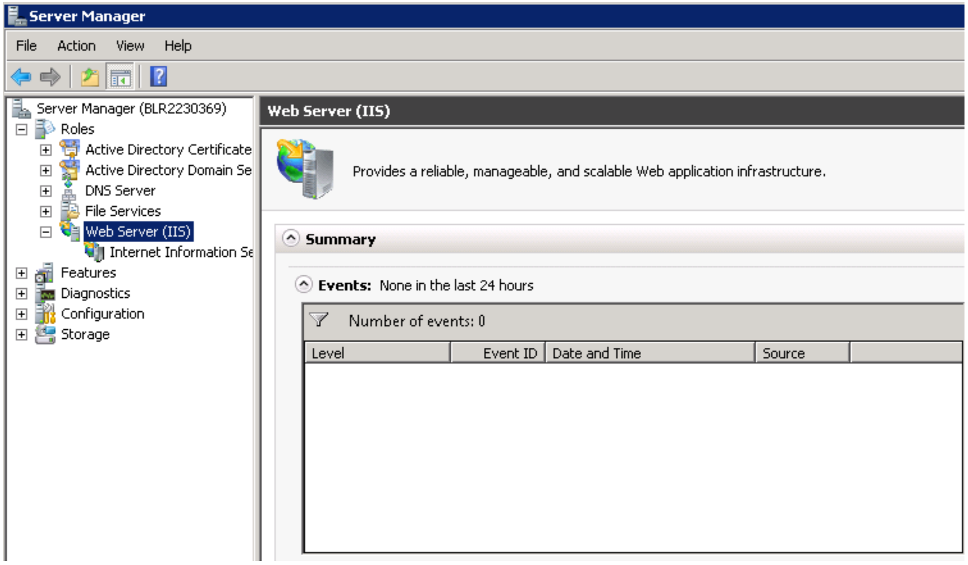
1. Add the CGI Role Service to IIS:

   a. Open the Server Manager application.

      In the pane on the left, expand **Roles**, right-click **Web Server (IIS)**, and select **Add Role Services**.

      The Add Role Services dialog opens.

Figure 12–4 The Server Manager application



Provides a reliable, manageable, and scalable Web application infrastructure.

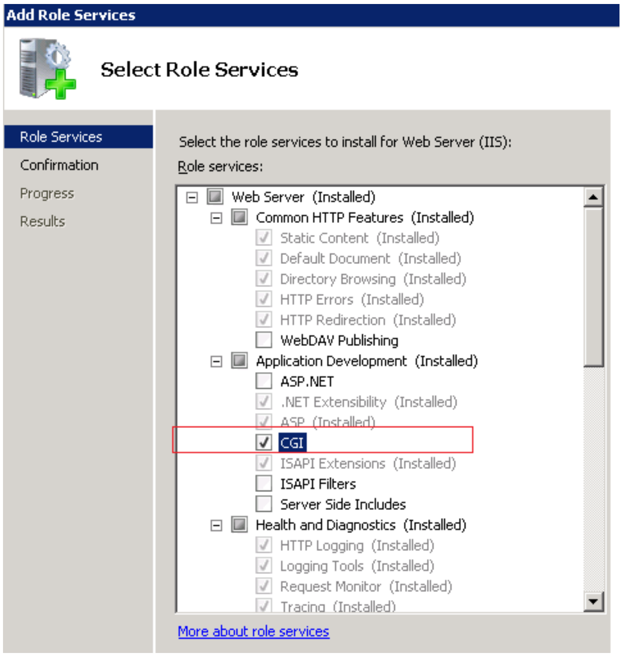
Events: None in the last 24 hours

Number of events: 0

Level	Event ID	Date and Time	Source

   b. Expand **Web Server (Installed)**, expand **Application Development (Installed)**, and select the **CGI** check box.

Figure 12–5 The Add Role Services dialog



Select the role services to install for Web Server (IIS):

Role services:

- ▣ Web Server (Installed)
  - ▣ Common HTTP Features (Installed)
    - ☑ Static Content (Installed)
    - ☑ Default Document (Installed)
    - ☑ Directory Browsing (Installed)
    - ☑ HTTP Errors (Installed)
    - ☑ HTTP Redirection (Installed)
    - □ WebDAV Publishing
  - ▣ Application Development (Installed)
    - □ ASP.NET
    - ☑ .NET Extensibility (Installed)
    - ☑ ASP (Installed)
    - ☑ CGI
    - ☑ ISAPI Extensions (Installed)
    - □ ISAPI Filters
    - □ Server Side Includes
  - ▣ Health and Diagnostics (Installed)
    - ☑ HTTP Logging (Installed)
    - ☑ Logging Tools (Installed)
    - ☑ Request Monitor (Installed)
    - ☑ Tracing (Installed)

c. Expand **Security** and select the **Windows Authentication** check box.

d. Expand **Management Tools** and select the **IIS 6 Management Compatibility** check box, which selects all of the sub-selections under **IIS 6 Management Compatibility**.

e. Select **Next**.

f. Select **Install**.

2. Configure IIS to handle PHP requests:

a. Open the Internet Information Services (IIS) Manager application for the NDES IIS instance.

Select the server, and double-click **Handler Mappings**.

b. In the **Actions** pane on the right, click **Add Module Mapping**.

c. Enter **php* for **Request path**.

d. Select **FastCGIModule** for **Module**.

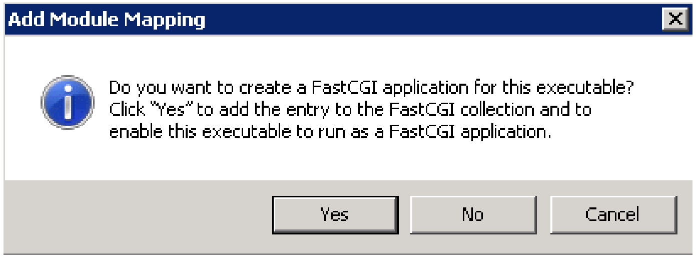
e. Enter `<PHP_INSTALLATION_HOME>/php-cgi.exe` as the Executable. If you browse, make sure that you select (*.exe) as the file type; otherwise it will be (*.dll) by default.

f. Enter a **Name** of *PHP using FASTCGI*.

g. Click the **OK** button.

The Add Module Mapping confirmation dialog opens.

Figure 12–6 The Add Module Mapping confirmation dialog



Do you want to create a FastCGI application for this executable? Click "Yes" to add the entry to the FastCGI collection and to enable this executable to run as a FastCGI application.

h. Click **Yes** to add the FastCGI application for PHP.

i. In the **Actions** pane, click **Edit Feature Permissions**....

The **Edit Feature Permissions** dialog opens.

Under **Script**, select **Execute**, then click **OK**.

j. In the **Actions** pane, click **Edit...**.

Click **Request Restrictions**, select the **Access** tab, and ensure that **Execute** is selected. Click **OK** to close the **Edit Module Mapping** dialog.

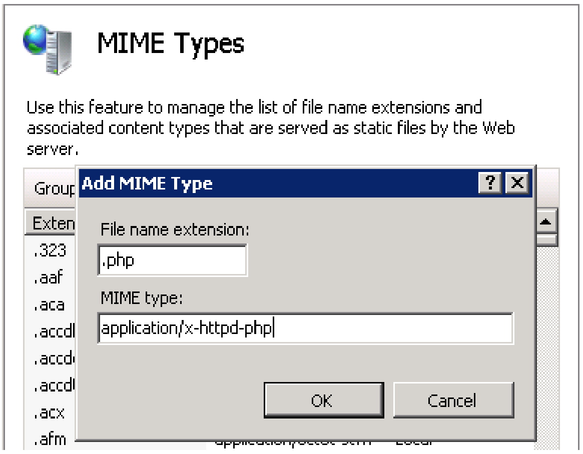
3. Add the PHP MIME type:

a. Select the NDES server instance in the IIS Manager application and double-click **MIME Types**.

b. In the **Actions** pane select **Add**.

The Add MIME Type dialog opens.

Figure 12–7 The Add MIME Type dialog



Use this feature to manage the list of file name extensions and associated content types that are served as static files by the Web server.

