Managing Devices with XenMobile MDM
Contents

Managing Devices with XenMobile MDM ................................................................. 21
  Device Manager ................................................................................................. 23
  Device Manager 8.5 ............................................................................................ 24
    Device Manager 8.5 ......................................................................................... 25
      About This Release ....................................................................................... 26
        What's New ................................................................................................. 27
        Feature Support by Device Platform ......................................................... 31
        Known Issues .............................................................................................. 33
      Device Manager System Requirements ......................................................... 35
      Preparing for Device Manager Installation ................................................... 38
      Device Manager Remote Support Requirements ............................................. 41
    Licensing ........................................................................................................ 42
      Renewing Device Manager Licenses ............................................................... 44
  Installing Device Manager Server ................................................................. 45
    Device Manager Pre-Installation Checklist ....................................................... 46
    Deploying XenMobile MDM - Device Manager ............................................. 48
      Choosing Device Manager Components to Install ....................................... 50
      To install Device Manager .......................................................................... 51
    Upgrading Device Manager to Version 8.5 ...................................................... 58
      Backing Up and Restoring Device Manager .................................................. 59
        To perform a full manual backup of Device Manager server ...................... 60
        To perform a directory and native SQL backup of Device Manager server .... 61
      Installing Patches for Device Manager ......................................................... 62
    Configuring High Availability on Device Manager ........................................... 63
      To install Device Manager on cluster node 1 ............................................... 65
      To install Device Manager on cluster node 2 ............................................... 66
      To configure a Device Manager Tomcat Cluster .......................................... 67
      To configure the Device Manager Server .................................................... 69
To configure Tomcat oscache.properties 

To run update-hilo-sql on all databases besides MS-SQL

To overwrite the .pem file

To start the Device Manager windows service

To test the cluster setup

Requesting an APNS Certificate

Apple MDM Push Certificate Migration Information

To create a CSR by using Microsoft IIS

To create a CSR on a Macintosh computer

To create a CSR by using OpenSSL

To submit the CSR to Citrix for signing

To submit the signed CSR to Apple to obtain the APNS certificate

To create a .pfx APNS certificate by using Microsoft IIS

To create a .pfx APNS certificate on a Macintosh computer

To create a .pfx APNS certificate by using OpenSSL

To import an APNS certificate into Device Manager

To renew an APNS certificate

Enrolling User Names with Special Characters

To enable two step enrollment for iOS devices

To enroll an iOS device in Device Manager

To unenroll an iOS device from Device Manager

To enroll your Android device in Device Manager

To unenroll your Android device from Device Manager

To enroll a Windows Phone 8 device

To enroll Windows 8 devices in Device Manager

To enroll a Windows Mobile device in Device Manager

To enroll a Symbian device in Device Manager

Configuring Enrollment Modes

To send enrollment notifications to iOS or Android devices

To configure a Notifications SMTP Server

To configure an SMS Notifications Gateway

To configure a notification template

Enrolling User Names with Special Characters

To enable two step enrollment for iOS devices

To enable auto discovery for client enrollment
To create a deployment policy for iOS apps ................................. 169
To push applications to iOS devices ........................................ 170
To remove an app from an iOS device ...................................... 172
Distributing iOS Volume Purchase Program Apps .................... 173
  To distribute Volume Purchase Program apps ....................... 174
  To run a code inventory report on Volume Purchase Program apps ................................................. 176
To update new versions of custom iOS apps ............................ 177
Adding Android Apps .......................................................... 178
  To add an internal Android app ....................................... 179
  To add an external Android app ..................................... 180
Citrix Worx Store for Android Apps (MDM-only) .................... 181
  To add Android apps to Device Manager.......................... 182
  To brand the Worx Store for Android ............................. 184
  To create a deployment package for Android apps ............... 185
Adding Windows Phone 8 Apps .............................................. 186
  To add an internal Windows Phone 8 app .......................... 187
  To add an external Windows Phone 8 app .......................... 188
  Configuring Apps for Windows Phone 8 Worx Home ............. 189
Adding Windows Mobile Apps ................................................ 190
Configuring Applications Access Policies .............................. 191
  To configure an Applications Access Policy ....................... 193
Application Tunnels ............................................................ 194
  To add an Application Tunnel ....................................... 195
  To update or delete an App Tunnel .................................. 197
  To create a remote support App Tunnel ............................ 198
Adding Files ........................................................................ 199
  To add a file to Device Manager ..................................... 200
Uploading Other Files .......................................................... 201
Configuring Macro Substitution for Uploading Files ............... 202
  Examples of Simple Mortscripts ................................. 206
  To upload a MortScript file ......................................... 207
Creating Policies .................................................................. 208
Scheduling Connections to Device Manager ............................ 209
Managing SharePoint Configurations .................................... 210
  To configure a SharePoint resource configuration site in Device Manager ........................................ 211
  To configure a SharePoint document control policy .......... 212
Configuring SharePoint on Android Devices 214
Managing iOS Configurations ........................................ 218
  To configure iOS security policies ............................ 219
  To configure network policies for iOS .............. 223
  To create an app lock policy for iOS 6 devices 224
  To create a profile removal policy for iOS 225
  To configure automatic profile removal for iOS 6 devices 226
  To configure geo-tracking on iOS devices 227
  To create a geo-fencing policy for iOS devices 229
  To store iOS user password ................................ 231
  To place a device in supervised mode by using the Apple Configurator ........................................ 232
Managing Android Configurations ................................. 233
  Configuring Policies for Android Devices.................. 234
    To configure basic options for Android devices ........ 235
    To uninstall Device Manager on an Android device 236
    To create an Android credential policy ......... 237
    To configure a policy to schedule connections for Android devices ........................................ 238
  How Scheduling Policies Affect Android Battery Usage 240
    To define password requirements and enable encryption on Android device .............................. 242
    To configure WiFi settings for Android devices失望 243
    To declare a GPRS access point in an Android device 244
  Configuring App Monitoring for Android Apps ................. 245
    Example XML Syntax for Blacklisting and Whitelisting Policies ............................................. 247
    To add a logo to a customized block screen on an Android device ................................. 248
    Common Issues with the App Monitoring Policy Implementation ............................................ 249
  Configuring Touchdown for Android Devices .................. 250
    Policy Combinations for Touchdown on Android Devices ...................................................... 251
    XenMobile-Certified TouchDown Policies .................. 252
    To configure a TouchDown policy to install and configure Exchange email accounts ........ 254
    To create a deployment package for the TouchDown policy .......................................................... 255
    To initiate a selective wipe of email data by using a TouchDown API ...................................... 256
    Configuring Deployment Rules for an Android Device Size ......................................................... 257
List of TouchDown Policies for Android Devices .......................................................... 258
List of TouchDown Application Settings for Android Devices ........................................... 265
To update a new version of a custom Android app ...................................................... 274
Configuring Exchange ActiveSync Policies for HTC devices ........................................... 275
Configuring Policies for Samsung SAFE Devices ............................................................ 276
Managing Samsung Knox Configurations ......................................................................... 279
To create a Samsung Knox Exchange ActiveSync configuration ......................................... 280
To create a Samsung Knox password policy ...................................................................... 281
To create a Samsung Knox browser configuration ............................................................. 282
To create a Samsung Knox enterprise VPN configuration ................................................ 283
To create a Samsung Knox app restriction policy ............................................................... 285
To create a Samsung Knox app uninstall configuration ...................................................... 286
To create a Samsung Knox Remote Support policy ............................................................ 287
To perform a selective wipe on a Samsung Knox device ................................................... 288
Managing Windows Phone 8 Configurations .................................................................... 289
To create an Enterprise Hub policy for Windows Phone 8 devices ..................................... 290
To access apps on the Windows Phone 8 Work Home App Store ........................................ 291
To create a Windows Phone 8 app deployment package .................................................... 292
To perform a selective wipe on a Windows Phone 8 device ............................................... 293
To configure Windows Phone 8 Exchange ActiveSync policies ......................................... 294
To configure Windows Phone 8 password policies ............................................................. 295
To create Windows Phone 8 storage policies ................................................................. 297
To configure Windows Phone 8 custom policies .............................................................. 298
Creating Windows 8 Tablet Registry Configurations ......................................................... 299
To create a Windows 8 tablet registry configuration .......................................................... 300
Managing Windows Mobile Configurations ...................................................................... 301
About the Windows Mobile Server Collection ................................................................. 302
To add a new server in the Windows Mobile server collection ............................................. 303
To edit or delete a server ................................................................................................. 304
To configure device IP address ranges ............................................................................ 305
About the Registry Collection for Windows Mobile ............................................................ 306
Configuring Registry Keys by Using the Device Manager Options Wizard .......................... 307
To uninstall Device Manager from a Windows Mobile device ............................................. 309
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>To create a certificate template for XenMobile certificate requests</td>
<td>402</td>
</tr>
<tr>
<td>To generate the XenMobile client certificate</td>
<td>403</td>
</tr>
<tr>
<td>To export the client certificate</td>
<td>404</td>
</tr>
<tr>
<td>To configure a Microsoft certificate server entity</td>
<td>405</td>
</tr>
<tr>
<td>To configure a Microsoft certificate services policy</td>
<td>406</td>
</tr>
<tr>
<td>Configuring an OpenTrust PKI Adapter for Device Manager</td>
<td>408</td>
</tr>
<tr>
<td>To obtain an authentication certificate from OpenTrust PKI</td>
<td>409</td>
</tr>
<tr>
<td>To set up access rights on OpenTrust PKI</td>
<td>410</td>
</tr>
<tr>
<td>To configure the OpenTrust PKI</td>
<td>411</td>
</tr>
<tr>
<td>To set the connection to the adapter</td>
<td>412</td>
</tr>
<tr>
<td>To configure an iOS profile to deliver certificates to iOS devices</td>
<td>413</td>
</tr>
<tr>
<td>To configure an OpenTrust adapter to use HTTP by using a self-signed certificate</td>
<td>415</td>
</tr>
<tr>
<td>To configure Device Manager to generate identity certificates from OpenTrust adapter</td>
<td>417</td>
</tr>
<tr>
<td>To activate logging on Device Manager for the adapter</td>
<td>426</td>
</tr>
<tr>
<td>Configuring the XenMobile RSA Adapter</td>
<td>427</td>
</tr>
<tr>
<td>XenMobile RSA Adapter Certificate Manager</td>
<td>428</td>
</tr>
<tr>
<td>Requirements</td>
<td>429</td>
</tr>
<tr>
<td>Installing and Configuring the XenMobile RSA Adapter</td>
<td>430</td>
</tr>
<tr>
<td>To install the RSA Adapter on Windows Server</td>
<td>432</td>
</tr>
<tr>
<td>To configure the RSA Adapter in the Device Manager web console</td>
<td>434</td>
</tr>
<tr>
<td>Configuring an MDM Server Trust Certificate on Device Manager</td>
<td>436</td>
</tr>
<tr>
<td>Configuring a Certificate Services Entity by Using XML</td>
<td>438</td>
</tr>
<tr>
<td>Device Manager 8.0.1</td>
<td>440</td>
</tr>
<tr>
<td>About This Release</td>
<td>442</td>
</tr>
<tr>
<td>Key Features</td>
<td>443</td>
</tr>
<tr>
<td>What’s New</td>
<td>446</td>
</tr>
<tr>
<td>Feature Support by Device Platform</td>
<td>450</td>
</tr>
<tr>
<td>Known Issues</td>
<td>452</td>
</tr>
<tr>
<td>System Requirements</td>
<td>454</td>
</tr>
<tr>
<td>Preparing for Device Manager Installation</td>
<td>457</td>
</tr>
<tr>
<td>Device Manager Remote Support Requirements</td>
<td>460</td>
</tr>
<tr>
<td>Plan</td>
<td>461</td>
</tr>
<tr>
<td>Device Manager Pre-Installation Checklist</td>
<td>462</td>
</tr>
<tr>
<td>Topic</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>To push applications to iOS devices</td>
<td>525</td>
</tr>
<tr>
<td>Citrix Worx Store for iOS Apps (MDM-only)</td>
<td>527</td>
</tr>
<tr>
<td>To create an app store for iOS devices</td>
<td>528</td>
</tr>
<tr>
<td>To create a deployment package with the Self-Serve app store for iOS devices</td>
<td>529</td>
</tr>
<tr>
<td>To add apps to Worx Store for iOS</td>
<td>530</td>
</tr>
<tr>
<td>Customizing the Citrix Worx App Store for iOS</td>
<td>531</td>
</tr>
<tr>
<td>To update new versions of custom iOS apps</td>
<td>533</td>
</tr>
<tr>
<td>To remove an app from an iOS device</td>
<td>534</td>
</tr>
<tr>
<td>To create a deployment policy for iOS apps</td>
<td>535</td>
</tr>
<tr>
<td>To create an app lock policy for iOS 6 devices</td>
<td>536</td>
</tr>
<tr>
<td>To configure automatic profile removal for iOS 6 devices</td>
<td>537</td>
</tr>
<tr>
<td>To configure geo-tracking on iOS devices</td>
<td>538</td>
</tr>
<tr>
<td>To create a geo-fencing policy for iOS devices</td>
<td>540</td>
</tr>
<tr>
<td>To create a profile removal policy for iOS</td>
<td>542</td>
</tr>
<tr>
<td>Distributing iOS Volume Purchase Program Apps</td>
<td>543</td>
</tr>
<tr>
<td>To distribute Volume Purchase Program apps</td>
<td>544</td>
</tr>
<tr>
<td>To run a code inventory report on Volume Purchase Program apps</td>
<td>546</td>
</tr>
<tr>
<td>To store iOS user password</td>
<td>547</td>
</tr>
<tr>
<td>To place a device in supervised mode by using the Apple Configurator</td>
<td>548</td>
</tr>
<tr>
<td>Managing Android Configurations</td>
<td>549</td>
</tr>
<tr>
<td>Configuring Policies for Android Devices</td>
<td>550</td>
</tr>
<tr>
<td>To configure basic options for Android devices</td>
<td>551</td>
</tr>
<tr>
<td>To uninstall Device Manager on an Android device</td>
<td>552</td>
</tr>
<tr>
<td>To create an Android credential policy</td>
<td>553</td>
</tr>
<tr>
<td>To configure a policy to schedule connections for Android devices</td>
<td>554</td>
</tr>
<tr>
<td>How Scheduling Policies Affect Android Battery Usage</td>
<td>556</td>
</tr>
<tr>
<td>To define password requirements and enable encryption on Android device</td>
<td>558</td>
</tr>
<tr>
<td>To configure WiFi settings for Android devices</td>
<td>559</td>
</tr>
<tr>
<td>To declare a GPRS access point in an Android device</td>
<td>560</td>
</tr>
<tr>
<td>Configuring App Monitoring for Android Apps</td>
<td>561</td>
</tr>
<tr>
<td>Example XML Syntax for Blacklisting and Whitelisting Policies</td>
<td>563</td>
</tr>
<tr>
<td>To add a logo to a customized block screen on an Android device</td>
<td>564</td>
</tr>
<tr>
<td>Configuring App Monitoring for Android Apps</td>
<td>561</td>
</tr>
<tr>
<td>Example XML Syntax for Blacklisting and Whitelisting Policies</td>
<td>563</td>
</tr>
<tr>
<td>To add a logo to a customized block screen on an Android device</td>
<td>564</td>
</tr>
<tr>
<td>Configuring App Monitoring for Android Apps</td>
<td>561</td>
</tr>
<tr>
<td>Example XML Syntax for Blacklisting and Whitelisting Policies</td>
<td>563</td>
</tr>
<tr>
<td>To add a logo to a customized block screen on an Android device</td>
<td>564</td>
</tr>
<tr>
<td>Common Issues with the App Monitoring Policy Implementation</td>
<td>565</td>
</tr>
<tr>
<td>Topic</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Configuring Touchdown for Android Devices</td>
<td>566</td>
</tr>
<tr>
<td>Policy Combinations for Touchdown on Android Devices</td>
<td>567</td>
</tr>
<tr>
<td>XenMobile-Certified TouchDown Policies</td>
<td>568</td>
</tr>
<tr>
<td>To configure a TouchDown policy to install and configure Exchange email accounts</td>
<td>570</td>
</tr>
<tr>
<td>To create a deployment package for the TouchDown policy</td>
<td>571</td>
</tr>
<tr>
<td>To initiate a selective wipe of email data by using a TouchDown API</td>
<td>572</td>
</tr>
<tr>
<td>Configuring Deployment Rules for an Android Device Size</td>
<td>573</td>
</tr>
<tr>
<td>List of TouchDown Policies for Android Devices</td>
<td>574</td>
</tr>
<tr>
<td>List of TouchDown Application Settings for Android Devices</td>
<td>581</td>
</tr>
<tr>
<td>To update a new version of a custom Android app</td>
<td>590</td>
</tr>
<tr>
<td>Configuring Policies for Samsung SAFE Devices</td>
<td>591</td>
</tr>
<tr>
<td>Configuring Exchange ActiveSync Policies for HTC devices</td>
<td>594</td>
</tr>
<tr>
<td>Citrix Worx Store for Android Apps (MDM-only)</td>
<td>595</td>
</tr>
<tr>
<td>To add Android apps to Device Manager</td>
<td>596</td>
</tr>
<tr>
<td>To create a deployment package for Android apps</td>
<td>598</td>
</tr>
<tr>
<td>Managing Windows Phone 8 Configurations</td>
<td>599</td>
</tr>
<tr>
<td>To create Windows Phone 8 storage policies</td>
<td>600</td>
</tr>
<tr>
<td>To configure Windows Phone 8 password policies</td>
<td>601</td>
</tr>
<tr>
<td>To configure Windows Phone 8 Exchange ActiveSync policies</td>
<td>603</td>
</tr>
<tr>
<td>To configure Windows Phone 8 custom policies</td>
<td>604</td>
</tr>
<tr>
<td>Creating Windows 8 Tablet Registry Configurations</td>
<td>605</td>
</tr>
<tr>
<td>To create a Windows 8 tablet registry configuration</td>
<td>606</td>
</tr>
<tr>
<td>Managing Windows Mobile Configurations</td>
<td>607</td>
</tr>
<tr>
<td>About the Windows Mobile Server Collection</td>
<td>608</td>
</tr>
<tr>
<td>To add a new server in the Windows Mobile server collection</td>
<td>609</td>
</tr>
<tr>
<td>To edit or delete a server</td>
<td>610</td>
</tr>
<tr>
<td>To configure device IP address ranges</td>
<td>611</td>
</tr>
<tr>
<td>About the Registry Collection for Windows Mobile</td>
<td>612</td>
</tr>
<tr>
<td>Configuring Registry Keys by Using the Device Manager Options Wizard</td>
<td>613</td>
</tr>
<tr>
<td>To uninstall Device Manager from a Windows Mobile device</td>
<td>615</td>
</tr>
<tr>
<td>Configuring a Connection to Device Manager on SMS Reception or Call</td>
<td>616</td>
</tr>
</tbody>
</table>
Credential Providers ............................................ 663
Methods of Certificate Issuance 665
Certificate Delivery........................................ 666
Certificate Revocation ................................. 667
Certificate Renewal .................................. 668
To create a credential provider using discretionary
CA entities .................................................. 669
To create a credential provider using external PKI
entities ....................................................... 671
Configuring a SAML Service Provider ................. 675
SAML Test Requirements ................................. 676
To add a SAML service provider ..................... 677
To configure a SAML identity provider 678
Configuring General Security Options .................. 679
Configuring Network Access Controls ................... 681
Configuring Device Manager with Microsoft Certificate
Services..................................................... 682
To enable Web enrollment for Microsoft Certificate
Services ................................................... 683
To enable IIS Web services ............................... 684
To configure Microsoft Internet Information Services for
self-signed or external certificates 685
To configure Microsoft Internet Information Services 686
To create a certificate template for XenMobile
certificate requests ....................................... 687
To generate the XenMobile client certificate 688
To export the client certificate .......................... 689
To configure a Microsoft certificate server entity 690
To configure a Microsoft certificate services policy 691
Configuring an OpenTrust PKI Adapter for Device
Manager ...................................................... 693
To obtain an authentication certificate from
OpenTrust PKI ............................................ 694
To set up access rights on OpenTrust PKI 695
To configure the OpenTrust adapter 696
To set the connection to the adapter 697
To configure an iOS profile to deliver certificates to
iOS devices ................................................ 698
To configure an OpenTrust adapter to use HTTP by
using a self-signed certificate 700
To configure Device Manager to generate identity
certificates from OpenTrust adapter 702
To activate logging on Device Manager for the
adapter ..................................................... 711
Deploying a New Tenant from Multi-Tenant Console ............... 855
Install and Setup .................................................................. 856
Installing PsTools from Microsoft for Multi-Tenant Console 857
To install Multi-Tenant Console ........................................... 859
Upgrading Device Manager MultiTenant Console to version 8.5 860
To reconfigure Java memory size for Multi-Tenant Console 864
To configure an R-DBMS for Multi-Tenant Console .............. 865
To set up Multi-Tenant Console .......................................... 866
To open Multi-Tenant Console ............................................ 867
Configuring Tenant Servers for the Multi-Tenant Console
Deployment ....................................................................... 868
To perform a Multi-Tenant Console Hot Deployment ............. 869
Manage ........................................................................... 871
To import and manage XenMobile application packages 872
To add a new tenant host ..................................................... 873
To add a connector to Multi-Tenant Console ....................... 874
Adding a Tomcat Server in Multi-Tenant Console ............... 875
Declaring a Database Server in Multi-Tenant Console 877
To add a new tenant on Multi-Tenant Console ..................... 878
Configuring Cross-Tenant Reporting with Role Restrictions 880
To provide data manually for cross-tenant reports ............... 881
To set up an initial Multi-Tenant Console partner implementation ................................................ 882
Maintain ........................................................................... 883
To back up your Multi-Tenant Console installation ............... 884
To add or change administrators on Multi-Tenant Console 885
To create a partner on Multi-Tenant Console ....................... 886
To create a user on Multi-Tenant Console......................... 887
To create a role on Multi-Tenant Console......................... 888
To assign a user to a role on Multi-Tenant Console .......... 889
To assign a partner to a tenant on Multi-Tenant Console 890
To view a customer inventory report ................................. 891
Remote Support .................................................................. 892
Remote Support 8.0.1 ......................................................... 893
System Requirements ....................................................... 894
Install and Setup .................................................................. 895
To install the Remote Support Client .................................. 896
To add a remote support device manager connection .......... 897
XenMobile MDM is a robust mobile device management solution that delivers role-based management, configuration, and security for both corporate and employee-owned devices. Upon user device enrollment, IT can provision policies and apps to devices automatically, blacklist or whitelist apps, detect and protect against jailbroken or rooted devices, and wipe or selectively wipe a device that is lost, stolen, or out of compliance. Users can use any device they choose, while IT can ensure compliance of corporate assets and secure corporate content on the device. With XenMobile MDM, you can do the following:

- Configure device settings, email and applications, policies, and device and application restrictions.
- Distribute internally built and externally available apps to users' iOS, Android, Samsung, Samsung Knox, HTC, Windows Phone 8, and Windows 8 devices.
- Provision devices simply and rapidly by enabling user self-service enrollment and by distributing configuration, policy, and application packages in an automated, role-based manner over-the-air.
- Secure devices, applications, and data by setting authentication and access policies, blacklisting and whitelisting applications, enabling application tunnels, and enforcing security policies at the gateway.
- Support users by remotely locating, locking, and wiping devices in the event of loss or theft, as well as remotely troubleshooting device and service issues.
- Monitor devices, infrastructure, service, and telecom expenses.
- Decommission devices by identifying inactive devices and wiping or selectively wiping devices upon employee departure.
- Run reports on user and device actions.

XenMobile MDM contains the following products:

- XenMobile Device Manager allows you to manage mobile devices, set mobile policies and compliance rules, gain visibility to the mobile network, provide control over mobile apps and data, and shield your network from mobile threats. With a “one-click” dashboard, simple administrative console, and real-time integration with Microsoft Active Directory and other enterprise infrastructure like PKI and Security Information and Event Management (SIEM) systems, Device Manager simplifies the management of mobile devices.

- The Secure Mobile Gateway provides access control for email and calendar services. You can configure Secure Mobile Gateway to allow or block access to connection requests from devices based on device status, app blacklists or whitelists, and a host of other compliance conditions. The status of requests blocked by Secure Mobile Gateway can be immediately viewed on the Device Manager dashboard so that you can take appropriate action.
XenMobile Multi-Tenant Console is a web console that enables service providers and organizations to administer several physical servers running Device Manager from a single site.

XenMobile Remote Support application provides several tools to assist in the inspection, troubleshooting, and modification of remotely controlled handheld devices.

XenMobile ZSM Lite is a component that enables access to query Blackberry and ActiveSync environments and provides the device and user information to Device Manager through the XenMobile Mobile Service Provider.
Device Manager

You can use XenMobile Device Manager to manage mobile devices, set mobile policies and compliance rules, gain visibility to the mobile network, provide control over mobile apps and data, and shield your internal network from mobile threats. With a “one-click” Dashboard, simple administrative web console, and real-time integration with Microsoft Active Directory and other enterprise infrastructure like Public Key Infrastructure (PKI) and Security Information and Event Management (SIEM) systems, Device Manager simplifies the management of mobile devices.
Device Manager 8.5

You can use XenMobile Device Manager to manage iOS, Android, Windows 8 and Windows Phone 8, Windows Mobile, and Symbian mobile devices. The topics in this section provide the following information:

- **About This Release:** Information about new features, supported features by platform, and known issues.

- **System Requirements:** Information about system requirements for installation and deployment of the Device Manager server software.

- **Licensing:** Information about how to obtain licenses for XenMobile Device Manager software.

- **Installing Device Manager Server:** Information and instructions on how to install and deploy the Device Manager server and database, as well as how to back up the system, upgrade, and install patches. Also includes information about how to deploy into a High Availability environment.

- **Requesting an APNS Certificate:** Instructions for how to request and configure an Apple APNs certificate for managing iOS devices.

- **System Administration:** Provides information about configuring Device Manager in order to import and manage users, create user groups and roles, enroll users into the system, define and upload applications, create and deploy policies and device configurations, use reports, and more.
Device Manager 8.5

You can use XenMobile Device Manager to manage iOS, Android, Windows 8 and Windows Phone 8, Windows Mobile, and Symbian mobile devices. The topics in this section provide the following information:

- **About This Release**: Information about new features, supported features by platform, and known issues.

- **System Requirements**: Information about system requirements for installation and deployment of the Device Manager server software.

- **Licensing**: Information about how to obtain licenses for XenMobile Device Manager software.

- **Installing Device Manager Server**: Information and instructions on how to install and deploy the Device Manager server and database, as well as how to back up the system, upgrade, and install patches. Also includes information about how to deploy into a High Availability environment.

- **Requesting an APNS Certificate**: Instructions for how to request and configure an Apple APNs certificate for managing iOS devices.

- **System Administration**: Provides information about configuring Device Manager in order to import and manage users, create user groups and roles, enroll users into the system, define and upload applications, create and deploy policies and device configurations, use reports, and more.
About This Release

The following features are available in this release:

- Configurable dashboard with device filtering and one-click bulk actions.
- Multi-factor, secure individual or mass device enrollment using notifications for iOS and Android devices.
- Policy creation and deployment for iOS, Android, Windows Mobile, and Symbian with management support for Windows 8 and Windows Phone 8, plus app deliver capabilities for Windows Phone 8.
- Ability to distribute in-house and externally available apps to iOS, Android, Samsung, Samsung Knox, HTC, Windows Phone 8 Devices.
- Browser policies to manage Citrix WorxWeb Lite browser app, such as URL filtering (and blocking), security settings, and bookmarking.
- Support for Samsung Safe and Knox policies on Android.
- Compliance and security enforcement with automated actions based on customizable compliance criteria.
- Secure document and file sharing with SharePoint resource configurations and secure, on-device document container.
- Encrypted email attachments to ensure only users on managed devices can safely read corporate documents.
- Direct integration with LDAP and Microsoft Active Directory.
- Extensive role-based access controls spanning the entire product feature set that you can apply to users and groups.
- Geo-fencing and geo-tracking for advanced perimeter-based device security and compliance and asset tracking.
- Event-based and ad-hoc notifications and automated actions.
- Remote wipe (full and selective) of device assets.
- App Lock to prevent non-compliant or malicious apps from launching on the device.
- App removal policies for remotely removing unwanted apps.
What's New

This release of the XenMobile Device Manager (MDM edition) contains the following new features:

- Citrix Mobile Enroll for iOS Users. The Citrix Mobile Enroll app enrolls first time iOS users into the XenMobile system, enabling those devices to be securely managed. Once enrolled, iOS users can receive the Worx Home app and access to the Worx Home Store. (Citrix Mobile Enroll is only needed once by first time XenMobile iOS device users.)

- New Worx Home App for iOS and Android. The new Worx Home app provides access to features provided by both XenMobile MDM and XenMobile App on their iOS and Android devices:
  - Secure document and file access through SharePoint DLP (local and shared) and ShareFile
  - Worx Store access to wrapped apps distributed through XenMobile App Controller
  - Location services, such as geo-locate and geo-fence
  - GoToAssist features for remote support, such as support ticket creation, chat, calls, and email
  - Log collection for support
  - Upgrading customers can update their Connect apps to Worx Home through the iTunes or Google Play app stores
  - Works Home Store for App Delivery. The new Worx Home Store for XenMobile MDM and Enterprise iOS and Android users provides access to a variety of apps from Device Manager and App Controller (and XenDesktop through App Controller). Upgrading Citrix Receiver users will retain their HDX apps, and new users can access their HDX apps if Citrix Receiver is installed on the device. The Worx Home Store can also be branded with a custom image or company logo.

- Works Home for Windows Phone 8. Worx Home for Windows Phone 8 provides an Enterprise app store for Windows Phone 8 devices for XenMobile MDM users, and can deploy Windows Phone 8 apps as well as custom built Windows Phone 8 app store apps.

- Citrix WorxMail for Secure Email Management. Citrix WorxMail provides secure email app for XenMobile Enterprise iOS and Android users enabling secure email access and access email on their devices, providing control over email body and attachment, the ability to set “Open in” controls and data leak protection, send email with ShareFile attachments (XenMobile Enterprise edition only), block users from email who have rooted or jailbroken devices, set policies to require password and/or Wi-Fi or Network connection for access, and more.

- Citrix WorxWeb for Secure Browser Management. Citrix WorxWeb for XenMobile Enterprise iOS and Android users enables policy control over native browser for secure web access, such as:
What's New

- Block unapproved web sites in the browser
- Provide custom bookmarks
- Block users who have rooted or jailbroken devices
- Require log in using pin or password, or pattern screenlock
- Require Wi-Fi or internal network controls
- Block screen capture, camera, location services, SMS
- Secure Browser with Citrix XenMobile Web for iOS. You can configure policies for the Citrix XenMobile Web to block unapproved web sites in the browser, provide custom bookmarks, enable selective or automatic cache clearing and browsing history deletion, enable or disable document or file download, prevent launching new URLs, plus the ability to show or hide the URL address bar.