File name extension: .php

MIME type: application/x-httpd-php

c. Complete the form as follows and press OK:

File name extension -Enter `.php`

MIME type - Enter `application/x-httpd-php`

12.5 Configuring Microsoft Exchange (Secure Mail) to Work With Mobile Security Manager

Use these steps to configure Mobile Security Manager and Microsoft Exchange to work together.

1. Provide Mobile Security Manager with information about your Exchange server.

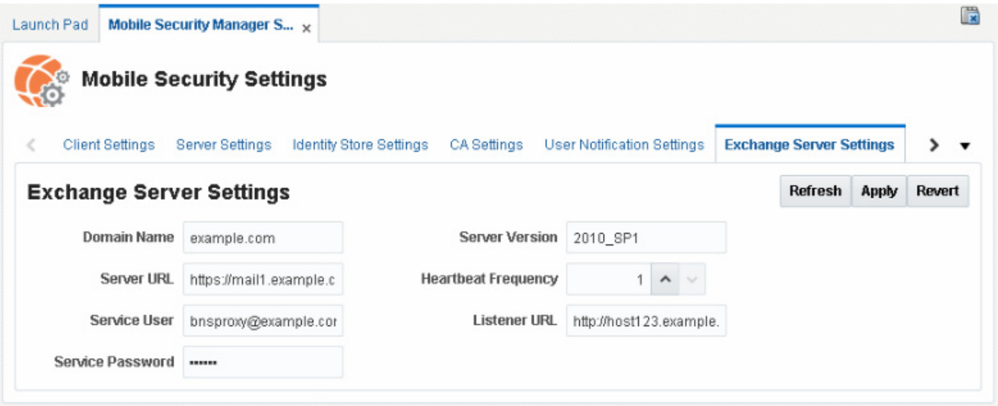
   a. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."

   b. Click **Exchange Server Settings** on the menu bar. (If Exchange Server Settings is not visible, use the arrow buttons to scroll the menu bar to the right. Or, click ▼ to view additional menu items.)

      The Exchange Server Settings page opens.

   c. Configure the Exchange Server Settings form. Use online help for field descriptions or see the *Help Reference for Oracle Mobile Security Suite Consoles*.

Figure 12–8 The Exchange Server Settings page



Exchange Server Settings

Domain Name	example.com	Server Version	2010_SP1
Server URL	https://mail1.example.c	Heartbeat Frequency	1
Service User	bnsproxy@example.cor	Listener URL	http://host123.example.
Service Password	••••••		

2. Modify the Workspace tab of the policy (or policies) to allow the end user access to e-mail on the Exchange server.

   a. Search for the first policy to update. To learn how, see Section 8.7.1, "How to Search for a Policy Record in Mobile Security Manager."

   b. In the search results section expand the policy details by clicking the policy record, then click the **Workspace** tab to open the Workspace policy.

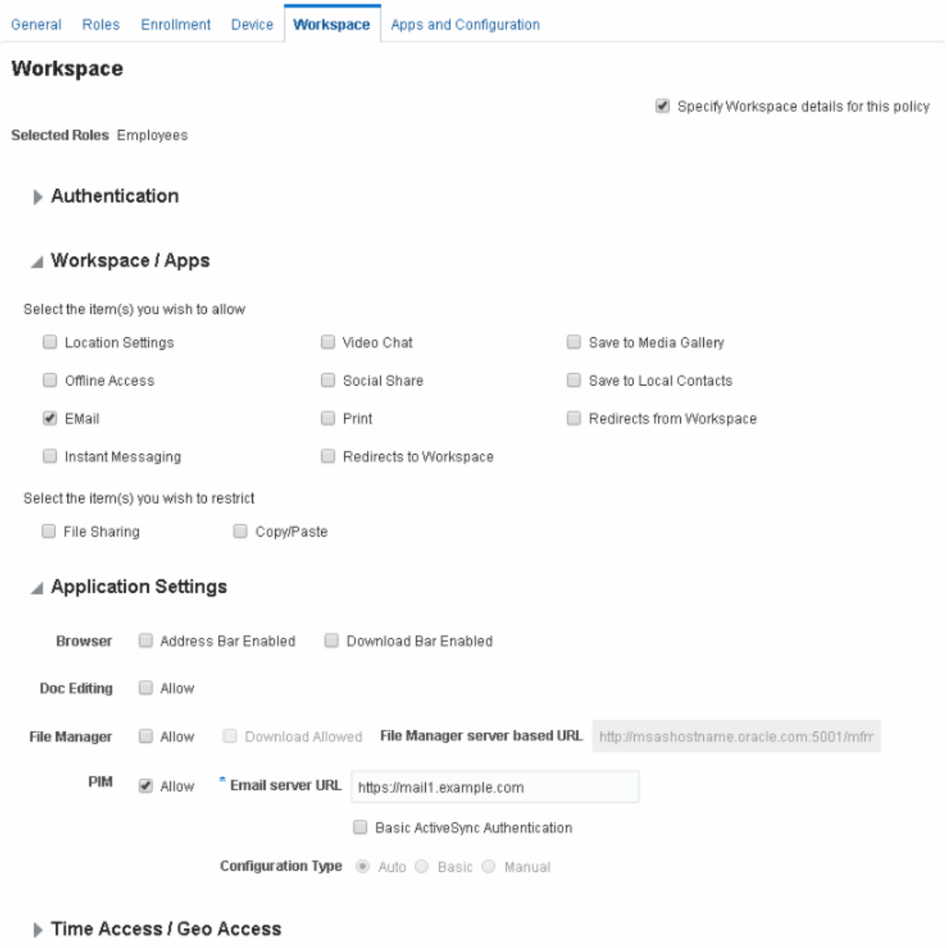
   c. Expand the **Workspace/Apps** section and select **Email** so that it is allowed.

   d. Expand the **Application Settings** section and select **Allow** next to **PIM** .

      In the **Email Server URL** field, enter the URL for the ActiveSync server, for example: `https://mail1.example.com`.

   e. Click **Apply** to save your changes, then repeat the steps for the remaining policies (if any) that need to be updated.

Figure 12–9 Policy settings that allow users to access e-mail



Workspace

☑ Specify Workspace details for this policy

Selected Roles Employees

▶ **Authentication**

◢ **Workspace / Apps**

Select the item(s) you wish to allow

□ Location Settings □ Video Chat □ Save to Media Gallery

□ Offline Access □ Social Share □ Save to Local Contacts

☑ EMail □ Print □ Redirects from Workspace

□ Instant Messaging □ Redirects to Workspace

Select the item(s) you wish to restrict

□ File Sharing □ Copy/Paste

◢ **Application Settings**

Browser □ Address Bar Enabled □ Download Bar Enabled

Doc Editing □ Allow

File Manager □ Allow □ Download Allowed **File Manager server based URL** http://msashostname.oracle.com:5001/mfrr

PIM ☑ Allow * **Email server URL** https://mail1.example.com

□ Basic ActiveSync Authentication

Configuration Type ● Auto ○ Basic ○ Manual

▶ **Time Access / Geo Access**

3. Import your Exchange Server's security certificate into the MSAS server's trust store.

   a. Download the Exchange Server certificate to a temp folder on the Oracle Access Management server, for example: `/tmp/example-exchange.cer`.

   b. Use the following WLST commands to import the certificate to the MSAS server's trust store:

```
wls:/offline>connect('@ADMIN_USER','@ADMIN_PWD','t3://@MSM_HOST:@MSM_ADMIN_
PORT')
wls:/idmdomain/serverConfig>svc = getOpssService(name='KeyStoreService')
wls:/idmdomain/serverConfig>svc.importKeyStoreCertificate(appStripe=
 '@MSAS_INSTANCE_NAME',name='ssltruststore',password='pass1234',
 alias='mailca', keypassword='',type='TrustedCertificate',
 filepath='/tmp/example-exchange.cer')
```

4. Configure Mobile Security Manager's APNS (Apple Push Notification Service) and GCM (Google Cloud Messaging) settings to receive push notifications from the APNS/GCM servers for new mail or calender requests.

   To configure APNS (for iOS devices):

   a. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."

b. Click **Apple Push Notification Service (APNS) Settings** on the menu bar. (If this option is not visible, use the arrow buttons to scroll the menu bar to the right. Or, click ▼ to view additional menu items.)

   The Apple Push Notification Service (APNS) Settings page opens.

c. Click **Add** to create a new row to the settings table. For **Certificate Name**, enter the name *Secure Mail*; for **Certificate Password**, enter the password; for **Certificate File**, click **Choose File** and upload the certificate.

   Use online help for field descriptions, or see the *Help Reference for Oracle Mobile Security Suite Consoles*.

d. Click **Apply** to save your changes.

To configure GCM (for Android devices):

a. Open the Mobile Security Settings page. To learn how, see Section 11.2.2, "How to Open the Mobile Security Settings Page."

b. Click **Google Cloud Messaging (GCM) Settings** on the menu bar. (If this option is not visible, use the arrow buttons to scroll the menu bar to the right. Or, click ▼ to view additional menu items.)

   The Google Cloud Messaging Service (GCM) Settings page opens.

c. Click **Add** to create a new row to the settings table. For **Application ID**, enter `com.nitrodesk.honey.nitroid`; for **Sender ID**, enter the Sender ID value; for **API Key**, enter the server authentication key that is saved on the third-party application server that gives the application server authorized access to Google services.

   Use online help for field descriptions, or see the *Help Reference for Oracle Mobile Security Suite Consoles*.

d. Click **Apply** to save your changes.

12.6 Configuring Oracle Mobile Security Suite to use Oracle Access Management 11gR2 PS2 for Authentication and SSO

If Oracle Access Management 11gR2 PS2 is already deployed in your environment, Oracle Mobile Security Suite can use that version for authentication and SSO, provided that OMSS is deployed on Oracle Access Management 11gR2 PS3 in a separate WLS domain. This configuration requires the use of Mobile and Social Services on Oracle Access Management 11gR2 PS3.

Before you Begin - Install Oracle Access Management 11gR2 PS3 on Host 1 and Oracle Access Management 11gR2 PS2 on Host 2.

1. Log on to the Oracle Access Management Console on Host 2 and create a WebGate profile for Mobile and Social Services using the default settings.

   The following options should be selected:

   - Allow Management Operations
   - Allow Token Scope Operations
   - Allow Master Token Retrieval
   - Allow Credential Collector Operations

   In the **Access Client Password** field, enter a password.

2. Next you will configure the OAM Authentication Token Service Provider on Host 1 (OAM 11.1.2.3) to use the WebGate on Host 2 to connect to OAM 11.1.2.2.

   Log on to the Oracle Access Management Console on Host 1 and choose **MobileSecurity** > **Mobile and Social Services**.

   In the **Service Providers** section locate **OAMAuthentication** and click **Edit**.

   The "out-of-the-box" Oracle Access Manager (OAM) Authentication Token Service Provider Configuration form opens.

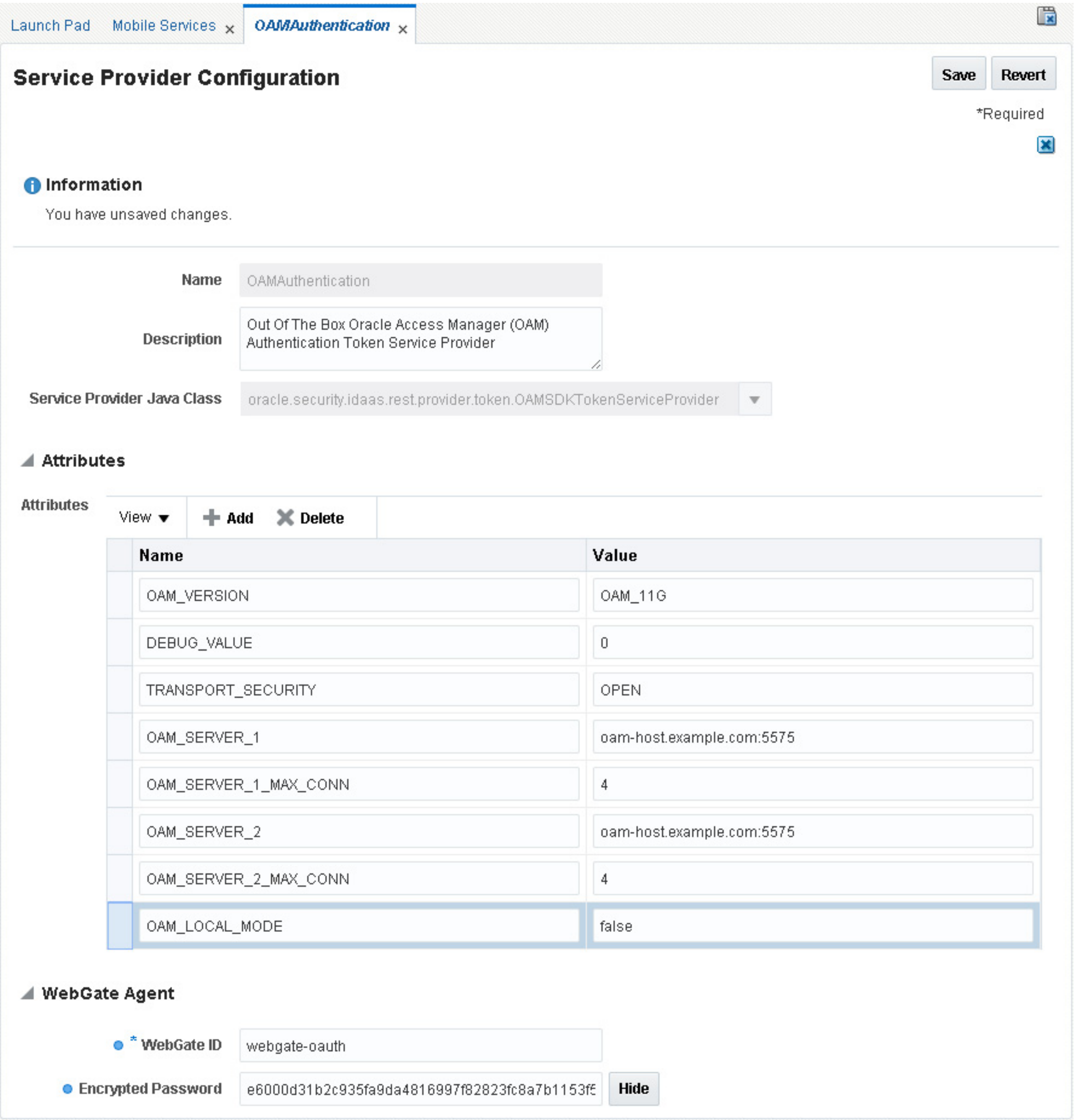
   Modify the form as follows:

   - For the `OAM_VERSION` attribute, keep the default value of `OAM_11G`.
   - Change the `OAM_SERVER_1` attribute value to use the correct OAM host name and port for the Host 2 server, for example:

     ```
     oam-host.example.com:5575
     ```

   - Change the `OAM_LOCAL_MODE` attribute value to `false`.
   - In the **WebGate Agent** section:
     - Change the **WebGate ID** value to the name of the WebGate you created using OAM 11gR2 PS2.
     - Replace the **Encrypted Password** by copying the `accessClientPasswd` value from the `ObAccessClient.xml` location on the OAM R2 PS2 server. For example: `(ParamName="accessClientPasswd" Value="`<*Encrypted password value to copy*>`")`

Figure 12–10 The OAM Authentication Token Service Provider Configuration form



Service Provider Configuration

*Required

Information

You have unsaved changes.

Name	OAMAuthentication
Description	Out Of The Box Oracle Access Manager (OAM) Authentication Token Service Provider
Service Provider Java Class	oracle.security.idaas.rest.provider.token.OAMSDKTokenServiceProvider

Attributes

Attributes

Name	Value
OAM_VERSION	OAM_11G
DEBUG_VALUE	0
TRANSPORT_SECURITY	OPEN
OAM_SERVER_1	oam-host.example.com:5575
OAM_SERVER_1_MAX_CONN	4
OAM_SERVER_2	oam-host.example.com:5575
OAM_SERVER_2_MAX_CONN	4
OAM_LOCAL_MODE	false

WebGate Agent

* WebGate ID	webgate-oauth
Encrypted Password	e6000d31b2c935fa9da4816997f82823fc8a7b1153f5

3. In this step you will modify the OAuth Mobile Service Provider on Host 1 to use Oracle Access Manager on Host 2. This will route Host 1 authentication requests to Host 2.