New Device Manager Features

- Streamline Enrollment for Citrix Mobile Enroll on iOS. You can streamline the iOS enrollment process on Citrix Mobile Enroll by configuring a publically trusted SSL cert on the XenMobile Device Manager server, which eliminates requiring the end user needing to install the root CA.
- Role Based Access Control (RBAC) Additions and Enhancements. This release has extended Role Based Access Controls (RBAC) for all of the Device Manager tabs across all features, providing more ease of use in assigning features to user roles.
- App Store Branding. You can now customize your iOS or Android app store to match your company branding, look and feel, giving it a custom logo.
- Terms and Conditions Acceptance Reporting. You can now generate a PDF report of users who have accepted and declined custom Terms and Conditions during the device enrollment process.
- Cisco AnyConnect VPN Policy for Android. Provides the ability to push and auto provision Cisco AnyConnect VPN configuration such as user credentials, identity certificates along with LDAP authentication and user ID certs.
- Support for Entrust PKI Credential Provider. This release provides Entrust PKI Credential Provider support to enable users the ability to request user identity and device identity certificates; ability distribute user identity certs to mobile services such as Exchange ActiveSync, WiFi, VPN, SharePoint; ability to automatically renew user ID and certs without disruption of service; and the ability to revoke certs for targeted user and/or devices.
- Renaming File for Upload. This feature allows you to rename a file or an app when you upload it to XenMobile Device Manager, so when you push the file to device, the name is changed to the custom label you specify.
- Automatic Notification for APNS Expiration. Admins can now configure notifications to be sent to alert users before their APNS expires.
New Device Manager Server Components

- XenMobile Mail Manager (XMM) with Windows PowerShell and Office 365 Support. The XenMobile Mail Manager (XMM) provides a non-Gateway solution that provides control of Microsoft Exchange traffic without requiring TMG or NetScaler. Instead of blocking mail traffic at the network Gateway through a proxy, XenMobile Mail Manager programs Exchange to block or allow access to mail based on your company’s security and compliance policies.

This solution provides a native Mobile Service Provider to XenMobile server that can snapshot and query Blackberry Exchange Servers (BED) for BlackBerry device management so you can manage Blackberry, BB10, Windows 8, Palm, Non-Samsung/HTC Android devices, integrating Citrix GoToAssist functionality. It also provides support for Windows PowerShell (command line APIs) to support Exchange Access Control for those customers who use a hosted (cloud) Exchange service, as an alternative to TMG for basic ActiveSync controls.

- XenMobile 8.5 XenMobile Mail Manager - BlackBerry Support.
  - BlackBerry 10. Ability to recognize Blackberry 10 devices through XenMobile Mail Manager (XMM) ActiveSync traffic filtering and either allow or deny ActiveSync traffic. Also, ability to quarantine Blackberry devices.
  - BlackBerry BES 5. Ability to monitor BlackBerry devices from BES 5 servers through XenMobile Mail Manager (XMM) and provide device management operations such as remote wipe or password reset.
  - XenMobile NetScaler Connector (XNC). The XenMobile NetScaler Connector (XNC) provides a device level authorization service of ActiveSync clients to NetScaler acting as a reverse proxy for the Exchange ActiveSync protocol. Authorization is controlled by a combination of policies defined within the XenMobile Device Manager and by rules defined locally by XNC. XDM provides whitelisting and blacklisting of devices based on compliance with high-level policies such as detection of jailbroken devices or detection of specific apps. The XNC local rules are typically are used to augment the XDM rules in cases where specific overrides are required; for example to block all devices using a specific operating system version.
New Android Features

- Support for Samsung Knox Container and Security Policies. This release provides policy support for the new Samsung KNOX security features, which provide the ability to push policies such as password, app restrictions, advanced remote support, kiosk mode to the Samsung device KNOX container using a deployment package.

- New Samsung SAFE Policies. This release provides support for the following Samsung SAFE policies:
  - Samsung KIOSK Mode
  - Ability to set lock and home screen wallpaper
  - Ability to populate required or whitelisted applications in the dock bar
  - Android App Lock Policy User Interface. XenMobile Device Manager web console provides a user interface for creating Android app lock policies, rather than editing the native XML file. App lock functionality is extended to include the ability to Allow or Lock Google Play, Camera, Settings, YouTube. Advanced settings include setting app Backlist or Whitelist, or restrict apps based on app package id. (Note that pre-existing Android app lock policies configured in XML will have to be manually converted).

- Android Device Lock. Allows you to lock a device and choose whether to allow the user to unlock using their own password, or enforce an admin-supplied password for unlock.

New Windows 8 Features

Support for App Delivery to Windows Phone 8. You can now publish and distribute Windows Phone 8 apps directly your users using an Enterprise Hub policy and an application enrollment token (AET), which enables you to bypass the Windows Phone Store.
Feature Support by Device Platform

XenMobile Device Manager allows you to manage the following mobile device platforms:

- Apple handheld devices (iPhone, iPad) using minimum version iOS 5.0
- Android handheld devices using minimum version 2.3
  - Device Manager supports the HTC API version 0.5.0
- Windows 8 Phone
- Windows 8 Pro or Enterprise devices (Sideload license required for enrollment)
- Microsoft Windows Mobile and its derivatives:
  - Windows Mobile minimum versions 5.x or 6.x (PocketPC or Smartphone Edition)
  - Pocket PC 2003
  - Windows CE minimum versions 4.x, 5.x or 6.x
  - BlackBerry handheld devices using BlackBerry OS minimum versions 5.x, 6.x, and 7.x
- Symbian S60 FP2 and above, Symbian 3

Due to platform restrictions or security features, not all features are supported on all platforms. The following table summarizes the features available by platform.

<table>
<thead>
<tr>
<th>Feature</th>
<th>iOS</th>
<th>Android</th>
<th>Windows Mobile</th>
<th>Windows 8 Phone</th>
<th>Windows 8 (Pro or Enterprise)</th>
<th>BlackBerry</th>
<th>Symbian</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dashboard</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Enhanced Enrollment Modes (OTP, Multfactor, Invitation-based)</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Invitation Client Download</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Email Attachment Encryption</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>App Lock (‘Kiosk Mode’)</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>App Tunnels Mobile SSL VPN</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storage Card Encryption Policy</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature Support by Device Platform</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local device data encryption</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storage Card Disable</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Autodiscovery Logon</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Automated Actions</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Notifications</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Agent Notification</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enterprise App Store</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Locate Device</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GeoTracking, Geo-Fencing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Secure Sharepoint DLP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Remote client installation (OTA)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provisioning of devices &amp; users</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hardware Inventory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Inventory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security - Jailbreak detection</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Full Wipe &amp; Lock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Selective Wipe</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software download &amp; install</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>File transfer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Device Remote Control</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Roaming Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reports (activity &amp; devices inventory)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Known Issues

The following is a list of known issues for the XenMobile 8.5 release:

- Same app in multiple app categories only appears in one category in Worx Home Store. In the Device Manager web console Applications tab, if you define multiple app categories, and then add the same app to multiple app categories, when you push that app to a device, the app will appear in only one app category. (EWB-12540)

- Selective Wipe on Samsung Knox not removing all configurations from Knox container. In some cases, performing a selective wipe of the Samsung Knox container on Samsung S4 devices is not completely wiping entire Exchange ActiveSync configuration from device. (EWB-12626)

- Windows Phone 8 Exchange ActiveSync policy does not allow for password configuration. The Windows Phone 8 Exchange ActiveSync policy does now allow you to set the user password. The device user will need to set that parameter from the device once the policy is pushed. (EWB-13590)

- XenMobile Remote Support Connection Configuration Unicode Display Issue. Unicode are not displayed correctly in connection’s configuration. If you enter Unicode characters in the Remote Support Tool Connection Configuration, they characters do not display correctly when you click OK to save the configuration. (EWB-13240)

- Unable to Search Windows Phone 8 Store imported apps in Device Manager. If you are adding apps from the Windows Phone app store, Device Manager doesn’t support searching for those apps on the store from the Device Manager web console. (EWB-13500)

- Windows 8 does not support strong authentication for enrollment. (EWB-11398)

- Worx Home on Android does not show SharePoint site in portrait view. (EWB-12983)

- Device encryption is not working on Nokia Lumia 820 devices running on Windows Phone 8. (EWB-12372)

- Worx Home on Android app crashes when user tries to upload encrypted files (greater than 3MB) from Local Docs to Shared Docs. (EWB-11406)

- Windows Phone 8 enrollment dependent on port 8443. Windows Phone 8 enrollment use port 8443 for passing in user name password, but if the Device Manager server is not IOS enabled then 8443 is not created, and thus Windows Phone 8 users cannot enroll. (EWB-11988)

- Windows Phone 8 Password Policy Limitation. On Windows Phone 8, there is a limitation when setting a Password policy with Password Complexity to ‘Alphanumeric or Numeric’, ‘Alphanumeric, Numeric or None’ - the result is that it only allows numeric passwords. This is a known issue with Microsoft. (EWB-11916)

- Configuring and deploying multiple Exchange ActiveSync configurations to the same devices resulting in deployment issues. When deploying more than one Exchange
Known Issues

- Windows Phone 8 internal apps (not from the Windows Phone Store) cannot be auto-uninstalled after they are removed from Worx Home Store (Windows ‘Company Store’). The apps must be uninstalled manually from the device. This is a Microsoft limitation. (EWB-13499)

- Windows Phone 8 external apps installed from Worx Home Store (Windows ‘Company Store’) are not being removed after admin issues a revoked or selective wipe of the Windows Phone 8 device. (EWB-12433)

- XenMobile Mail Manager (XMM) Local Rule changes cannot be automatically reversed. Whenever any XMM access rules are changed, added or removed, the entire set of rules (Local, XDM, + Default) is re-evaluated. There is no attempt to automatically “undo” rules that have been removed. (XMM-12626)

- XenMobile Mail Manager supports only one LDAP configuration per-installation. XMM supports only one LDAP configuration per installation. If you want to manage the traffic of more than one LDAP configuration (such as the root domain, sub domain, and so on), you will need to install XMM for each domain. You can set LDAP connection properties to use the Global Catalog Server, which will give you access to global groups across domains. To do this, you modify the connection string from “LDAP:” to “GC:”. For example, instead of “LDAP://dc=citrix, dc=com”, use “GC://dc=citrix, dc=com”. (XMM-81)
Device Manager System Requirements

You can use the following system requirements for installing Device Manager.

**Windows Server**

- Microsoft Windows 2012 Server 64-bit Standard or Enterprise Edition
- Microsoft Windows 2008 Server R2 Standard or Enterprise Edition

**Note:** If you plan to use device certificate templates with Microsoft Certificate Services, the Windows server running the Active Directory Certificate Services must be running Microsoft Windows 2008 Server R2 SP1, Standard or Enterprise Edition.

If you plan to use the SharePoint access management feature, make sure your deployment meets the following minimum Windows 2008 requirements:

- Microsoft Windows 2008 Server R2 Standard or Enterprise Edition with Service Pack 1 or with fix KB976217 installed on the server.
- Microsoft Windows 2008 Server R2 with fix KB976217 installed on the server.

**Java Requirements**

- Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7.

**Note:** If you are upgrading to Device Manager 8.0.1 from a previous version, you need to perform an upgrade to Java 7 from the previous version. See [Upgrading Device Manager](#) for more information.

The Java Cryptography Extension (JCE) is an officially released Standard Extension to the Java Platform. JCE provides a framework and implementation for encryption, key generation and key agreement, and Message Authentication Code (MAC) algorithms. For more information, see [Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7](#) on the Oracle web site.

**Note:** Oracle Java components all must be downloaded separately from the download web site. The JCE components must be installed in the JDK's Java Runtime Environment in order to properly support enrollment of iOS devices. Please follow the installation Read Me instructions that accompany the Java JCE download package [Java SE Downloads](#) on the Oracle web site.

After you download and extract the JCE package, copy the files local_policy.jar and US_export_policy.jar to the `<java-home>\jre\lib\security` folder and overwrite the existing files.
Hardware Requirements

- Physical or virtual Server host environment
- Intel Xeon 3Ghz or AMD Opteron-1.8Ghz server class
- 1 Gb RAM minimum (4 Gb RAM minimum recommended for 64-bit OS)
- 500 Mb free disk space minimum

Example of a recommended configuration for 1,500 mobile users:

- Dual-processor computer with Intel server architecture.
- A minimum of 2 GB RAM available for the application. 1 GB free disk space (without database).

Example of a recommended configuration for 10,000 mobile users:

- Dual quad-core processor computer with Intel server architecture
- 8 GB RAM available for the application. 1 GB free disk space, 5 GB for database

Web Browser Requirements for the Device Manager Web Console

- Internet Explorer Versions 8 and 9
  
  **Note:** The Device Manager 7.0 Dashboard feature is not supported on Internet Explorer 8.0, but is supported on Internet Explorer 9.
- Firefox versions 17.0.8 and 23

Device Manager Repository Database Requirements

The Device Manager Server repository requires a relational Database Management System (R-DBMS). The supported databases are:

- PostgreSQL v8.2 is the default RDBMS that is packaged and installed with Device Manager Server
- Microsoft SQL Server 2005, 2008 or 2008 R2, and 2012

**Important:** The server on which the database is installed requires NTFS partitions.

User Account Needed: For the database server, you will need to create a service account that has admin rights to SQL server, including the following access rights: Creator, Owner, and Read/Write permissions.
Windows Service Account Requirements

The Windows service accounts for the Device Manager Server and the database must be a Local Administrator of the computer on which the Device Manager Server is installed.

SharePoint Data Leak Prevention Requirements

XenMobile supports SharePoint Data Leak Prevention (DLP). To deploy SharePoint DLP in your environment, use the following guidelines:

- SharePoint 2010 or Office 365.
- Windows Server 2008 R2 - SharePoint 2010 with Service Pack 1 is required or install KB976217.
- Windows Server 2008 - Rest API calls will fail unless KB976217 is also installed.

Installation Requirements

When you install XenMobile, use the following guidelines:

**Note:** Domain membership is not required for the Device Manager server.

- Do not install a new version of IIS, and uninstall IIS if it exists on this server.
- Create an external DNS record for the Device Manager server, such mobile.yourcompany.com.
- Obtain an Apple APNS certificate which is needed during the installation of Device Manager. You can obtain an APNS certificate by using the XenMobile APNS Certificate Setup Guide.
- Install Java SE 7 Update 11 (jdk-7u4-windows-x64.exe) on the Device Manager server.
- Install Java Cryptography Extension (JCE) USJP 7 on the Device Manager server.

**To install the Java Cryptography Extension**

1. Install Java SE 7u11.
2. Open the JCE zip file and copy local_policy.jar and US_export_policy.jar to your computer desktop.
3. Navigate to the folder `/Java/jdk1.7.0_x/jre/lib/security` and copy the files from Step 2 to this folder.
There are four prerequisite requirements that must be prepared prior to installation of Device Manager server. Each prerequisite has a subset of requirements that belong to the providing service and the infrastructure groups responsible for implementation and change control. A successful installation requires all prerequisites are met.

### Obtaining an APNS certificate from Apple for iOS Devices

Management of Apple iOS devices by using the native MDM capabilities of the mobile device hardware and operating system requires an APNS certificate to communicate via Apple Push Network Services. In order to obtain a certificate from Apple, follow the steps outlined in the APNS Certificate Request Guide.

### Designating a DMZ IP Address and DNS Host Name

The Device Manager server is designed to be an edge gateway server that resides in the network DMZ. Device Manager requires a static IP address that can be reached from the internet, as well as a registered and published DNS host name so that devices can reach the server during enrollment and communicate with it regularly. It is strongly recommended to use a separate A-record or CNAME record for any host living in a DMZ for anonymity of the true server host name.