   Using the Oracle Access Management Console on Host 1, choose **MobileSecurity** > **Mobile OAuth Services** > ***YourDomain*** > **ServiceProviders > OAuthServiceProvider.**

   The Mobile Service Provider Configuration form opens.

   Modify the form as follows and click Save:

   - Change the `oam.WEBGATE_ID` attribute value to the name of the WebGate you created using OAM 11gR2 PS2.

   - Replace the `oam.ENCRYPTED_PASSWORD` attribute value by copying the `accessClientPasswd` value from the `ObAccessClient.xml` location on the

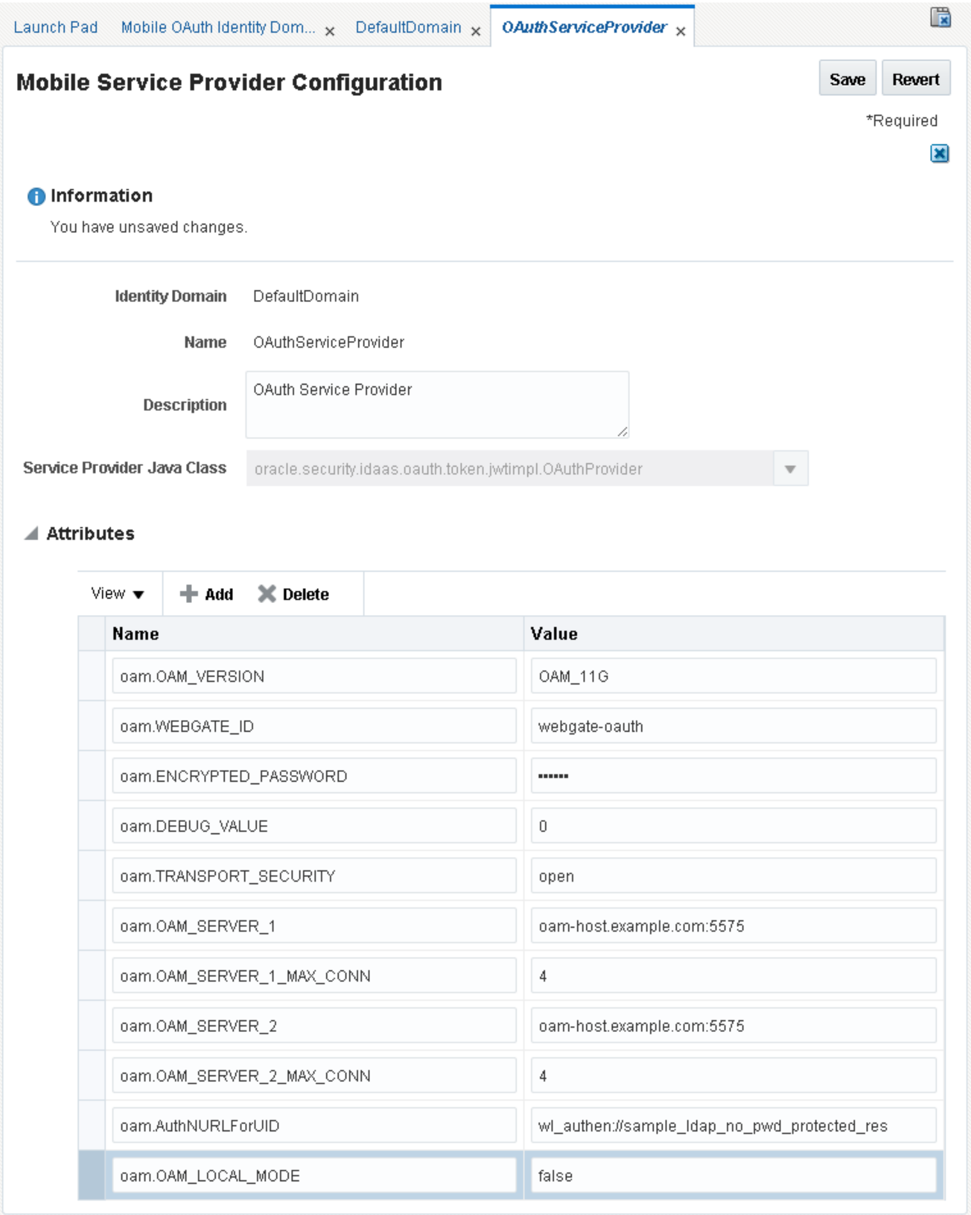
OAM R2 PS2 server. For example: `(ParamName="accessClientPasswd" Value="`*<Encrypted password value to copy>*`")`

- Change the `oam.OAM_SERVER_1` attribute value to use the correct OAM host name and port for the OAM 11gR2 PS2 server, for example:

  `oam-host.example.com:5575`

- Change the `oam.OAM_SERVER_2` attribute value to use the same OAM host name and port for the OAM 11gR2 PS2 server (`oam-host.example.com:5575`).
- Change the `oam.OAM_LOCAL_MODE` attribute value to false.

Figure 12–11 The OAuth Mobile Service Provider Configuration form



Mobile Service Provider Configuration

*Required

Information

You have unsaved changes.

Identity Domain	DefaultDomain
Name	OAuthServiceProvider
Description	OAuth Service Provider
Service Provider Java Class	oracle.security.idaas.oauth.token.jwtimpl.OAuthProvider

Attributes

Name	Value
oam.OAM_VERSION	OAM_11G
oam.WEBGATE_ID	webgate-oauth
oam.ENCRYPTED_PASSWORD	••••••
oam.DEBUG_VALUE	0
oam.TRANSPORT_SECURITY	open
oam.OAM_SERVER_1	oam-host.example.com:5575
oam.OAM_SERVER_1_MAX_CONN	4
oam.OAM_SERVER_2	oam-host.example.com:5575
oam.OAM_SERVER_2_MAX_CONN	4
oam.AuthNURLForUID	wl_authen://sample_ldap_no_pwd_protected_res
oam.OAM_LOCAL_MODE	false

4. On Host 1, you will prepare to merge the credential information from the Host 2 `cwallet` file into the Oracle Access Manager database. The Host 2 `cwallet` file was created when you created the WebGate profile on Host 2.

a. Navigate to the `fmwconfig` location on Host 1.

   At a command prompt on Host 1 type:

   ```
   cp jps-config-jse.xml jps-config-db-mig.xml
   ```

b. On Host 1, create the `/tmp/oam` directory, then paste the Host 2 `cwallet.sso` file into `/tmp/oam`:

   At a command prompt on Host 1 type:

   ```
   # mkdir /tmp/oam
   # cp <host>/cwallet.sso /tmp/oam
   ```

c. On Host 1, edit the `cwallet.sso` file, add the following values, and save the file:

   ```
   <serviceInstance location="/tmp/oam" provider="credstoressp"
   name="credential.file.source">
     <property name="location" value="/tmp/oam" />
   </serviceInstance>

   <jpsContext name="FileSourceContext">
     <serviceInstanceRef ref="credential.file.source"/>
   </jpsContext>

   <jpsContext name="FileDestinationContext">
     <serviceInstanceRef ref="credstore.db"/>
   </jpsContext>
   ```

5. Migrate the credentials by running this WLST command on Host 1:

   ```
   migrateSecurityStore(type="credStore",
   configFile="fmwconfig/jps-config-db-mig.xml",
   src="FileSourceContext",dst="FileDestinationContext")
   ```

6. Verify that the migration was successful by testing with Enterprise Manager.

   a. Open the Enterprise Manager (EM) console and navigate as follows: ***welogicdomain*** > **Security** > **Credentials**.

   b. Expand **OAMAgent** in the Credential Store.

   c. Check that a key was generated that corresponds with the WebGate profile that you created on Host 2 (OAM 11gR2 PS2). For example, if you created a WebGate with WebGate ID "webgate-oauth," you should have a key called "webgate-oauth_Key."

7. Restart the OAM server on Host 1.

13

Troubleshooting Oracle Mobile Security Suite

This chapter provides troubleshooting tips for Oracle Mobile Security Suite.

It includes the following sections:

13.1 Gathering Mobile Security Workspace Logs

You can mail client logs using the "Send logs" function, which packages all necessary client logs.

To gather detailed logs for troubleshooting:

1. Navigate to the Mobile Security Workspace application settings.
2. Turn on Log Mode.
3. Set Log Level to **verbose**.

13.2 Configuring Mobile Security Access Server Logs

Mobile Security Access Server (MSAS) components generate log files containing messages that record all types of events. For information on how to view and manage log files to assist in monitoring system activity and in diagnosing problems, see "Managing Log Files" in *Oracle Fusion Middleware Administering Oracle Mobile Security Access Server*.

13.3 Logging Mobile Security Manager Component Event Messages

Logging is the mechanism by which components write messages to a file. Administrators can use the logging mechanism to capture component events with information at various levels of granularity. Oracle Mobile Security Manager uses the same logging infrastructure and guidelines as any other component in Oracle Fusion Middleware 11*g*. This is accomplished by using the package `java.util.logging`, which is standard and available in all Java environments. The logging system writes output to flat files only.

Configuring logging and locating log files are the focus of this section. Diagnosing problems using the information in log files is outside the scope of this manual. This section contains the following topics:

- About Oracle Diagnostic Log File Management
- About Component Loggers for Oracle Mobile Security Manager
- About Logging Message Types and Levels
- Configuring the Loggers and Log Handlers Using Fusion Middleware Control
- Configuring the Loggers and Log Handlers Using the logging.xml File
- Configuring the Loggers and Log Handlers Using Fusion Middleware Control
- Configuring the Loggers and Log Handlers Using the logging.xml File

13.3.1 About Oracle Diagnostic Log File Management

Oracle Diagnostic Logging (ODL) is the principal logging service used by Oracle Mobile Security Manager. For ODL logging to work, both loggers and log handlers need to be configured. Loggers send messages to handlers, and handlers accept messages and output them to log files.

Oracle Mobile Security Manager makes use of the files in the below table

Table 13–1 Logging Files Used for Oracle Mobile Security Manager

File Type	Description
Logging Configuration File	Provides logging level and other configuration information for logging. This file is stored in the following path: `$`*`DOMAIN_HOME`*`/config/fmwconfig/servers/`*`SERVER-NAME`*`/logging.xml`
Log File	Logged information is stored in the following location: `$DOMAIN_HOME/servers/SERVER-NAME/logs/SERVER-NAME-diagnostics.log`

Logging configuration is controlled by the `logging.xml` file described in Section 13.3.5, "Configuring the Loggers and Log Handlers Using the logging.xml File." This file can either be edited directly (Section 13.3.5, "Configuring the Loggers and Log Handlers Using the logging.xml File") or edited through the Oracle Fusion Middleware Control (Section 13.3.4, "Configuring the Loggers and Log Handlers Using Fusion Middleware Control").

> **See Also:** For information about component loggers for Oracle Mobile Security Manager, see Section 13.3.2, "About Component Loggers for Oracle Mobile Security Manager."
>
> For information about logging message types and levels, see Section 13.3.3, "About Logging Message Types and Levels."

13.3.2 About Component Loggers for Oracle Mobile Security Manager

Oracle Mobile Security Manager has its own loggers, separate from the loggers of other Oracle Identity Management components, that can be configured independently to send different amounts of information to one or more log handlers.

Table 13–2 describes the component loggers for Oracle Mobile Security Manager.

Table 13–2 Oracle Mobile Security Manager Component Loggers

Logger Name	Description
oracle.idm.msm.common	Logs events related to settings, configuration and security which are used by other features internally.
oracle.idm.msm.identity	Logs events related to identity management.
oracle.idm.msm.notification	Logs events related to user and device notification.
oracle.idm.msm.platform	Logs events related to services provided by the platform (mainly used by other features). Includes services for persistence management, bean management, and so on.
oracle.idm.msm.policy	Logs events related to policy management.
oracle.idm.msm.repository	Logs events related to app catalog management, device operation queue and device configurations.
oracle.idm.msm.runtime	Logs events related to runtime services such as registration service, profile service, device management, SCEP service, and son.
oracle.idm.msm.service	Logs events related to all the services which are exposed to the external client. This also includes the ReST service implementations.

13.3.3 About Logging Message Types and Levels

The amount of data output by a logger is controlled by its level; the higher the level, the more information is logged. You can specify logging message types and levels in ODL. ODL recognizes five message types: `INCIDENT_ERROR`, `ERROR`, `WARNING`, `NOTIFICATION`, and `TRACE`. Each message type can also take a numeric value between `1` (highest severity) and `32` (lowest severity) that you can use to further restrict message output. When you specify a message type, ODL returns all messages of that type, as well as the messages that have a higher severity. For example, if you set the message type to `WARNING`, `ODL` also returns messages of type `INCIDENT_ERROR` and `ERROR`.

Message types and levels are described in greater detail in "Setting the Level of Information Written to Log Files" in *Oracle Fusion Middleware Administrator's Guide*.

Table 13–3 ODL Message Types and Levels

Message Type and Numeric Value	Description
INCIDENT_ERROR:1	A serious problem that may be caused by a bug in the product and that should be reported to Oracle Support. Examples are errors from which you cannot recover.
ERROR:1	A serious problem that requires immediate attention from the administrator and is not caused by a bug in the product. An example is if Oracle Fusion Middleware cannot process a log file, then you can correct the problem by fixing the permissions on the document.

Table 13–3 (Cont.) ODL Message Types and Levels

Message Type and Numeric Value	Description
WARNING:1	A potential problem that should be reviewed by the administrator. Examples are invalid parameter values or a specified file does not exist.
NOTIFICATION:1	A major lifecycle event such as the activation or deactivation of a primary sub-component or feature. This is the default level for `NOTIFICATION`.
NOTIFICATION:16	A finer level of granularity for reporting normal events.
TRACE:1	Trace or debug information for events that are meaningful to administrators, such as public API entry or exit points.
TRACE:16	Detailed trace or debug information that can help Oracle Support diagnose problems with a particular subsystem.
TRACE:32	Very detailed trace or debug information that can help Oracle Support diagnose problems with a particular subsystem.

13.3.4 Configuring the Loggers and Log Handlers Using Fusion Middleware Control

Oracle Fusion Middleware components generate log files containing messages that record all types of events. Administrators can set logging levels and create new log handlers for Oracle Mobile Security Manager using Fusion Middleware Control, as described in this section.

If you want to configure loggers and log handlers through editing the logging.xml file instead of using Fusion Middleware Control, see Section 13.3.5, "Configuring the Loggers and Log Handlers Using the logging.xml File."

Accessing Logging Configuration

To access the logging configuration using Oracle Fusion Middleware Control:

1. Log in to Oracle Fusion Middleware Control.

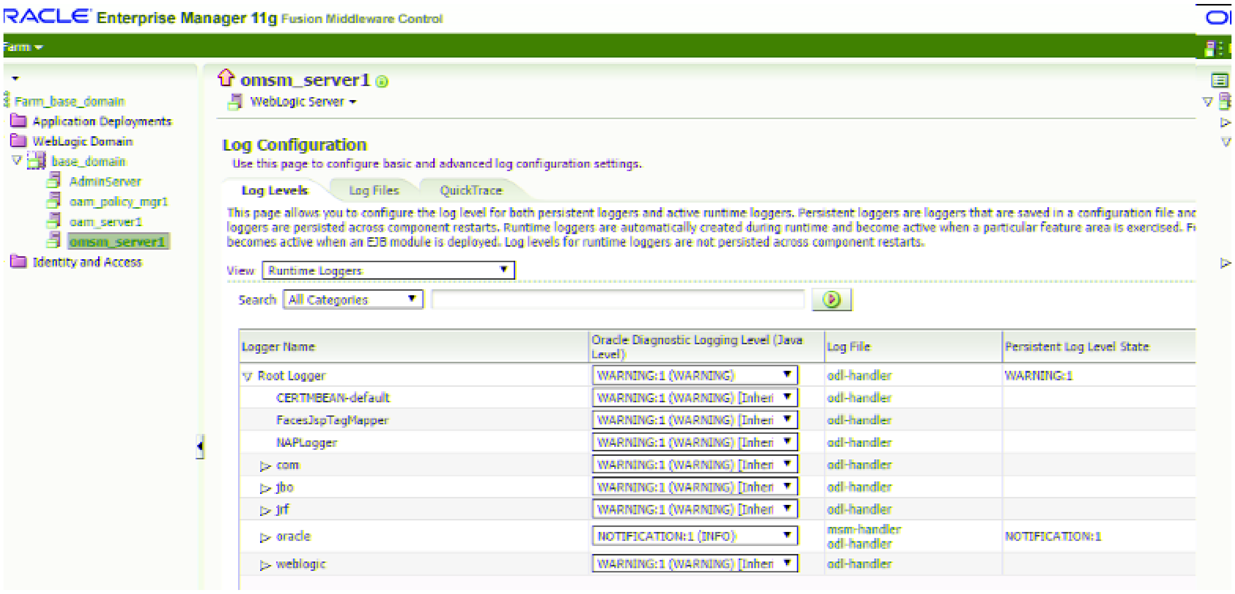
   For information on logging in to Oracle Fusion Middleware Control, see "Displaying Fusion Middleware Control" in *Oracle Fusion Middleware Administrator's Guide*.