### Opening Ports in the Firewall

There are many inbound and outbound ports that must be configured on the network between the Internet and the DMZ, and from the DMZ to your secure network.

The following table is designed to provide a guide for the TCP/IP port requirements for the Device Manager server and mobile device agent connections.

<table>
<thead>
<tr>
<th>Port</th>
<th>Description</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>By default, the Device Manager SMTP configuration of the Notification Service uses port 25. However, if your SMTP server uses a different port, make sure that your firewall does not block that port.</td>
<td>Device Manager Server</td>
<td>SMTP Server</td>
</tr>
<tr>
<td>44 3</td>
<td>Over-the-Air (OTA) Enrollment and Agent Setup (Android and Windows Mobile)</td>
<td>Internet</td>
<td>Device Manager Server</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------------------------------------------------------</td>
<td>----------</td>
<td>----------------------</td>
</tr>
<tr>
<td></td>
<td>Over-the-Air (OTA) Enrollment and Agent Setup (Android and Windows Mobile), Device Manager management console, Device Manager Remote Support Client</td>
<td>Secure network and WiFi</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device Manager server enterprise connection to Apple iTunes App Store (ax.itunes.apple.com). Used for publishing recommended iTunes App Store apps from the available iOS applications within the Device Manager management console and the iOS Agent.</td>
<td>Apple network</td>
<td></td>
</tr>
<tr>
<td>44 3</td>
<td>Device Manager Nexmo SMS Notification Relay outbound connection.</td>
<td>Device Manager Server</td>
<td>Nexmo SMS Relay server</td>
</tr>
<tr>
<td>38 9 or 63 6</td>
<td>LDAP/LDAPS connection from Device Manager server to Directory Service Host (Active Directory Global Catalog server or equivalent LDAP directory service host)</td>
<td>Device Manager Server</td>
<td>LDAP or Active Directory Services</td>
</tr>
<tr>
<td>44 3</td>
<td>SSL OTA Enrollment or Agent Setup (Android and Windows Mobile), All device-related traffic and data connections (iOS, Android, and Windows Mobile).</td>
<td>Internet</td>
<td>Device Manager Server</td>
</tr>
<tr>
<td></td>
<td>SSL OTA Enrollment or Agent Setup (Android and Windows Mobile), All device-related traffic and data connections (iOS, Android and Windows Mobile), Device Manager management console.</td>
<td>Secure network and WiFi</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device WiFi to ‘discovery.mdm.zenprise.com’ on port 443 for autodiscovery enrollment.</td>
<td>Autodiscovery</td>
<td>‘discovery.mdm.zenprise.com’</td>
</tr>
<tr>
<td>14 33</td>
<td>Remote database server connection to separate SQL Server (Optional).</td>
<td>Device Manager Server</td>
<td>SQL Server</td>
</tr>
<tr>
<td>21 95</td>
<td>Apple APNS (Push Notification Service) outbound connection to gateway.push.apple.com, used for iOS device notifications and device policy push.</td>
<td>Device Manager Server</td>
<td>Internet (Apple APNS Service Hosts on public IP network 17.0.0.0/8)</td>
</tr>
<tr>
<td>21 96</td>
<td>Apple APNS (Push Notification Service) outbound connection to feedback.push.apple.com, used for iOS device notifications and device policy push</td>
<td>Device Manager Server</td>
<td>Internet (Apple APNS Service Hosts on public IP network 17.0.0.0/8)</td>
</tr>
<tr>
<td>52 23</td>
<td>Apple APNS (Push Notification Service) outbound connection from iOS devices connected via Wi-Fi network to *.push.apple.com</td>
<td>iOS device on WiFi network service</td>
<td></td>
</tr>
<tr>
<td>84 43</td>
<td>Over-the-Air (OTA) Enrollment for iOS Devices only</td>
<td>Internet, secure network, or WiFi</td>
<td>Device Manager Server</td>
</tr>
</tbody>
</table>
Preparing for Device Manager Installation

<table>
<thead>
<tr>
<th>Port</th>
<th>Description</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>Remote Support Console access to Device Manager to retrieve device list. (Port 443 recommended.)</td>
<td>Remote Support Console</td>
<td>Device Manager Server</td>
</tr>
<tr>
<td></td>
<td>Mobile Application Tunnel access to Application Server (port configured in the tunnel definition)</td>
<td>Device Manager Server</td>
<td>Internal Application Server</td>
</tr>
</tbody>
</table>
Device Manager Remote Support Requirements

**Hardware Requirements**

- Intel Xeon/Pentium 4 -1 GHz minimum workstation class
- 512 Mb RAM minimum
- 100 Mb free disk space minimum

**Operating System Requirements**

- Microsoft Windows 2003 Server Standard Edition or Enterprise Edition with a minimum of Service Pack 1
- Microsoft Windows 2000 Professional with Service Pack 4
- Microsoft Windows XP with a minimum of Service Pack 2
- Microsoft Windows Vista with a minimum of Service Pack 1
- Microsoft Windows 7
Licensing

If you would like to obtain a new or renewed license for your XenMobile products, contact your account representative. Citrix XenMobile requires you to purchase one of the following licensing options:

**XenMobile MDM Edition**

Citrix XenMobile MDM edition is licensed under a per-user or per-device model. User licensing is based on how many unique users have registered devices on the administrative device management server. Device licensing is based on how many unique devices are registered on the administrative device management server. XenMobile MDM edition can be sold under three consumption options:

- Perpetual licensing entitles the user to consume the product in perpetuity (forever).

- Annual licensing entitles the user to consume the product on an annual basis only. Product consumed under the annual model is valid for 12 months from point of purchase after which the license will expire and the administrative management server will no longer be able to control the registered devices. Customers may purchase a new XenMobile annual license prior to the expiration date to assure access to their XenMobile environment will continue to function uninterrupted. Co-termination is not supported for XenMobile annual licenses - meaning Citrix will not support synchronization of the expiration dates for XenMobile annual licenses, either individually or in combination with XenMobile perpetual licenses. Annual licenses are good for 12 months from the date of purchase. This cannot be modified to provide 12 months from the date of the expiration associated with any existing annual licenses.

- Hosted Cloud-based Service entitles the purchased number of users or devices to consume the hosted service for 12 months from point of purchase after which the service will expire and administrators will no longer be entitled to access the administrative device management server.

**XenMobile Enterprise Edition**

XenMobile Enterprise edition is licensed under a per-user model only. XenMobile Enterprise edition can be sold under two consumption options:

- Perpetual licensing entitles the user to consume the product in perpetuity (forever).

- Annual licensing entitles the user to consume the product on an annual basis only. Product consumed under the annual model is valid for 12 months from point of purchase after which the license will expire and the administrative management server will no longer be able to control the registered devices. Additionally, the licenses will cease to support valid connections to the XenMobile environment. Customers may purchase a new XenMobile annual license prior to the expiration date to assure access to their XenMobile environment will continue to function uninterrupted. Co-termination is not supported for XenMobile annual licenses - meaning Citrix will not support
Renewing Device Manager Licenses

You can view the current status of your Device Manager license and update your license on the About tab.

If your current license has already expired and you cannot log in to the Device Manager web console, when you receive your updated license, you can manually add the license file to the server running Device Manager to reinitiate the software.

To update an expired license

1. Log on to the server running Device Manager.

2. Browse to the following location on the server: `C:\program files(x86)\Citrix\XenMobile Device Manager\tomcat\webapps\zdm\WEB-INF` and save the valid file in the WEB-INF folder with a name, such as `license.crt`.

3. Restart the XenMobile service.

   The license file is updated and you can now log on to the server running Device Manager.
Installing Device Manager Server

XenMobile provides an installation setup wizard which automatically installs the necessary components to use Device Manager. During installation, you install the following components:

- Device Manager
- PostgreSQL database
- License files

Before you install make sure that you:

- Disable TCP/IPv6 on the network adapter and in Windows Registry Editor.

  **Caution:** Using Registry Editor incorrectly can cause serious problems that may require you to reinstall your operating system. Citrix cannot guarantee that problems resulting from the incorrect use of Registry Editor can be solved. Use Registry Editor at your own risk.

- Disable the User Account Control Setting in Control Panel.

The installation topics in this section cover the default installation of the Device Manager server on a Windows 2008 Server.
Device Manager Pre-Installation Checklist

Before you install Device Manager, you should collect and record configuration information in order to complete a successful installation. This section includes a checklist that helps you define the information you need about the following:

- TCP/IP LAN connection of 100 Mbps or faster
- Static IP address
- Domain Name Server (DNS) name published on the Internet and your local area network
- Ports
- Virtual private network (VPN) connection settings
- Authentication, including Active Directory settings

This checklist lists the tasks you should complete and the configuration values you should note before you install Device Manager 8.0.

For instructions about installing and configuring Device Manager, see Installing Device Manager Server.

Device Manager Basic Network Connectivity

The Device Manager server is designed to be an edge gateway server that lives in the network DMZ. It will need to have a static IP address that is reachable from the public Internet, as well as a registered and published DNS host name so that devices can reach the server during enrollment and communicate with regularly. It is strongly recommended to use a separate A-record or CNAME record for any host living in a DMZ for anonymity of the true server host name.

<table>
<thead>
<tr>
<th>Public IP address</th>
<th>DNS host name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Note the default gateway IP address.</td>
<td></td>
</tr>
</tbody>
</table>

Device Manager Service Account

The Device Manager service account must be a local administrator of the server. When you configure the service account, it must have the credentials to create databases in Microsoft SQL during Device Manager installation.

<table>
<thead>
<tr>
<th>Service account user name</th>
<th></th>
</tr>
</thead>
</table>
Database Installation

Device Manager creates a database during installation to an existing SQL server. Citrix recommends that the database reside on a separate server than Device Manager.

<table>
<thead>
<tr>
<th>SQL server fully qualified domain name or IP address</th>
<th></th>
</tr>
</thead>
</table>
Deploying XenMobile MDM - Device Manager

XenMobile Device Manager is comprised of the following separate server components:

- **Device Manager** is one server or virtual machine that is typically located in the DMZ. This component provides advanced mobile device management and security for iOS, Android, Symbian, Windows Phone 8, Windows 8, Windows CE, and Windows Mobile devices.

- **Secure Mobile Gateway** is the automated enforcement component that can prevent unmanaged or out-of-compliant devices from accessing your secure network mail environment. You can install the Secure Mobile Gateway on an Exchange Client Access server, Exchange Front End, Microsoft ISA Server, or Microsoft Threat Management Gateway and does not have any prerequisites to complete before installation. For more information, see Secure Mobile Gateway 8.0.1. You can install the Secure Mobile Gateway you install the Device Manager server.

You can install Device Manager to support a variety of existing network topologies and a large number of users. Some of the installation considerations are related to how the users connect to their Information System (Wi-Fi, Cellular, and Ethernet), existing security rules (DMZ, firewalls), user authentication (Directories), and more. Device Manager should be installed on a standalone physical server or dedicated virtual machine.

In order to be compliant with existing IT infrastructures Device Manager can be deployed in various scenarios:

- **Simple installation**: Device Manager is installed behind the firewall.

- **Multi DMZ installation**: Device Manager is installed in the private DMZ behind a proxy located in the public DMZ.

The following figure shows a typical Device Manager deployment.

Figure 1. Device Manager Deployment
Citrix recommends deploying the Device Manager server in the DMZ as a perimeter security server for mobile device management. In a conventional single DMZ architecture, prior to installing the Device Manager Server, make sure that the firewalls authorize the network streams.

You can also install Device Manager server in the secure network, however it will require all network firewall ports described in “Device Manager Server” in Product Requirements from the Internet to be allowed inbound to your secure network where the server is placed to fully operate correctly. See Preparing for Device Manager Installation for information on required ports for a Device Manager deployment.

For Device Manager Server connectivity, make sure that it has:

- A TCP/IP LAN connection of 100 Mbps or more
- A static IP address, and dedicated Domain Server Name (DNS) published both to the Internet and your network
- Availability of all required network firewall ports to allow device traffic inbound from the Internet to the Device Manager server, as well as from the Device Manager server to and from your internal network

The computers on which you install Device Manager Remote Support connects to the Device Manager server IP address on port 80 (by default) to retrieve the list of connected devices and on a port selected (port 82 for example) during installation for remote control of devices through the Device Manager server.
Choosing Device Manager Components to Install

If you are installing Device Manager on your computer for the first time, select Full install, which installs:

- The Device Manager server
- The Device Manager repository database (PostgreSQL) and automatic creation of the database and requisite tables
- The integrated web application server hosting the Device Manager server

Note: If you install an Application Server prior to installing Device Manager, remove Application Server before installing Device Manager.

Installing Databases

Device Manager includes the PostgreSQL database server installation. If you installed a SQL database server on your computer or another server, clear the PostgreSQL check box in the list of components during the installation wizard. The install type switches automatically to Custom. When using a Microsoft SQL server please refer to the installation instructions provided by Microsoft for the SQL server installation. If you do not clear the check box, the PostgreSQL installation wizard appears with configuration instructions.

If you install PostgreSQL, an installation wizard appears. The installation program automatically selects all the default PostgreSQL options required to install an Device Manager server. However, you can check any additional options you want to install. You can also change the installation location with the Browse button.

During installation of PostgreSQL, define the service account that runs the PostgreSQL server. The Service name, Account name, and Account domain fields are already completed. You need to enter a password for the service account.

If the user account does not exist, you receive a prompt to confirm creation of the account. In addition, if the password you chose is not a strong password, then you are prompted to replace the password with a random strong password. Click No in the message dialog box to keep the password you originally entered.

Installing License Files

After you configure the PostgreSQL database, you can then install licenses. If you are using a different SQL database and did not install PostgreSQL, after choosing the initial components and installation location, you install the licenses.
To install Device Manager

Before you install Device Manager, make sure you do the following:

- Disable TCP/IP6 on the NIC and in the registry.
- Disable the User Account Control setting in Control Panel.

The setup wizard includes several discrete tasks. You need to complete the all of the tasks in this topic in consecutive order to complete the entire wizard.

To start the installation wizard


2. On the Installer Language page, select your preferred language, click OK and then on the Welcome page, click Next to start the installation.

3. On the License Agreement page, read the terms and then click I Agree to accept the terms and conditions.

4. On the Choose Components page, do one of the following and then click Next.
   - If you are installing Device Manager for the first time, select Full.

   Full installation includes the following:

   - Installation of the Device Manager Server
   - Installation of the Device Manager repository database (PostgreSQL) and automatic creation of the database and requisite tables
   - Installation of the integrated web application server hosting the Device Manager server

   **Note:** If the Application Server has already been installed, Citrix recommends that you remove it prior to a fresh installation. Installing the Application Server component only and reusing an existing Device Manager database is supported.

   - If the SQL database server is already installed on your computer or on another server, clear the check box. The install type changes to Customized. For example, clear the PostgreSQL component if you are using a local or separately installed instance of Microsoft SQL Server.

   The following steps assume that you are installing Device Manager for the first time and that you chose Full in Step 4.

5. On the Choose Install Location page, leave the default install location or click Browse to select a folder on your computer and then click Install. The PostgreSQL component installs.
To complete the PostgreSQL installation

1. On the PostgreSQL Installation notes page, click Next.

2. On the Installation options page, keep the default options, select additional options, or click Browse to change the installation location and then click Next.

3. On the Service configuration page, define the server account that runs the PostgreSQL server by doing the following and then click Next.
   a. Leave the default settings for service name, account name, and account domain. Also, leave the Install as a service option selected.
   b. In Account password and Verify password, enter a password that meets the password policy of your organization.