2. From the navigation pane, expand the farm, then WebLogic Domain, and then navigate to your domain.

3. Right-click your domain and select **Logs**, and then, **Log Configuration**.

   The Log Configuration page is displayed. All the packages available for logging are displayed on the log configuration screen, as shown in Figure 13–1.

Figure 13–1 Oracle Mobile Security Manager Log Configuration in Oracle Fusion Middleware Control



Log Configuration

Use this page to configure basic and advanced log configuration settings.

This page allows you to configure the log level for both persistent loggers and active runtime loggers. Persistent loggers are loggers that are saved in a configuration file anc loggers are persisted across component restarts. Runtime loggers are automatically created during runtime and become active when a particular feature area is exercised. Fı becomes active when an EJB module is deployed. Log levels for runtime loggers are not persisted across component restarts.

View: Runtime Loggers

Search: All Categories

Logger Name	Oracle Diagnostic Logging Level (Java Level)	Log File	Persistent Log Level State
Root Logger	WARNING:1 (WARNING)	odl-handler	WARNING:1
CERTMBEAN-default	WARNING:1 (WARNING) [Inheri	odl-handler	
FacesJspTagMapper	WARNING:1 (WARNING) [Inheri	odl-handler	
NAPLogger	WARNING:1 (WARNING) [Inheri	odl-handler	
com	WARNING:1 (WARNING) [Inheri	odl-handler	
jbo	WARNING:1 (WARNING) [Inheri	odl-handler	
jrf	WARNING:1 (WARNING) [Inheri	odl-handler	
oracle	NOTIFICATION:1 (INFO)	msm-handler odl-handler	NOTIFICATION:1
weblogic	WARNING:1 (WARNING) [Inheri	odl-handler	

The packages specific to Oracle Mobile Security Manager can be accessed under `oracle.idm.msm.`

Configuring Logging Level

The different log levels are available for selection under the Oracle Diagnostic Logging Level column.

To configure the Oracle Diagnostic Logging level:

1. Click the **Log Levels** tab.
2. In the Diagnostic Logging Level column, choose the logging level for the corresponding logger.
3. Click Apply to submit and apply log level configuration changes, which take affect immediately.

Creating New Log Handlers

In addition, you can create and configure new log handlers using the **Log Files** tab.

1. Click the **Log Files** tab.
2. Click the **Create** button to display a fresh **Create Log File** form.
3. Enter a name and file system path for this log file.
4. Click the desired Log File Format. For example: `... Text.`
5. Set the logging attributes.
6. Associate a Logger.
7. Specify the Rotation Policy.
8. Click **OK** to submit the configuration.

If you want to create a log handler like an existing one, click the **Create Like** button.

13.3.5 Configuring the Loggers and Log Handlers Using the logging.xml File

You can configure loggers and create log handlers in Fusion Middleware Control. However, you can also edit the `logging.xml` file to configure loggers and log handlers, as described here.

The `logging.xml` file is located in the following directory:

```
DOMAIN_NAME/config/fmwconfig/servers/SERVER_NAME/logging.xml
```

DOMAIN_NAME and *SERVER_NAME* are the domain name and server name, respectively, specified during the installation of Oracle Mobile Security Manager.

13.3.5.1 Logger and Log Handler Configuration Example

The `logging.xml` file has a `<log_handlers>` configuration section, followed by a `<loggers>` configuration section. Each log handler is defined within the `<log_handlers>` section, and each logger is defined within the `<loggers>` section.

The file has the following basic structure:

```
<logging configuration>
<log_handlers>
     <log_handler name='console-handler' level="NOTIFICATION:16"></log_handler>
     <log_handler name='odl-handler'></log_handler>
     <!--Additional log_handler elements defined here....-->
</log_handlers>
<loggers>
     <logger name='' level='WARNING:1'>
          <handler name='odl-handler'/>
          <handler name='wls-domain'/>
          <handler name='console-handler'/>
     </logger>
     <logger name="example.logger.one" level="NOTIFICATION:16">
          <handler name="console-handler"/>
     </logger>
     <logger name="example.logger.two" />
     <logger name="example.logger.three" />
     <!--Additional logger elements defined here....-->
</loggers>
</logging_configuration>
```

When configuring a logger to write messages to either the console or a file, make configuration changes to both the logger and the handler. Setting the level attribute for the logger configures the amount of detail (and therefore, the volume of messages) that the logger sends to the handler. Similarly, setting the level attribute for the handler configures the amount of detail that the handler accepts from the logger.

13.3.5.2 Configuring Log Handler Level

You configure individual log handlers in the `<log_handlers>` section of the `logging.xml` file. Configure the level attribute for the handler to set the amount of detail that the handler will accept from loggers.

To configure the log handler-level attribute

1. Open the `DOMAIN_NAME/config/fmwconfig/servers/SERVER_NAME/logging.xml` file.
2. Change the level attribute as shown in the following examples.

In this example XML code, the level attribute for the omsm-handler is set to WARNING:32:

```
<log_handler name='omsm-handler'
class='oracle.core.ojdl.logging.ODLHandlerFactory'
filter='oracle.dfw.incident.IncidentDetectionLogFilter' level='WARNING:32' >
```

For the omsm-handler to be able to write TRACE level messages to the console, change the level attribute as shown:

```
<log_handler name='omsm-handler'
class='oracle.core.ojdl.logging.ODLHandlerFactory'
filter='oracle.dfw.incident.IncidentDetectionLogFilter' level='TRACE:1' >
```

3. Save your changes and restart the application server.

13.3.5.3 Configuring Additional Settings for Log Handlers

You can configure additional properties for log handlers that write to a file. For example, this excerpt from logging.xml configures the omsm-handler:

```
<log_handler name='omsm-handler'
class='oracle.core.ojdl.logging.ODLHandlerFactory'
filter='oracle.dfw.incident.IncidentDetectionLogFilter'>
    <property name='path'
value='${domain.home}/servers/${weblogic.Name}/logs/${weblogic.Name}-diagnostic.lo
g'/>
    <property name='maxFileSize' value='10485760'/>
    <property name='maxLogSize' value='104857600'/>
    <property name='encoding' value='UTF-8'/>
    <property name='useThreadName' value='true'/>
   <property name='supplementalAttributes' value='J2EE_APP.name,J2EE_MODULE.name,
WEBSERVICE.name,WEBSERVICE_PORT.name,composite_instance_id,component_instance_id,
composite_name,component_name'/>
</log_handler>
```

See "Configuring Settings for Log Files" in the *Oracle Fusion Middleware Administrator's Guide* for information about both the Fusion Middleware Control tool and the WLST command-line tool

See "Logging Custom WLST Commands" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference* for information about the WLST command-line tool

13.3.5.4 Configuring Loggers

Individual loggers are configured in the <loggers> section of the logging.xml file. Oracle Mobile Security Manager has eight loggers that can be configured to send messages to log handlers. Setting the level attribute for the logger configures the amount of detail (and, hence, the volume of messages) that the logger sends to its handlers. Nesting one or more <handler> elements inside of <logger> elements assigns handlers to loggers. The following excerpt shows a logger called oracle.idm.msm.service. The level attribute is set to TRACE:32 and the logger sends messages to two handlers:

```
<logger name='oracle.idm.msm.service' level='TRACE:32' useParentHandlers='false'>
   <handler name='odl-handler'/>
   <handler name='omsm-handler'/>
  </logger>
```

A logger can inherit a parent logger's settings, including the parent's level setting and other attributes, as well as the parent logger's handlers. To disable inheritance, set the `useParentHandlers` attribute to `false`, as shown in the previous excerpt.

At the top of the logger inheritance tree is the root logger. The root logger is the logger with an empty name attribute, as shown in the following example.

```
 <loggers>
   <logger name="" level="WARNING:1">
      <handler name="odl-handler"/>
      <handler name="wls-domain"/>
      <handler name="console-handler"/>
    </logger>

    <!-- Additional loggers listed here -->
</loggers>
```

If a logger is configured with only its name attribute, the logger will inherit the rest of its attributes from the root logger, as shown in the following example:

```
<loggers>
    <logger name="oracle.idm.msm.service"/>
    <!-- Additional loggers listed here -->
</loggers>
```

To configure loggers:

1. Open the `*DOMAIN_NAME*/config/fmwconfig/servers/*SERVER_NAME*/logging.xml` file.
2. Locate the logger you want to configure. The lists of Oracle Mobile Security Manager loggers is at.
3. Define the level attribute for the `<logger>` element. See the example at the beginning of this section.
4. Add one or more `<handler>` elements to the `<logger>` element.
5. When you are finished editing both the `<loggers>` and `<log_handlers>` sections of `logging.xml`, save the file.
6. Restart the application server for the changes to take effect.

13.3.5.5 Sample Oracle Mobile Security Manager Log Output

The following ODL log excerpt illustrates the kind of output you can expect in Oracle Mobile Security Manager.

```
[2015-03-10T14:29:38.713-07:00] [omsm_server1] [ERROR] [MSM-00906]
[oracle.idm.msm.identity] Identity Service exception.

[2015-03-12T01:57:03.391-07:00] [omsm_server1] [TRACE] [] [oracle.idm.msm.policy]
[SRC_CLASS: oracle.idm.msm.policy.impl.PolicyManagerImpl] [SRC_METHOD:
getMobilePolicy] Error while Getting Policy:Default Policy_
DeviceProvisioningPolicy

[2015-03-12T01:57:03.315-07:00] [omsm_server1] [TRACE:16] []
[oracle.idm.msm.policy] [SRC_CLASS:
oracle.idm.msm.policy.impl.spi.oes.PolicyManagerSpiImpl] [SRC_METHOD:
getPolicyNames] ENTRY ( ( "MOBILE_POLICY_NAME" eq "policy" ) or ( "POLICY_
DESCRIPTION" eq "policy" ) or ( "POLICY_GROUP" eq "policy" ) )
```

13.4 Troubleshooting Common Oracle Mobile Security Suite Deployment Issues

This section lists common issues that may occur when deploying the Oracle Mobile Security Suite. They are as follows:

- Ports are not opened (incoming and outgoing from Mobile Security Access Server).
- Ports are blocked either by an external firewall, an application firewall, or `iptables` configuration
- Host names are not resolving or there are DNS problems
- Users do not know how to obtain server certificates
- Mobile Security Access Server certificates are not trusted on mobile device
- Clock on the Mobile Security Access Server and the Windows domain controller are out of synchronization
- Users entered `UID` used instead of the full `UPN`
- Kerberos Service Principal Names (SPNs) are not properly defined for back-end applications
- Workspace and containerized applications signed with different certificates
- When building a workspace static library project for iOS, a simulator or connected devices were selected as a build target instead of a generic iOS device
- Clock on the Mobile Security Access Server and the mobile device are out of synchronization
- Trying to perform PKINIT with older Windows version, prior to Win2k8 R2
- Incorrect user certificate templates were used
- Incorrect web settings sent everything directly or blocked everything
- Missing trust relationships were in multi-domain environments
- Use of alternate UPN suffixes were used with KINIT (requires configuration change)
- Some back-end servers were running old SSL stacks that cannot handle newer ciphers reported by the Mobile Security Access Server (requires configuration change)
- Old or low-end Android devices
- Invite button is disabled if user does not have an email address in the LDAP directory.

13.5 Troubleshooting Missing LDAP Group in the List of Roles Applicable to the User

When a user accesses the Oracle Access Management Access Manager console, the Domain Users LDAP group does not appear in the list of roles applicable to the user. The Domain Users role does appear in the role list, but does not have many users associated with it.

The Domain Users LDAP group does not appear because Domain Users is a special group which is part of the User's security attributes and not part of the memberOf attribute in AD which provides information on group memberships. The primaryGroupID attribute is a single-valued attribute that contains the primaryGroupToken of the group that is the primary group of the object. The primary group of the object is not included in the memberOf attribute. For example, by default, the primary group of a user object is the primaryGroupToken of the Domain Users group, but the Domain Users group is not part of the user object's memberOf attribute.

Refer to https://msdn.microsoft.com/en-us/library/ms677943.aspx for information on memberOf and primaryGroupId.

To work around this issue, create a separate LDAP group and make the users members of that group.

13.6 Troubleshooting the Mobile Security Access Server Issues

This section describes troubleshooting tips for Oracle Mobile Security Access Server. It contains the following topics:

- Troubleshooting Mobile Security Access Server SSL-Related Issues
- Troubleshooting Mobile Security Access Server Connection Errors

13.6.1 Troubleshooting Mobile Security Access Server SSL-Related Issues

If you are experiencing Mobile Security Access Server problems related to Secure Sockets Layer (SSL), the following tips may help you with the issues:

- Ensure that the Mobile Security Access Server certificate and certificate chain are not revoked.
- If the Mobile Security Access Server certificate contains a Subject Alternate Name, ensure that the server certificate subject name is also in the Subject Alternate Name (SAN) attribute of the certificate.
- Ensure that all subject names/subject alternative names are resolved by DNS.
- Ensure that the certificate chain file is built with the issuing CA, followed by zero or more intermediate CA(s) and then the Root CA.

13.6.2 Troubleshooting Mobile Security Access Server Connection Errors

If you are experiencing Mobile Security Access Server issues related to internet connection, the following tips may help you with the issues:

- Ensure that the mobile device is online/connected to the internet.
- Ensure that the Mobile Security Access Server is running.
- Ensure that all necessary Mobile Security Access Server ports are open on the host firewall or other firewalls in the route from the mobile device to the Mobile Security Access Server.
- Ensure that no other services are using the configured Mobile Security Access Server ports.
- Ensure that the applications behind Mobile Security Access Server are accessible from the Mobile Security Access Server.
- Ensure that the Mobile Security Access Server can resolve in DNS the host names of the applications behind Mobile Security Access Server.

- If the connection error occurs for only a single application behind the Mobile Security Access Server, ensure that the application is running.
- Ensure that there are no systems in the network blocking proxy traffic if a firewall or reverse proxy is configured.

13.7 Troubleshooting Kerberos-Enabled Applications Issues

This section lists tips for troubleshooting Kerberos-enabled applications. The troubleshooting tips are as follows:

> **Note:** Normal requests come in as `*scheme*://*hostname*:*port*/*path*` and the resulting Service Principal Name (SPN) is `HTTP/*hostname*`.
>
> MSM requests come in as *scheme*://*mfm_hostname*:*port*/*fileserver_hostname*:*port*/*share_path*. The Service Principal Name (SPN) needs to be set to `HTTP/`*fileserver_hostname*.
>
> There is also the form where the request comes in as *scheme*://*mfm_hostname*:*port*/`mfm`/*fileserver_hostname*:*port*/*share_path*. The Service Principal Name (SPN) still needs to be set to `HTTP/`*fileserver_hostname*.

1. You must configure Web applications that are accessed through the Mobile Security Access Server for Kerberos with a Service Principal Name (SPN) for each application server that is accessed by an alias instead of its host name.

   For example, if the hostname is `sp1.oracle.internal` but it is accessed as `HTTP/sharepoint.oracle.internal`, the SPN must be set as `http://sharepoint.` Additional certificate requirements apply for the Mobile Security Access Server certificate.

   From a machine within the domain of the application server (Mobile Security Access Server can be used if it is joined to the same domain), perform the following steps:

   a. Open a command window.

   b. At the command-line prompt, type:

   ```
   setspn  -l customer_application_hostname
   ```

   c. Verify an SPN exists for the URL that the device is trying to access.

   d. If the SPN is missing, then type:

   ```
   setspn -a customer_application_hostname
   ```

   e. Verify the SPN by typing:

   ```
   setspn  -l customer_application_hostname
   ```

2. Configure Internet Information Services (IIS) applications such as Microsoft SharePoint for Negotiate authentication. It can be followed by NTLM authentication if desired.