   **Important:** If the password you enter does not meet the password requirements, after Step 9, a Password notification appears prompting you to enter another password. If the password does not comply with the organization policy, the installation may fail without warning.

   An Account error message appears stating that the user is not found and prompts you to create the user.

4. Click Yes.

   **Important:** Although the message appears as an error, the message means that the designated account doesn’t yet exist and that you must create the account.

5. If a Password notification warning appears, click No.

6. On the Initialize database cluster page, do the following:
   a. In Locale, click English.
   b. In Superuser name, define an administrator account for the database.
   c. In Password and Password (again), type a password and then click Next.

7. On the Enable procedural languages page, leave the default PL/pqsql check box selected and then click Next.

8. On the Enable contrib modules page, select any modules you want to enable, click Next and then click Next again to start the database installation.

9. When the database installation is complete, click Finish.

To install Device Manager
To configure the connection to the RDBMS

1. On the Device Manager License page, click Browse to specify the .crt license file on your computer that contains the valid license keyword provided by Citrix and then click Next. Next, you configure the Device Manager Repository connection parameters for the selected relationship database management system (RDBMS).

2. On the Configure database connection page, in Database driver, click an RDBMS, such as PostgreSQL.

   - If you click PostgreSQL or a Microsoft SQL database, you need to configure a user name, as well as the following:
     - In Password, enter the password you defined when you installed the PostgreSQL database.
     - In Database name, enter a database name or leave the default value.
   
   - If you click a database other than PostgreSQL, such as SQL Server, you need to configure the SQL Server home name or IP address, as well as the communication TCP port of the database server. The default TCP port is 1433.

   **Note:** Be sure to use the correct local or domain account user name, password, and desired database name that you configured during installation. The account used for Microsoft SQL should also have db_owner, db_creator, db_writer, and processadmin rights.

3. Click Check the connection and then click Next.

4. In the Confirmation message, click Create and then when a message appears stating that the connection to the database is successful, click OK.

To configure and register Crystal Reports

With Crystal Reports, you can process the mobile device connection and session logs to generate activity reports online by using the Device Manager web console, or offline from the Device Manager repository database. The reports include watermark with registration information. To remove the watermark, you need a Crystal Reports Developer Edition license and a keycode for the product. If you did not enter a license serial number during installation, you can define it later by following these steps:

1. Edit the crconfig.xml configuration file located at in the Device Manager setup folder, which is typically \Program Files\Xenmobile on a Windows server:

   tomcat\webapps\Device Manager\WEBINF\classes\crconfig.xml

2. Add your serial number by editing the <keycode></keycode> element. For example, if your serial number is XXXX-YYYY-ZZZZ, modify the line as follows:

   <keycode>XXXX-YYYY-ZZZZ</keycode>

3. On the Crystal Report Java Reporting Components configuration page, to leave a watermark on the reports, leave the keycode blank. Or, to remove the watermark, enter your keycode for the product and then click Next.
To configure the server connectors

On this page, you configure the connection between the Device Manager agent and the Device Manager server for the initial download of the Device Manager agent and subsequent updates and for establishing connections between the Device Manager agent and the Device Manager server in a common operation.

1. On the Configure the modes of connection page, configure the following:
   a. Enable iOS. Select this check box if you manage iOS devices.

   **Important:** You can only configure this option during installation. If you do not select this option and you want to enable the mode in the future, you must reinstall the application server.

   b. Both HTTP and HTTPS access (recommended). Click this option to complete the standard configuration, which enables HTTP downloading and common HTTPS connections.

   c. Only https. Click only if the agent setup wizard file (setup.cab or setup.apk) is not downloaded through the web browser of the Windows Mobile or Android device.

   d. Only http. Click only if a VPN is already installed between the Device Manager server and the mobile devices.

   e. Authentication code for applications/tunnels. Enter a prefix that Device Manager uses to create authentication keys used by the software. Use a simple alphanumeric word or passphrase. Use mixed cases, numbers, and letters only. Then, record this value for use later when you configure the system.

2. On the Define an HTTP connector page, leave the default port of 80 or enter a different port number.

3. In Maximum concurrent connections, leave the default value of 20 or enter a different number. In standard operating mode, HTTP connections are used only to upload the Device Manager agent and to connect the Device Manager Remote Support application. Each connection represents how many device (client) connections you want to allow at any one given time, simultaneously.

   If you want to allow inbound connections to the Device Manager server through the HTTP port, you can also give your server an external IP address.

   **Note:** If you change the Device Manager server IP address, the change is transparent to users if the external address in the Device Manager Server SSL certificate has not been changed and you chose the All local Addresses option during configuration. It’s also recommended that you update the security rules.

4. On the Define an HTTPS connector page, leave the default port of 443 or enter a different port number.

5. In Maximum concurrent connections, leave the default value of 400 or enter a different number and then click Next.

6. On the Define an HTTPS connector for iOS enrollment page, you can leave the default port of 8443 or enter a different number. iOS standards typically rely on a connection through port 8443.
7. In Maximum concurrent connections, leave the default value of 20 or enter a different number and then click Next.

To integrate the PKI

The Device Manager server has an integrated Public Key Infrastructure (PKI) which incorporates several Certification Authorities (CAs) to manage the key pairs and certificates required to authenticate the server and mobile devices. The certificates are in X509 v3 format. The Device Manager server is always authenticated, although the device authentication is optional and is only activated and applied if the license includes the Device Manager Secure Device option.

**Note:** If the Device Manager Secure Device option is not included in your Device Manager license, Device Manager does not use the CA for mobile devices.

1. On the Define the root certification authority page, do the following and then click Next.

   a. In Keystore file path, do not change the default path. The server configuration provided the file path.

   b. In Keystore password, enter the private password used for each component of the local CA. Although you can use the same password for each CA keystore component, it is recommended to use separate passwords for the Root CA, Server CA, Device CA, and Web Service CA certificates. Be sure to write down all keystore passwords uses and save in a safe location. Matching green color password fields confirm that you entered the same password in the Password and Confirm password fields. Passwords must have at least eight characters, and can consist of alphanumeric and ASCII symbol values. Passwords are case sensitive.

   c. In Common name, leave the default name to associate with the creation of the CA component and certificate. If you change this field, your devices may not receive the proper chain of certificates and will not be able to enroll.

   d. In Organizational unit, enter a value typically given to the entity or group that has management authority over the CA.

   e. In Organization, enter a value typically given to the entity or company that is parent to owning the CA and its rights.

   **Note:** The Root CA certificate is used to issue and sign certificates for intermediate server and client-device CAs. It is also used to regenerate intermediate certificates in the event of compromise. It may be installed in the operating system as a trusted CA root certificate. To avoid alert messages by Internet Explorer 7 as to the validity of certificates issued by this CA, install the root certificate in the operating system.

2. On the Define the server certification authority page, do the following and then click Next. The intermediate mobile device CA is used to issue and sign mobile device certificates. It is also used to regenerate mobile certificates in the event of compromise.

   a. In Keystore file path, do not change the default path. The value is required by the server configuration.
To install Device Manager

b. In Keystore password, enter the private password used for each component of the local CA. Although you can use the same password for each CA keystore component, it is recommended to use separate passwords for the Root CA, Server CA, Device CA, and Web Service CA certificates. Be sure to write down all keystore passwords uses and save in a safe location. Matching green color password fields confirm that you entered the same password in the Password and Confirm password fields. Passwords must have at least eight characters, and can consist of alphanumeric and ASCII symbol values. Passwords are case sensitive.

c. In Common name, leave the default name to associate with the creation of the CA component and certificate.

d. In Organizational unit, enter a value typically given to the entity or group that has management authority over the CA.

e. In Organization, enter a value typically given to the entity or company that is parent to owning the CA and its rights.

3. On the Define the certificate for HTTPS page, do the following and then click Next. The server shows the HTTPS certificate (SSL server connection) to the mobile devices in order to prove the server identity. The certificate prevents man-in-the-middle attacks. A man in the middle attack is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker. The attacker must be able to intercept all messages going between the two victims and inject new ones, which is straightforward in many circumstances (for example, an attacker within reception range of an unencrypted Wi-Fi wireless access point, can insert himself as a man-in-the-middle)

a. In Keystore file path, do not change the default path. The value is required by the server configuration.

b. In Keystore password, enter the private password used for each component of the local CA. Although you can use the same password for each CA keystore component, it is recommended to use separate passwords for the Root CA, Server CA, Device CA, and Web Service CA certificates. Be sure to write down all keystore passwords uses and save in a safe location. Matching green color password fields confirm that you entered the same password in the Password and Confirm password fields. Passwords must have at least eight characters, and can consist of alphanumeric and ASCII symbol values. Passwords are case sensitive.

c. In IP address or FQDN, enter the IP address or fully qualified domain name (FQDN) of the server.

d. In Organizational unit, enter a value typically given to the entity or group that has management authority over the CA.

e. In Organization, enter a value typically given to the entity or company that is parent to owning the CA and its rights.

**Important:** Retain and safely store all keystore passwords used for the four separate CA installation steps. These are the PKCS#12 certificate files containing the PKI key pairs (*.p12 files) and can be found in the default keystore file path listed during each installation step. Also, keep backup copies of the four *.p12 extension files.
4. On the Define the APNs certificate file for iOS page, in Private key password, enter the associated private key password used to generate the original Certificate Signing Request (CSR).

5. In Certificate file path, specify the file system location of a pre-authenticated Apple Push Notification Service (APNS) certificate file downloaded and converted to PKCS#12 format from the Apple iOS Developer for Enterprise portal.

   Note: APNS certificates are provisioned by Apple, Inc. To obtain an APNS certificate, sign in to the following site with your Apple ID: https://identity.apple.com/pushcert. Inspect the values that appear to the page's text area. If the certificate and password match, the Next button is enabled. Click Next.

6. On the Configure tunnel port(s) used by remote support page, define the port range used by remote support for Android and Windows Mobile devices and then click Next. When you connect to the Device Manager server from the Device Manager web console and then click the Security Report icon in your Internet browser, the certificate displays.

To designate the Device Manager administrator

To connect to the Device Manager web console, you need to configure an account with the administrator role.

1. On the Extended management of the users page, in User name, enter the administrator's name.

2. In Password and Confirm password, enter your password. The password must have at least eight characters.

3. Click Check the user name and then click Next.

To complete the server configuration

1. On the Configure all page, click Finish. When the installation is complete, a page appears stating that the setup was completed successfully.

2. Click Next and then click Finish.

Next, you install the Remote Support and the device provisioning modules.
Upgrading Device Manager to Version 8.5

Upgrading the Device Manager server is a simple, in-place upgrade process. The automated Setup Wizard updates your existing Device Manager installation and database in one step. As a best practice, it is advised to backup the database and Device Manager core application directories and save them to a location as a roll-back plan.

Supported Upgrade Paths:

- 7.1.0 > 8.0.1
- 8.0.1 -> 8.5.0

Note:

If you are running Device Manager version 8.0.1, you should already have the correct version of Java on your server. If you do not, make sure that you are running Oracle Java SE 7 JDK (JDK Download Edition) update 11 and above and Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7. For more information, see Upgrading Device Manager 8.0.1.

Before you upgrade: Before upgrading, make sure that you perform a backup of your Device Manager database and application directory as described here: To perform a full manual backup of Device Manager server

To upgrade Device Manager to version 8.5

1. As Administrator, run the Device Manager executable installation file.

2. Follow the directions in the Setup Wizard.
Backing Up and Restoring Device Manager

Backing up your Device Manager server installation and core application file system directory is crucial to a good disaster recovery or business continuity plan. This section describes backing up and restoring Device Manager.

You can back up Device Manager by using the following methods:

- Stop all services and then make a copy of the entire application directory on the server.
- Copy the application directories required for restoration and also perform a native SQL database server backup by using the PostgreSQL utility called pgAdmin. You can also use Microsoft SQL Server Management Studio for your version of Microsoft SQL Server.

If you want to restore Device Manager, you also use pgAdmin or Microsoft SQL Server Management Studio.
To perform a full manual backup of Device Manager server

A very simple method for backing up a default installation of the Device Manager server is to stop all services and make a copy of the entire application directory on the server.

1. From the Services utility on the Device Manager server, stop the XenMobile Device Manager and the XenMobile Device Manager Database - PostgreSQL 8.3 services. MS SQL database installations should follow the best practices used for the particular type of SQL server installation. Online and Offline backups are acceptable as long as the backup database and transaction logs are maintained together for restoration.

2. Back up the XenMobile Device Manager database and application environment. This is accomplished by making a full directory copy of the Device Manager application directory typically located at: C:\Program Files (x86)\Citrix\XenMobile Device Manager

3. Save the full directory copy to a safe external location such as tape backup or external media storage system. This full directory backup contains the Database, Application, PKI configuration and certificates, and all configuration and log files.
To perform a directory and native SQL backup of Device Manager server

Another method of backup for Device Manager server is to copy the application directories required for restoration and also perform a native SQL database server backup utilizing the default PostgreSQL utility pgAdmin. If utilizing a Microsoft SQL Server database installation the Microsoft SQL Server Management Studio utility is used. The following steps will guide you through the process using the default PostgreSQL pgAdmin III utility only.

1. From the Services utility on the Device Manager server, stop the XenMobile Device Manager service.

2. Start the pgAdmin III utility from Start > All Programs > PostgreSQL 8.3. Database backup is performed using the pgAdmin III utility if using the default PostgreSQL database. For a Microsoft SQL Server database installation use the Microsoft SQL Server Management Studio application and follow the instructions provided by Microsoft or your database administrator to back up your database according to your needs.

3. Enter the password for the default postgres administrator account for the database. This was recorded during installation.

4. Expand the Databases branch of the servers tree in the pgAdmin utility, right-click on the xdm database object, and then select Backup.

5. Enter a directory location and new filename for the backup file then click OK.

6. When completed the backup process will show the following message window. When finished, click Done. The resulting backup file will be saved off to your predetermined location for archival and retrieval when a database restore is necessary.

7. Next, while the Device Manager service is stopped, backup at least the following directories within the main application folder:

   - C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\conf
   - C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\webapps\zdm\WEB-INF

8. Verify the backed-up directory has a complete copy of the Tomcat configuration and PKI certificates. These files are located under the parent directory: C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\conf

9. Verify that the backup directory also contains the license file normally found at:
   C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\webapps\zdm\WEB-INF

10. The Device Manager application and database environment is now fully backed up and can be restored to the same or different system host.
Installing Patches for Device Manager

If a patch has been issued to resolve a problem that applies to your situation and Device Manager implementation, you may download the appropriate patch(es) for your system.

Patches follow the naming convention of ‘a_patch_###.xxxx.jar’ where ### signs are the version release number for Device Manager and xxxx refers to the patch number.

To install the patch, copy the file ‘a_patch_###.xxxx.jar’ to the following directory
%systemroot%\Program Files (x86)\Zenprise\ZenpriseDevice Manager\tomcat\webapps\zdm\WEB-INF\lib or the directory in which you installed Device Manager.

After you copy the file to the directory, restart the Device Manager service.
Configuring High Availability on Device Manager

You can deploy up to three instances of Device Manager to create a high availability pair, which is also called a \textit{cluster}. You configure one Device Manager instance as the primary role in the cluster and the other Device Manager instances as the secondary role in the cluster. In this deployment, the primary Device Manager listens for requests, and serves user requests. The secondary Device Manager synchronizes its data with the data on the primary. The two instances of Device Manager work as an active-passive pair, in which only one instance of Device Manager is active at a time.

If the current primary Device Manager stops responding for any reason, the current secondary Device Manager takes over and becomes the primary. The new primary Device Manager begins to serve user requests.

Device Manager in a cluster configuration requires a network load balancer to create a high availability pair as well as to distribute the load between Device Manager servers.

You need to configure the following:

- Windows Server 2008 R2. Install each Device Manager instance on a separate Windows server.
- Configure the Windows servers as a cluster.
- Virtual IP address or host name on the load balancer. Device Manager uses this information to route user requests.
- SSL session persistence for ports 443 and 8443 on the load balancer.
- SQL Server database accessible from the Device Manager node(s) and user credentials to connect to the database. Each node connects to the same database.
- Network Time Protocol (NTP) server to synchronize time for all nodes and SQL DB server.

After you install Device Manager and configure the initial settings, there are some additional configuration steps. These include:

- Editing an xml file to replicate session information on all cluster nodes in the Tomcat cluster.
- Enabling clustering on Device Manager.
- Configuring properties on the Tomcat server.
- Copy certificates from cluster node 1 to cluster node 2.
- Stopping and starting the Device Manager Windows service.
You can also use the PostGRE SQL database for high availability. If you use this database, you need to run a utility to import database information to Device Manager.
To install Device Manager on cluster node 1:

1. Clear the Database server check box if there is already a MS SQL server in your network.

2. On the Configure database connection screen, create a Device Manager database on your MS SQL server.

3. Install Device Manager on Cluster Node 1.

4. On the certificate creation screen, use the public virtual IP address or FDQN of the hostname configured in the F5 virtual server configuration.

After Device Manager is successfully installed, open a web browser from the same host; for example, Device Manager cluster node 1. Then, open http://localhost/zdm and verify that the Device Manager web console appears. Then, stop the Device Manager Windows service.
**To install Device Manager on cluster node 2**

1. Install Device Manager on Cluster Node 2 and clear database install. Remember to use the same database name as that of Cluster Node 1.

2. Copy the following files from Cluster Node 1 in `<installation_dir>\tomcat\conf` to the same place on Cluster Node 2.
   - `https.p12`
   - `pki-ca-devices.p12`
   - `pki-ca-root.p12`
   - `pki-ca-servers.p12`
   - `pki-ca-root.crt.pem`

3. Import the certificates; do not create new certificates. The installer prompts you to enter passwords with which certificates were created (during installation of cluster node 1). Only the Keystore password text box appears.

4. Enter the same keystore password which was used in 'cluster node 1' for the following screens.

After Device Manager is successfully installed, open a web browser from the same host - cluster node 2, go to http://localhost/zdm and then verify that the Device Manager web console appears. Stop the Device Manger Windows service.
To configure a Device Manager Tomcat Cluster

Tomcat clustering is used to replicate session information on all cluster nodes. In an event of a Tomcat server being unavailable on a cluster node, device connections can fail over to Tomcat servers on other cluster nodes because the state is being preserved across all nodes in the cluster.

**Note:** Make sure to update the configuration files/command to all cluster nodes.

1. Open file `<installation_dir>/tomcat/conf/server.xml` in wordpad and add a `<cluster>` section after the following element: `<Engine name="Catalina" defaultHost="localhost">:

```
<Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster">
<Manager className="org.apache.catalina.ha.session.DeltaManager"
   expireSessionsOnShutdown="false"
   notifyListenersOnReplication="true"/>
<Channel className="org.apache.catalina.tribes.group.GroupChannel">
   <Membership className="org.apache.catalina.tribes.membership.McastService"
      address="228.0.0.8"
      port="45560"
      frequency="500"
      dropTime="3000"/>
<Receiver className="org.apache.catalina.tribes.transport.nio.NioReceiver"
   address="auto"
   port="4000"
   autoBind="100"
   selectorTimeout="5000"
   minThreads="3"
   maxThreads="6"/>
<Sender className="org.apache.catalina.tribes.transport.ReplicationTransmitter">
<Transport className="org.apache.catalina.tribes.transport.nio.PooledParallelSender"/>
</Sender>
<Interceptor className="org.apache.catalina.tribes.group.interceptors.TcpFailureDetector"/>
<Interceptor className="org.apache.catalina.tribes.group.interceptors.MessageDispatch15Interceptor"/>
</Channel>
```

```
<!--
<Deployer className="org.apache.catalina.ha.deploy.FarmWarDeployer"
   tempDir="${catalina.base}/war-temp"
   deployDir="${catalina.base}/war-deploy"
   watchDir="${catalina.base}/war-listen"
   watchEnabled="true"/>
-->
```
To configure a Device Manager Tomcat Cluster

</Cluster>

2. After copying the above contents, check for the following elements in server.xml:

- Membership. Determines cluster membership. address: 228.0.0.8 (multicast address)
- port. 45560 (multicast address and the port determine cluster membership)
- frequency. 500 (broadcast ping send frequency. Must be smaller than timeToExpiration)
- dropTime.3000
- Receiver. Responsible for listening to session tomcat session replication messages
  - address. auto (listening address)
  - port. 4000 (port number used to listen for session replication messages)
  - autoBind. 100 (number of ports to try : 13000 to 13099)
  - selectorTimeout. 5000 (select operation selector timeout)
  - minThreads. 3 (work thread pool configuration)
  - maxThreads. 6 (work thread pool configuration)
To configure the Device Manager Server

1. Edit the ew-config.properties file
   (<installation_dir>/tomcat/webapps/zdm/WEB-INF/classes).

2. Change the following line from false to true:

   ####################################################################
   #
   # CLUSTERING
   #
   ####################################################################
   cluster.everywan.enabled=false

   To

   cluster.everywan.enabled=true

3. Add the following line: cluster.hibernate.cache-provider=com.opensymphony.oscachefl.hibernate.OSCacheProvider Your cluster configuration should look like the following example:

   ####################################################################
   #
   # CLUSTERING
   #
   ####################################################################
   cluster.everywan.enabled=true
   cluster.hibernate.cache-provider=com.opensymphony.oscachefl.hibernate.OSCacheProvider

4. For the DAO configuration, verify that the following properties exist. If not, add them.

   · For MS SQL. dao.configLocation=classpath:com/sparus/nps/dao/hibernate-native.cfg.xml
   · For MySQL database. dao.configLocation=classpath:com/sparus/nps/dao/hibernate-mysql-hilo.cfg.xml
   · For other databases:
     dao.configLocation=classpath:com/sparus/nps/dao/hibernate-hilo.cfg.xml

5. Please add the following properties in ew-config.properties:

   # Everywan cluster shared secret for application connection
   everywan.secret=everywan

   # Everywan node name (used on load balancer front end)
   cluster.everywan.nodeName=auto
To configure the Device Manager Server

    # Everywan direct IP access (ex. used by remote support)
    cluster.everywan.directAccess=auto

    # Everywan broadcast
    cluster.everywan.broadcast.address=228.0.0.8
    cluster.everywan.broadcast.port=45561

    Note: It is recommended that you change the
          cluster.everywan.nodeName=auto to node1 and node2 rather than leave as
          auto, as follows:

    The following parameters are used:

    · cluster.everywan.nodeName. “node1” (or node2, node3. and so on).
    · cluster.everywan.directAccess. “auto” (search for the first IP address of the first
      network interface). If you want to assign a specific IP address, use :
      “ip:192.168.1.251”.
    · cluster.everywan.broadcast.address. “228.0.0.8” (UDP broadcast address).
    · cluster.everywan.broadcast.port. “45561” (UDP broadcast port).

    Important: This broadcast address, “228.0.0.8:45561” must be different from
                the one used by Tomcat server in server.xml.

    For cluster.everywan.directAccess, you can use the following parameters:

    Important: In order for Remote Support to work if the node has two or more nics, you
               might need to put the node IP here.

    · eth1. Use the first IP address of eth1 interface.
    · ip:192.168.1.128. Use the specified IP address.
    · lo. Use the first IP address of the lo interface (127.0.0.1).
To configure Tomcat oscache.properties

File oscache.properties is located under 
<installation_dir>/tomcat/webapps/zdm/WEB-INF/classes.

1. Use wordpad to open the file. At the end of the file, look for JGroups configuration. It looks like the following example:

```
cache.cluster.properties=UDP(mcast_addr=228.0.0.8;mcast_port=45566;diagnostics_addr=228.0.0.8;diagnostics_port=45567;mcast ... ;down_thread=false;up_thread=false):pbcast.GMS(join_timeout=5000;join_retry_timeout=2000;shun=false;print_local_addr=true)
```

```
cache.cluster.multicast.ip=228.0.0.8
```

2. Check the following parameters:

- `mcast_addr= 228.0.0.8`
- `mcast_port=45566`
- `diagnostics_addr= 228.0.0.8`
- `diagnostics_port=45567`
- `cache.cluster.multicast.ip= 228.0.0.8`

- `mcast_addr` and `mcast_port`, `diagnostics_addr` and `diagnostics_port` are used to check the Hibernate cache consistency among the cluster nodes. They must have the same values on all the cluster nodes.

- `cache.cluster.multicast.ip` must have the same address as `mcast_addr`. 
To configure the Tomcat applicationContext.xml file

1. Open applicationContext.xml file under
   <installation_dir>\tomcat\webapps\zdm\WEB-INF\ and verify the following values:

   <import resource="classpath:push_services.xml" />
   <import resource="classpath:ios_configuration.xml" />

   <import resource="classpath:cluster_configuration.xml" />
   <import resource="classpath:deploy-scheduler.xml" />
To run update-hilo.sql on all databases besides MS-SQL

Only for PostGres database, run the PostGres administrator utility pgadmin3.exe located under <installation directory>\postgres\bin.

1. Open File > Add Server and then connect to the postgres database name/instance.

2. Open the query tool and then import update-hilo.sql located under <installationdir>\tomcat\webapps\zdm\sql-scripts\sql_update\PostgreSQL and then execute the same.
To overwrite the .pem file

1. Back up the following files:
   - cacerts.pem
   - cacerts.pem.jks
   - certchain.pem
   - https.crt.pem
   - https.p12.pem
2. Copy and overwrite the files from 'Cluster node 1' <installation_dir>\tomcat\conf to 'Cluster Node 2' <installation_dir>\tomcat\conf.
To start the Device Manager windows service

1. Start Device Manager windows service on both nodes.

2. Verify that individual instances are working. (For example, open browser with URL http://127.0.0.1/zdm).

3. Create a test user on any Device Manager instance.
To test the cluster setup

1. The Virtual Server IP address (in this case, IP 172.30.1.221) should be reachable.

2. Verify that ports 80, 443, and 8443 are open on the virtual server IP address. You can telnet to the virtual server IP address and port 80, 443 and 8443 or a port scanner utility.

3. Open a browser and go to URL http://172.30.1.221/zdm. This should redirect to one of the cluster nodes and eventually open the Device Manager web console.

4. Open a browser and go to URL https://172.30.1.221/zdm. This should redirect to the one of the cluster nodes and eventually open the Device Manager web console.
Requesting an APNS Certificate

In order to enroll and manage iOS devices with Device Manager, you need to set up and create an Apple Push Notification Service (APNS) certificate from Apple. This section outlines the following basic steps for requesting the APNS certificate:

- Use a Windows 2008 R2 Server and Microsoft Internet Information Server (IIS) or a Mac computer to generate a certificate signing request (CSR).
- Request an APNS certificate from Apple.
- Import the certificate to Device Manager.

Note:

- The APNS certificate from Apple enables mobile device management via the Apple Push Network. If you accidentally or intentionally revoke the certificate, you will lose the ability to manage your devices.
- If you used the iOS Developer Enterprise Program to created a Mobile Device Manager push certificate, you may need to take action due to the migration of existing certificates to the Apple Push Certificates Portal. For details, see Apple MDM Push Certificate Migration Information.

The topics that outline the step-by-step procedures are listed in order in this section as follows:

<table>
<thead>
<tr>
<th>To create a CSR by using Microsoft IIS</th>
<th>Generate a CSR with a Windows 2008 R2 Server and Microsoft IIS or on a Mac computer. Citrix recommends this method.</th>
</tr>
</thead>
<tbody>
<tr>
<td>To create a CSR on a Macintosh computer</td>
<td>Submit the CSR to Citrix for signing. Citrix signs the CSR with its MDM signing certificate and returns the signed file in a .plist format.</td>
</tr>
<tr>
<td>To submit the signed CSR to Apple to obtain the APNS certificate</td>
<td>Submit the signed CSR to Apple at Apple Push Certificate Portal (Apple ID required) and then download the APNS certificate from Apple.</td>
</tr>
<tr>
<td>To create a .pfx APNS certificate by using Microsoft IIS</td>
<td>Export the APNs certificate as a PCKS #12 (.pfx) certificate (on IIS, Mac, or SSL).</td>
</tr>
<tr>
<td>To create a .pfx APNS certificate on a Macintosh computer</td>
<td></td>
</tr>
<tr>
<td>To create a .pfx APNS certificate by using OpenSSL</td>
<td></td>
</tr>
<tr>
<td>To import an APNS certificate into Device Manager</td>
<td>Import the certificate into Device Manager.</td>
</tr>
</tbody>
</table>
Apple MDM Push Certificate Migration Information

MDM push certificates created in the iOS Developer Enterprise Program have been migrated to the Apple Push Certificates Portal. This migration affects the creation of new MDM push certificates and the renewal, revocation, and downloading of existing MDM push certificates. The migration does not affect other (non-MDM) APNS certificates.

If your MDM push certificate was created in the iOS Developer Enterprise Program, the following situations apply:

- The certificate has been migrated for you automatically.
- You can renew the certificate in the Apple Push Certificates Portal without affecting your users.
- You need to use the iOS Developer Enterprise Program to revoke or download a preexisting certificate.

If none of your MDM push certificates is near expiration, you don’t need to do anything. If you do have an MDM push certificate that is approaching expiration, contact your MDM solution provider. Then, have your iOS Developer Program Agent log on to the Apple Push Certificates Portal with their Apple ID.

All new MDM push certificates must be created in the Apple Push Certificates Portal. The iOS Developer Enterprise Program will no longer allow the creation of an App ID with a Bundle Identifier (APNS topic) that contains com.apple.mgmt.

**Note:** You must keep track of the Apple ID used to create the certificate. In addition, the Apple ID should be a corporate ID and not a personal ID.
To create a CSR by using Microsoft IIS

The first step for generating an APNS certificate request for iOS devices is to create a Certificate Signing Request (CSR). On a Windows 2008 R2 Server, you can generate a CSR by using Microsoft IIS.

1. Open Microsoft IIS.
2. Double-click the Server Certificates icon for IIS.
3. In the Server Certificates window, click Create Certificate Request.
4. Type the appropriate Distinguished Name (DN) information and then click Next.
5. Select Microsoft RSA SChannel Cryptographic Provider for the Cryptographic Service Provider and 2048 for bit length and then click Next.
6. Enter a file name and specify a location to save the CSR and then click Finish.
To create a CSR on a Macintosh computer

1. On a Macintosh computer running Mac OS X, under Applications > Utilities, start the Keychain Access application.

2. Open the Keychain Access menu and then click Preferences.

3. Click the Certificates tab, change the options for OCSP and CRL to Off and then close the Preferences window.


5. The Certificate Assistant prompts you to enter the following information:
   a. Email Address. Email address of the individual or role account who is responsible for managing the certificate.
   b. Common Name. Common name of the individual or a role account who is responsible for managing the certificate.
   c. CA Email Address. Email address of the Certificate Authority.

6. Select the Saved to disk and Let me specify key pair information options.

7. Click Continue.

8. Enter a name for the CSR file, save the file on your computer and then click Save.

9. Specify the key pair information by selecting the Key Size of 2048 bits and the RSA algorithm and then click Continue. The CSR file is ready for you to upload as part of the APNS certificate process.