3. Internet Information Services (IIS) applications use an application pool with an application-pool identity. This pool cannot be a local account on the web server. Typically, it can be set to a built-in account of `NETWORK` that has permission to access the Active Directory for authentication.

When a service account is used for the pool identity, ensure that the account has permission to access and authenticate to Active Directory.

a. Ensure that the authentication provider is set to `Negotiate`.

b. Ensure that Windows authentication is set.

c. Ensure that Anonymous User is not set.

Note: The following commands are useful to debug network issues with Wireshark:

1. In display filter, type:

   ```
   kerberos
   ```

2. In display filter, type:

   ```
   ntlmssp
   ```

3. In display filter, type:

   ```
   http
   ```

13.8 Troubleshooting Mobile Devices Access Issues

This section describes how to troubleshoot mobile device access issues. It contains the following topics:

- Troubleshooting General Mobile Device Access Issues
- Troubleshooting Mobile Security Access Server SSL-Related Issues
- Turning on Client Debug Logs
- About the Normal Sequence of Request for Registration and Authentication

13.8.1 Troubleshooting General Mobile Device Access Issues

The following are some general tips for troubleshooting mobile device access issues:

1. Ensure that the Mobile Security Access Server host name can be resolved by the mobile device.

2. Ensure that the Proxy Auto Configuration (PAC) files are accessible from the URL location specified in the mobile configuration.

3. Ensure that the Mobile Security Access Server host name as specified in the PAC files can be resolved by DNS.

4. Ensure that the Mobile Security Access Server name matches the PROXY statement in the PAC file on the Oracle Mobile Security Access Server.

5. If the mobile device is configured for Wi-Fi, ensure that the proxy with the URL of the `bmax.pac` file is specified.

6. If the mobile device is configured for VPN, ensure that the proxy with the URL of `bmax.pac` file is specified on the VPN and is not needed in the Wi-Fi configuration.

7. Ensure that the Mobile Security Access Server configuration files, `bmconfig_*.json`, are correctly configured in the Mobile Security Workspace application settings.

8. If the Mobile Security Access Server is configured for PKINIT or KINIT authentication, ensure that the user account being used is not locked in Active Directory.
9. If the Mobile Security Access Server is configured for Oracle Access Manager authentication, ensure that the user account is not locked.
10. If the Mobile Security Access Server is configured for PKINIT authentication, ensure that client certificates have the correct attributes for mutual authentication and smart-card login.
11. If the Mobile Security Access Server is configured for PKINIT authentication, ensure that the CA certificate chain for the Mobile Security Access Server certificate is installed in the mobile device key chain (network profiles).

13.8.2 Troubleshooting Mobile Device Access SSL-Related Issues

See Section 13.6.1, "Troubleshooting Mobile Security Access Server SSL-Related Issues."

13.8.3 Turning on Client Debug Logs

Navigate to the Mobile Security Workspace application settings. Turn on Log Mode and set Log Level to `Verbose` or `Debug`.

13.8.4 About the Normal Sequence of Request for Registration and Authentication

During the normal registration process for a Mobile Security Workspace, the following sequence of requests are sent:

1. A sequence of requests to an authentication URL, with the expected response of `HTTP 407`. The number of requests may be different depending on the authentication method being used. The UPN or user ID of the authenticating user should appear at the beginning of the line associated with the authentication request. These requests occur every authentication.
2. A final request to an authentication URL, with the expected response of `HTTP 200` or `HTTP 302`. A response of `HTTP 200` means that the authentication was initiated directly within the Mobile Security Workspace, while a response of `HTTP 302` means that the authentication was initiated by a redirect from an external application such as the Safari web browser or a containerized application. If an `HTTP 403` is returned in response to any authentication request, it means that the authentication failed. This request occurs every authentication.
3. A request to `/action`. This request sets up the workspace for offline authentication and PIN/Password reset. This request only occurs during registration and PIN/Password resets.
4. A request to `/ecp/ecpservice/registercontainer`. This request registers the workspace with the Mobile Security Manager server. The expected response is `HTTP 200`. This request only occurs during registration.
5. A request to `/ecp/ecpservice/policy/get`. This is a request to retrieve the policies associated with the workspace. The expected response is `HTTP 200`. This request occurs every authentication.
6. A request to `/ecp/ecpservice/settings/get`. This is a request to retrieve the company settings. The expected response is `HTTP 200`. This request occurs every authentication.

7. A request to /ecp/ecpservice/getcommands. This is a request to retrieve pending commands for the workspace. The expected response is HTTP 200. This request occurs periodically after registration.

Depending on the deployment configuration, there will also be a number of requests to the bmax.pac file and stunnel.pac. If there are no requests to the stunnel.pac or bmax.pac it likely means that the Mobile Security Access Server is not accessible from the mobile device.

13.9 Troubleshooting the Microsoft Exchange Notification Integration Issues

This section describes troubleshooting tips for the integration with Microsoft Exchange notifications.

If notifications are not being sent to clients, verify the following:

- The Exchange configuration matches the version of Exchange.
- The permission of the proxy account to act on behalf of users is configured in Exchange. This permission is different for Microsoft Exchange Server 2007 than for Exchange Server 2010.

Details for the two versions are described in the following topics:

- Exchange Impersonation on Exchange 2007
- Exchange Impersonation on Exchange 2010

13.9.1 Exchange Impersonation on Exchange 2007

Microsoft Exchange Server 2007 requires two rights for Exchange Impersonation to work:

- *ms-Exch-EPI-Impersonation: This right is applied to the Client Access Server and grants the Service Account permission to function as an Exchange Impersonation account on that CAS.
- oms-Exch-EPI-May-Impersonate: This right is applied on either a user-by-user basis for each of the users that require impersonation to be enabled, or it can be applied on a mailbox database.

For example, Joe Client is a user with a device, and EWS Proxy is the account used to impersonate the user for notification.

To add these rights:

```
Add-ADPermission
   -Identity (Get-ExchangeServer
   -IdentityYOUR_   CAS).DistinguishedName
   -User (Get-User -Identity "EWS Proxy").Identity
   -extendedRight ms-Exch-EPI-Impersonation

Add-ADPermission
   -Identity (Get-User -Identity "Joe Client").DistinguishedName
   -User (Get-User -Identity "EWS Proxy").Identity
   -extendedRight ms-Exch-EPI-May-Impersonate
```

13.9.2 Exchange Impersonation on Exchange 2010

Exchange 2010 requires rights for Exchange Impersonation to work.

To configure Exchange Impersonation for specific groups of users:

- Ensure that the AD users are placed in the group (for example, Notification Users) and the service account has a Management scope for the AD group that Exchange is recognized. For example:

  ```
   New-ManagementScope -Name:"ExchImpersonationScope" -RecipientRestrictionFilter
  {memberofgroup -eq "CN=BNS Users,OU=QA,DC=bitzermobile,DC=com"}
  ```

- Define Assign Role:

  ```
  New-ManagementRoleAssignment -Name:"ExchImpersonationRole"
  -Role:ApplicationImpersonation -User:"ewsproxy@example.com"
  -CustomRecipientWriteScope:"ExchImpersonationScope
  ```

If you receive the following message, you may have configured an incorrect Exchange version for the Exchange notification integration:

```
microsoft.exchange.webservices.data.ServiceVersionException: Method
SubscribeToPushNotificationsOnAllFolders is only valid for Exchange Server version
Exchange2010 or later
```

If there are no errors it may be caused by Apple Push Notification server, as it does not guarantee for delivery. Verify your Apple credentials provided are valid.

13.10 Troubleshooting Mobile File Manager Connection Issues

The Mobile File Manager will fail to connect if a Windows file share on any of the following Windows versions is being referenced by a DNS alias instead of the native system host name:

- Windows Server 2012
- Windows Server 2008
- Windows 8
- Windows 7
- Windows Vista

To fix the issue, either access the file share using the native host name, or complete the following steps to modify the registry on the file share server:

1. Locate and click the following key in the registry of the server:

   ```
   HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\LanmanServer\Parameters
   ```

2. On the **Edit** menu, click **Add Value**, and then add the following registry value:

   Value name: DisableStrictNameChecking

   Data type: REG_DWORD

   Radix: Decimal

   Value: 1

3. Restart the Windows Server service on the file share server.