10. Click Done when the Certificate Assistant completes the CSR process.
To create a CSR by using OpenSSL

If you cannot use a Windows 2008 R2 Server and Microsoft Internet Information Server (IIS) or a Mac computer to generate a Certificate Signing Request (CSR) to submit to Apple for the Apple Push Notification service (APNS) certificate, you can use OpenSSL.

**Note:** In order to use OpenSSL to create a CSR, you need to first download and install OpenSSL from the OpenSSL website.

1. On the computer where you installed OpenSSL, execute the following command from a command prompt or shell.

   ```bash
   openssl req -new -keyout Customer.key.pem -out CompanyAPNScertificate.csr -newkey rsa:2048
   ```

2. The following message for certificate naming information appears. Enter the information as requested.

   You are about to be asked to enter information that will be incorporated into your certificate request.
   What you are about to enter is what is called a Distinguished Name or a DN.
   There are quite a few fields but you can leave some blank
   For some fields there will be a default value,
   If you enter '.', the field will be left blank.
   ----- 
   Country Name (2 letter code) [AU]:US
   State or Province Name (full name) [Some-State]:CA
   Locality Name (eg, city) []:RWC
   Organization Name (eg, company) [Internet Widgits Pty Ltd]:Customer
   Organizational Unit Name (eg, section) []:Marketing
   Common Name (eg, YOUR name) []:John Doe
   Email Address []:john.doe@customer.com

3. At the next message, enter a password for the CSR private key.

   Please enter the following 'extra' attributes to be sent with your certificate request
   A challenge password []:
   An optional company name []:

4. Send the resulting CSR to Citrix.

   Citrix prepares the signed CSR and returns the file to you through email.
To submit the CSR to Citrix for signing

Before you can submit the certificate to Apple, you need to submit the newly created Certificate Signing Request (CSR) to your Citrix sales representative or to a XenMobile technical support representative through email. Citrix needs to sign the CSR so the certificate can be used with Device Manager. Citrix will return the signed file to you as a .plist file.

1. Email the CSR file to Citrix.

2. Citrix prepares a signed CSR and returns it to you in email as a .plist file.
To submit the signed CSR to Apple to obtain the APNS certificate

After receiving your Certificate Signing Request (CSR) from Citrix, you need to submit it to Apple to obtain the APNS certificate.

**Note:** Some users have reported problems logging into the Apple Push Portal. As an alternative, you can log on to the Apple Developer Portal (http://developer.apple.com/devcenter/ios/index.action) before going to the identity.apple.com link in step 1.

1. In a browser, go to https://identity.apple.com/pushcert.

2. Click Create a Certificate.

3. If this is the first time you are creating a certificate with Apple, select the I have read and agree to these terms and conditions check box and then click Accept.

4. Click Choose File to upload your CSR, browse to the CSR on your computer and then click Upload. A confirmation message should appear stating that the upload is successful.

5. Click Download to retrieve the .pem certificate.

**Note:** If you are using Internet Explorer and the file extension is missing, click Cancel two times and then download from the next window.
To create a .pfx APNS certificate by using Microsoft IIS

To use the APNS certificate from Apple with Device Manager, you need to complete the certificate request in Microsoft IIS, export the certificate as a PCKS #12 (.pfx) file and then import the APNS certificate into Device Manager.

**Important:** You need to use the same IIS server for this task as the server you used to generate the CSR.

1. Open Microsoft IIS.
2. Click the Server Certificates icon.
3. In the Server Certificates window, click Complete Certificate Request.
4. Browse to the Certificate.pem file from Apple. Type a friendly name or the certificate name and then click OK.
5. Select the certificate that you identified in Step 4 and then click Export.
6. Specify a location and file name for the .pfx certificate and a password and then click OK.
   
   **Note:** You will need the password for the certificate during the installation of Device Manager.
7. Copy the .pfx certificate to the server on which Device Manager will be installed.
8. Log on to the Device Manager web console as an administrator or as a user with access to the About tab.
9. Click the About tab and then click Update APNS Certificate.
10. In the Update APNS Certificate dialog box, browse to the APNS certificate .pfx file on your computer and then enter a new password.
11. Click Load APNS Certificate.
12. Click Update.
To create a .pfx APNS certificate on a Macintosh computer

1. On the same Macintosh computer running Mac OS X that you used to generate the CSR, locate the Production identity (.pem) certificate that you received from Apple.

2. Double-click the certificate file to import the file into the keychain.

3. If you are prompted to add the certificate to a specific keychain, keep the default login keychain selected and then click OK. The newly added certificate will appear in your list of certificates.

4. Click the certificate and then on the File menu, click Export to begin exporting the certificate into a PCKS #12 (.pfx) certificate.

5. Give the certificate file a unique name for use with the Device Manager server, choose a folder location for the saved certificate, select the .pfx file format and then click Save.

6. Enter a password for exporting the certificate. Citrix recommends that you use a unique, strong password. Also, be sure to keep the certificate and password safe for later use and reference.

7. The Keychain Access application will prompt you for the login password or selected keychain. Enter the password and then click OK. The saved certificate is now ready for use with the Device Manager server.

   **Note:** If you don’t plan to keep and preserve the computer and user account that you originally used to generate the CSR and complete the certificate export process, Citrix recommends that you save or export the Personal and Public Keys from the local system. Otherwise, access to the APNS certificates for reuse will be voided and you will have to repeat the entire CSR and APNS process.
To create a .pfx APNS certificate by using OpenSSL

After you use OpenSSL to create a Certificate Signing Request (CSR), you can also use OpenSSL to create a .pfx APNS certificate.

1. At a command prompt or shell, execute the following command.

   openssl pkcs12 -export -in MDM_Zenprise_Certificate.pem -inkey Customer.key.pem -out apns_identity.p12

2. Enter a password for the .pfx certificate file. Remember this password because you need to use the password again when you upload the certificate to Device Manager.

3. Note the location for the .pfx certificate file and then copy the file to the Device Manager server, so you can use the Device Manager web console to upload the file.
To import an APNS certificate into Device Manager

After you have requested and received a new APNS certificate, you import the APNS certificate into Device Manager to either add the certificate for the first time or to replace an existing certificate.

1. Log on to the Device Manager web console as an administrator or as a user with access to the About tab.

2. Click the About tab and then click Update APNS Certificate.

3. In the Update APNS Certificate dialog box, browse to the .p12 file on your computer and then enter a new password.

4. Click Load APNS Certificate.

5. Click Update.
To renew an APNS certificate

To renew an APNS certificate, you need to perform the same steps you would if you were creating a new certificate. Then, you visit the Apple Push Certificates Portal and upload the new certificate. After logging on, you see your existing certificate or you may see a certificate that was imported from your previous Apple Developers account. On the Certificates Portal, the only difference when renewing the certificate is that you click Renew. You must have a developer account with the Certificates Portal in order to access the site.

**Note:** To determine when your APNS certificate expires, in Device Manager, click the About tab and then look in the APNS certificate information section. If the certificate is expired, however, do not revoke the certificate.

1. Generate a CRS by using Microsoft IIS.
2. Submit the new CSR to Citrix for signing.
4. Click Renew.
5. Generate a PKCS #12 (.pfx) APNS certificate by using Microsoft IIS.
6. Update the new APNS certificate to the Device Manager server. Log on to the Device Manager web console, click the About tab and then click Update APNS Certificate.
7. In the Update the APNS Certificate dialog box, locate the APNS file, enter the certificate password and then click Update.
In order to get users' devices under management, you need to enroll the devices into Device Manager. You first install the Device Manager client software on the user device, authenticate the user's identity, and then install Device Manager and user's profile, so you can manage the device remotely and securely. After the devices are enrolled, you can perform device management tasks, such as applying policies, deploying applications, pushing data to the device, locking, wiping, locating lost or stolen devices, and more.

To enroll users, you must first add users to Device Manager if you have not yet established an Active Directory connection. The topics in this section describe the subsequent required steps for enrolling users:

- Configure enrollment modes - Default, SHP.
- Configure notification servers - SMTP and SMS.
- Configure the enrollment notification template.
- Send enrollment notification.

**Note:** Before you can enroll iOS device users, you need to request an APNs certificate. See Requesting an APNS Certificate for more information.
Enrolling Client Devices - By Platform

XenMobile MDM supports the following device platforms: iOS, Android, Windows Phone 8 and Windows Tablet, Windows Mobile, and Symbian. For a list of platforms versions and the features supported for each platform, see Feature Support by Device Platform. When you install Device Manager client apps, you enroll users and their devices into the Device Manager system.

Enrollment consists of the three following basic steps depending on the device type.

- **Downloading client application software on the user device.**
  
  **Note:** For iOS devices, users install the Citrix Enroll app to enroll. After you enroll on iOS and Android, the Worx app will be pushed to users' devices though the base deployment package.

- **Authentication with Device Manager from the device.**

- **Installing security certificates on the device (iOS and Android).** For Windows Phone 8, Device Manager will install Root and intermediate CA certificates for SSL authentication, and a client certificate used for further communication with the server.
To enable two step enrollment for iOS devices

If you would like to eliminate one of the steps required for iOS enrollment and combine the installation of the user and MDM profile, you can configure the Device Manager server to do so. Before you enable this setting, ensure that the MDM server cert is trusted.

For information on how to add a public cert to the Device Manager server, see Configuring an MDM Server Trust Certificate on Device Manager.

1. On the Device Manager server, from the following location C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\webapps\zdm\WEB-INF\classes open the ew-config.properties file in a text editor.

2. Change the existing default value (true) to (false) as shown below:

   ios.mdm.enrollment.installRootCaIfRequired=false

3. Save the file.

4. Restart the XenMobile Device Manager service.

5. Enroll your iOS devices.
To enroll an iOS device in Device Manager

To enroll your iOS device, go to the Apple iTunes App Store on your device and download the Citrix Enroll app.

1. From the iTunes app store, download and install the Citrix Enroll app.

2. On the iOS device Home screen, tap the Citrix Mobile Enroll app to start it.

3. On the Welcom screen, tap Enroll.

4. Enter your corporate email address and then tap the right arrow.

5. Enter the password and then tap Next.

6. When the device, user account, and Device Manager server is authenticated, a Safari browser starts with instructions for configuring the device. Tap 1 Install Company Profile.

7. On the Install Profile screen, tap Install.

8. If a warning page appears, tap Install.

9. Tap Done. The profile finishes installing.

10. On the Configure your device page, tap 2 Complete Enrollment.

11. When enrollment is finished, close the app.

Once you have complete the enrollment process, if you have added apps - such as Citrix Worx Home and Citrix Receives - to the iOS base package, then the user will see installation messages to install those apps. A Device Manager base package will automatically push policies, configurations, and apps to the user as soon as they enroll, and are useful for making sure first time users get the data they need on their device for device management. For more information on base package setup, see How Base Packages Work.
To unenroll an iOS device from Device Manager

1. On your iOS device, open the Citrix Worx Home app.
2. Tap and hold your finger on Configuration.
3. In the menu, tap Reset Application Data.
4. Tap and hold your finger on Configuration again, and in the menu, tap Unenroll.
5. On the iOS Home screen, hold your finger on the Citrix app and then tap the X icon to delete the app.
6. On the Home screen, tap the iOS Settings app.
7. In Settings, tap General.
8. Scroll down and then tap Profiles.
9. Select and remove all Citrix profiles.
To enroll your Android device in Device Manager

1. Download and install the Citrix Worx Home app on your Android phone from Google Play.

2. Tap the app and on pages that asks if you want to install Citrix Worx Home, tap Install.

3. When the installation finished, tap Open.

4. Enter your corporate email address and then tap Next.

5. Next, enter your username, password, and passcode, and then tap Sign On.

6. In the Activate device administrator screen, tap Activate.

7. Enter your Device Manager user password and tap Sign On.

8. Enter your device security token and then tap Sign On.
To unenroll your Android device from Device Manager

1. Open the Control app, tap the Menu key on your phone and then tap Settings.
2. Tap Personal, tap Security and then tap Device Administrators.
3. Clear the Connect check box.
4. In the Device Administrator dialog box, tap to select Deactivate and then tap OK.
5. In Settings, tap Application Manager.
6. Select the Worx Home, tap Uninstall and then tap OK.
To enroll a Windows Phone 8 device

In order to enroll a Windows Phone 8 device in Device Manager, users need their Active Directory or internal network email address, password, and the server web address for the Device Manager server. Then, they follow this procedure on their devices to enroll.

**Note:** If you plan to deploy apps through the Windows Phone company store, before your users enroll, make sure that you have configured an Enterprise Hub policy (with a signed Citrix Worx Home Windows Phone 8 App 'CitrixWorxHome').

1. On the main screen of the Window 8 phone, tap the Settings icon.

2. Tap company apps.

3. On the company apps screen, tap add account.

4. On the next screen, enter an email address and password and then tap sign in.

5. On the next screen, enter the web address of the Device Manager server, such as: https://<xenmobile_devicemanager_server>:<portnumber>/<instancename>/wpe. For example: https://mycompany.mdm.com:8443/zdm/wpe.

   **Note:** The port number has to be adapted to your implementation, but should be the same port that you used for an iOS enrollment.

6. Enter the user name and domain if authentication is validated through a user name and domain.

7. Tap sign in.

8. If a screen appears noting a problem with the certificate, the error is due to the use of a self-signed certificate. If the server is trusted, tap continue. Otherwise, tap Cancel.

9. When the account is added, you have the option of selecting 'Install company app or Hub. If your administrator has configured a Company App store, select this option and then click done. If you deselect this option, in order to receive the Company app store, you will need to re-enroll.


11. To force a connection to the server, tap the refresh icon. If the device does not manually connect to the server, Device Manager connects to the device every fifteen minutes.

12. To unenroll, tap the trash can icon.
To enroll Windows 8 devices in Device Manager

To enroll a Windows 8 device, you need to be running Windows 8 Professional or Enterprise, and you need to enable sideloading for the device. Sideloading enables you to install the Worx Home app, and is a process that allows the Device Manager server to install the Worx Home app on the device.

1. On the Windows 8 Tablet, open Internet Explorer.

2. Enter the web address of the Device Manager Citrix Mobile Connect app download: https://<devicemanager_servername>/zdm/CitrixMobileConnectAgent.exe

3. Click Save.

4. Browse to the Downloads folder in Windows Explorer and then tap Run as administrator.

5. In the Run the Install Shield dialog box, tap Yes.

6. After the app (and service) installs, on Start, tap the Citrix Mobile Connect app to open it.

7. You will be asked to re-renter your Windows credentials. Enter the Windows domain or User Principal Name (UPN) and then tap Lookup. XenMobile searches for the server address name.

8. Next, on the logon page, enter the Device Manager server name and then tap login. The console status page appears.

9. Tap Save. You will see a status page showing that you have been enrolled.
To enroll a Windows Mobile device in Device Manager

1. Browse to the Device Manager web address for your organization. The web address is in the following format:
   https://<zdmServerName>.domain.com/<zdmInstanceName>/setup

   **Note:** You can use the HTTPS prefix only if you have a certificate issued by a trusted authority, such as Thawte or VeriSign.

2. Tap **Open**.

3. After the Device Manager client installs, when prompted to restart the device, tap **ok**.

4. Open the XenMobile app and then enter the user credentials.

5. Verify the connection. In the heading area, a solid green ball indicates a good connection.
To enroll a Symbian device in Device Manager

1. Browse to the Device Manager enrollment web address. The format is: https://<DmServerName>.domain.com/<DmInstanceName>/setup. The setup file downloads on the Symbian device and the installation starts.

2. On the Install screen, tap OK.

3. Tap Phone Memory as the location where XenMobile agent installs.

4. When the installation is complete, tap Yes to open XenMobile.

5. On the Security Details screen, tap OK to allow XenMobile to access the phone.

6. Enter the first four numbers of the server code as 2831 and then tap OK.

7. On the Control Request Accepted screen, tap OK.

8. Enter the user name and password, server name, port, and instance name for the XenMobile server and then tap OK. The connection information appears.

9. Tap Options to review server connection details and then tap Close to finish the setup.
Configuring Enrollment Modes

You can use the enrollment options in Device Manager to configure device enrollment modes. You can use the Enrollment feature to send enrollment notifications to groups of users. The notifications invite users to easily enroll their devices into Device Manager, use a download link for the Device Manager client software, and use a link to enroll by using their internal network credentials.

You can use the enrollment options to choose varying levels of security for Device Manager enrollment, such as a one-time PIN to ensure the identity of the user, or multi-factor authentication (user, password, and PIN).

When you send an enrollment invitation, Device Manager uses the default mode in the Options dialog box unless you first modify the enrollment mode before sending the notification. The following table lists the enrollment modes you can configure in Device Manager.

<table>
<thead>
<tr>
<th>Enrollment mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>High Security</td>
<td>This enrollment mode sends the user the following three emails:</td>
</tr>
<tr>
<td></td>
<td>‣ An email with a download link that allows the user to download and install the Connect client app.</td>
</tr>
<tr>
<td></td>
<td>‣ An email with an enrollment invitation web address, that allows the user to launch the client app and enroll the user's device.</td>
</tr>
<tr>
<td></td>
<td>‣ An email with a one-time PIN that the user must enter when enrolling the device, along with the user’s Active Directory (or local) user name and password.</td>
</tr>
<tr>
<td></td>
<td>When using this method, the user can only enroll by using the web address in the notification. If the user loses the notification invitation, the user cannot enroll with the sent invitation. You can, however, send another invitation.</td>
</tr>
<tr>
<td>Invitation URL</td>
<td>This enrollment mode sends a single notification to the user that contains a web address that, when clicked, opens the Connect client app. The Device Manager server name and an Enroll button appears. The user taps Enroll to enroll the user's device into Device Manager.</td>
</tr>
<tr>
<td>Invitation URL + Password</td>
<td>This enrollment mode sends a single notification to the user that contains a web address that, when clicked, opens the Connect client app. The Device Manager server name appears, along with a field where the user must enter a password.</td>
</tr>
<tr>
<td>Enrollment Mode</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Invitation URL + PIN    | This enrollment mode sends the following emails:  
  - An email with an enrollment invitation web address that allows the user to download and open the client app, install the app and enroll the users' device in Device Manager.  
  - An email with a one-time PIN that the user must enter when enrolling the device, along with the user's Active Directory (or local) password.  

Using this method, the user can only enroll by using the web address in the notification. If the user loses the notification invitation, the user cannot enroll with the sent invitation. You can, however, send another invitation. |
| Two Factor              | This enrollment mode sends a single notification to the user that contains a web address and a one-time PIN. When the user clicks the web address, the Connect client app opens. The Device Manager server name appears, along with two fields where the user must enter a password and the PIN number. |
| Username + Password     | This enrollment mode sends a single notification to the user that contains a web address that, when clicked, opens the Connect client app. The user then enters a user name and password to enroll the user's device into Device Manager. |
| Username + PIN          | This enrollment mode sends the following emails:  
  - An email with an enrollment invitation web address that allows the user to download and open the client app and then enter a user name and password to enroll the device into Device Manager.  
  - An email with a one-time PIN that the user must enter when enrolling the device, along with the user's Active Directory (or local) password.  

If the user loses the notification invitation, the user cannot enroll with the sent invitation. You can, however, send another invitation. |
To send enrollment notifications to iOS or Android devices

This topic shows you how to configure enrollment for iOS and Android devices using enrollment notifications a username + pin number authentication for enrollment, although you can choose from several different types of enrollment modes with varying levels of security.

1. In the Device Manager web console, select the Enrollment tab.

2. Click New -> Enrollment Invitation.

3. In the Create new enrollment invitation dialog box, on the General tab select the following parameters:
   a. Platform. iOS or Android
   b. Device Ownership. Employee or Corporate.
   c. Enrollment Mode. Password + PIN - This setting enables the user to enroll by using the user’s password and the one time PIN that is generated when the invitation is sent.

4. Next, select the User tab and then select the following parameters:
   a. User. Enter either the user’s local name or AD user name according to the user lookup method configured in your AD LDAP server connection in the Options dialog box. For example, if your LDAP server was configured so the user lookup method is SAM-Account-Name then you would enter 'djones' for David Jones. Note that this entry is case sensitive. In the Notification section, select Notify user(s) immediately, and then choose the following templates.
   b. Notification. To ensure that the notification is sent, select the Notify users immediately option and select this option.
   c. Template for Agent Download. iOS or Android Download Link: This sends the user an email and/or SMS (depending on how the template is configured) that enables the user to download the Citrix MDM client application (the Citrix MDM Connect “agent”).
   d. Template for Enrollment URL: Enrollment Invitation. This sends the user an email and/or SMS (depending on how the template is configured) that enables the user to enroll with the Device Manager system.
   e. Template for Enrollment PIN. This sends the user a one time PIN number that is used to authenticate the user during the enrollment process.
   f. Phone Number/Carrier: If you want the user to receive the notification via SMS using the recipient’s mobile carrier, enter their mobile number here and the carrier.
5. Click Create.

6. Check your device and open the email/text message.

7. Click the download URL to install the Citrix MDM Connect application from the iTunes or Google Play store.

8. When you click the download link for iOS, it will launch the Apple iTunes app store where you can download the Connect agent. When you click the download link for Android, it will launch a page that provides a link to the Google Play app store where you can download the Connect app for Android.

9. If you received a the PIN number, make a note of the PIN number before you begin enrollment of your device.

10. After you have downloaded and installed the Connect agent application to your iOS or Android device, click the Enrollment invitation to start the app and enroll your device.
To configure a Notifications SMTP Server

In order to send enrollment invitations to your users, you need to configure the Device Manager SMTP notifications server and SMS Gateway notification server to be used for user enrollment.

1. From inside the Device Manager web console, select the Options link.

2. In the Options dialog, select Notification Server on the left, and then click New -> SMTP Server.

3. In the general tab, configure the SMTP server you want to use for Device Manager notifications.
   a. Name: Name used to represent this SMTP server account.
   b. Description. Optional description of the server.
   c. SMTP Server. SMTP server hostname.
   d. SMTP Port. Port to be used by the SMTP server.
   e. Secure Channel Protocol. Select secure protocol used by your SMTP server, if configured.
   f. Username. SMTP server login username login.
   g. Password. SMTP Server login user password.
   h. Microsoft SPA. Select if your SMTP server is an Exchange server and is configured to use Microsoft Secure Password Authentication (SPA).
   i. From Name. Name shown in the From box when a client receives a notification email from this server. E.g., Corporate IT.
   j. From Email. Email to be used if an email recipient replies to the notification.
   k. Test Configuration. Click to send a test email notification.

4. In the Advanced tab, configure the following settings:
   a. Number of SMTP Retries. Number if times to retry a failed send.
   b. SMTP Timeout. Timeout duration to wait when sending an SMTP request. Increase this number if message sending is continually failing due to timeouts. Caution: Decreasing this number could increase the number of timed out and undelivered messages.

5. Max. Number of SMTP Recipients. Maximum number of recipients per email message.

6. Click Create.
To configure an SMS Notifications Gateway

Device Manager currently supports only Nexmo SMS gateway, which is a commercial (pay to use) 3rd party gateway. If you do not have a Nexmo subscription, or haven't activated a Nexmo account, then Device Manager will attempt to send SMS messages using the recipient's SMS Carrier Gateway.

You must have Nexmo configured as your SMS Notifications Gateway in order to send SMS notifications, both manual and automated.

1. In the Options dialog under Notifications Server, select New -> SMS Gateway.
2. In the SMS Gateway dialog configure the following information:
   a. Name. Name to be used for the SMS Gateway configuration.
   b. Description. Optional description for the gateway.
   c. Gateway. Nexmo is the only SMS commercial (pay to use) 3rd party gateway that Device Manager currently support. If you do not have a Nexmo subscription, or haven't activated a Nexmo account, then Device Manager will attempt to send SMS messages using the recipient's SMS Carrier Gateway (if a phone is Verizon, for example, it will be sent to 1234567890@vtext.com.)
   d. Key. Key provided by Nexmo when activating your Nexmo account.
   e. Secret. Secret (or password) provided by Nexmo.
   f. Virtual Phone Number. When sending to North America phone numbers (with the +1 prefix), this must be a Nexmo virtual phone number; otherwise, it can be a meaningful label or name. Virtual numbers can be purchased on the Nexmo website.
   g. HTTPS. Select if you want to use HTTPS to transmit SMS requests to Nexmo.
   h. Default Recipient Country. Default SMS country prefix to be used for recipients in your organization. Always starts with a +. Click the link to look up codes for specific countries.
   i. Use Carrier Gateway. Select if you want to use the recipient's carrier gateway to send the SMS notifications. If carrier gateway is not available, then Nexmo will be used to send the message (if an account has been purchased and activated). Deselect this option if you only want to use Nexmo and never use the carrier gateway.
   j. Test Configuration. Click to send a test message using the current configuration. Any obvious errors, such as authentication or virtual phone number errors, will be detected immediately. Messages are received in the same timeframe as those sent between mobile phones.
To configure an SMS Notifications Gateway

3. Click Create.
To configure a notification template

Create and update notification templates used in Automated Actions, Enrollment, and standard Notification message sending. Templates are used to send messages over 3 different channels: SMTP, SMS, and Agent Push (currently iOS only). Notification templates that state 'No' in the Deletable column are default, predefined notification templates that are provides for the main features listed above, but these can be edited and customized to suit your needs.

1. In the Device Manager web console, click the Options link.

2. In the Options dialog, select Notifications Templates from the left side.

3. Click New to create a new template, or double-click an existing template to edit it.

4. In the Notification Template Settings tab, enter or modify the following information:
   a. **From.** Enter a sender for the notification here, which can be either a name, an email address, or both.
   b. **To.** This pre-built notification templates utilizes user macros to ensure that notifications are sent to the correct SMTP recipient address. We recommend that you do not modify macros in templates. For more information on Device Manager macros, see Using Device Manager Macros in Policies.
   c. **Subject.** Subject for the notification.
   d. **Message.** Pre-built notification templates utilize macros to ensure that notifications messages contain the proper information and links. You can edit this field, but we recommend that you do not modify the macro.

5. In the Notification SMS tab, enter or modify the following information:
   a. **To.** This pre-built notification templates utilize user macros to ensure that notifications are sent to the correct SMS recipient number. We recommend that you do not modify macros in templates. For more information on Device Manager macros, see Using Device Manager Macros in Policies.
   b. **Message.** Pre-built notification templates utilize macros to ensure that notifications messages contain the proper information and links. You can edit this field, but we recommend that you do not modify the macro.

6. In the Notification Agent tab, enter or modify the following information that will be sent to the user through agent push:
   a. **To.** Pre-built notification template utilize user macros to ensure that notifications are sent to the correct recipient number. We recommend that you do not modify macros in templates. For more information on Device Manager macros, see Using Device Manager Macros in Policies.
   b. **Message.** Pre-built notification templates utilize macros to ensure that notifications messages contain the proper information and links. You can edit this field, but we recommend that you do not modify the macro.
To configure a notification template

c. Sound File. You can choose a sound file to be played when the message is received by the recipient.

7. Click Create (or OK if you are editing the template).
Enrolling User Names with Special Characters

The following special characters *can* be used for user names of the devices you want to enroll using Device Manager:

```
\ ~ | ! @ # $ % ^ & * ( ) _ -
```

The following special characters *cannot* be used in user names for device enrollment:

```
< > / \ = + , "
```
To enable two step enrollment for iOS devices

If you would like to eliminate one of the steps required for iOS enrollment and combine the installation of the user and MDM profile, you can configure the Device Manager server to do so. Before you enable this setting, ensure that the MDM server cert is trusted.

For information on how to add a public cert to the Device Manager server, see Configuring an MDM Server Trust Certificate on Device Manager.

1. On the Device Manager server, from the following location C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\webapps\zdm\WEB-INF\classes open the ew-config.properties file in a text editor.

2. Change the existing default value (true) to (false) as shown below:

   ios.mdm.enrollment.installRootCaIfRequired=false

3. Save the file.

4. Restart the XenMobile Device Manager service.

5. Enroll your iOS devices.
To enable auto discovery for client enrollment

Auto-discovery allows you to simplify the enrollment process for your end users by allowing them to enter their corporate email address and password during enrollment, without needing to know the Device Manager server information on their device when they enroll.

Auto discovery requires sending Citrix Ops some specific Device Manager deployment information (listed below). Once Citrix receives this information, when your users enroll their devices, Citrix can extract your server domain information and map that to a MDM server address so that the device user has to only enter their AD password to enroll. This information is maintained in the Citrix XenMobile database so that it's always accessible and available.

To send this information create a support ticket or email us at zencloudops@zenprise.com.

1. To enable auto discovery for client enrollment with Device Manager, send an email to zencloudops@zenprise.com, providing the following information:

   a. Your domain name: yourcompany.com

   b. Device Manager server hostname: dmhostname

   c. Instance name: If you are running a cloud hosted version of device manager, this was the instance name you chose during registration. For on-premise installations of Device Manager, this value is usually ZDM or XDM.

   d. Admin email: An email address for your Device Manager administrator. (Optional)

   e. iOS enrollment port: Port being used for iOS enrollment, optional if you changed the default port of 8443.

   f. Port: The port through which the Device Manager server accepts connection (optional unless you plans to change the default 443 port)

2. When this information has been recorded by the Citrix Ops team, you will receive an email confirming that your users can start enrolling with auto-discovery.
To create custom terms and conditions for enrollment

Many companies have their own policies and legal review around using technology that connects to a corporate network. You can use your own, company-branded terms and conditions policy file to enforce your users to Accept or Decline the Terms of Use, for both iOS and Android devices. This file must be in the .pdf format to work properly.

1. First, convert your Terms and Conditions to a .pdf file.

2. Name the file “enduserterms.pdf”.

3. From the Device Manager web console, select the Files tab.

4. Click New File.

5. In the Import a file to the XenMobile MDM database dialog box, click Choose File to select your terms and conditions .pdf file, and then click Import.

6. After you upload the .pdf file, you need to add it to the Base Package for either iOS or Android, so it will be presented to users during enrollment and they will be asked to accept the terms and conditions. If a user declines to agree, then the user will not be enrolled into Device Manager. You can also add it to a regular (non-base) deployment package, if you prefer, depending on how you have your user groups set up to receive packages.
To configure Self Help Portal enrollment

You can configure Device Manager enrollment so users can use the Self Help Portal to enroll their devices.

If you have Self Help Portal privileges in Device Manager, you can select the type of enrollment mode with which you would like users to enroll. After you set the enrollment mode, users can log on to the Self Help Portal and generate enrollment links that allow them to download the Connect app and enroll their devices, or they can choose to send themselves an enrollment invitation. When they receive an enrollment invitation, they can download and install the Device Manager client software and enroll their devices.

1. In Device Manager, click Options in the upper-right side of the console window.

2. In the XenMobile Server Options dialog box, on the left side, click Enrollment Setting.

3. Select an enrollment mode, such as High Security or Two Factor and then click Edit.

4. In the Configure Enrollment Mode dialog box, enter the following information:

   **Note:** The options may differ depending on the enrollment mode you choose.

   a. In Expire After, set an expiration deadline to ensure that if a user does not enroll within a certain time period, the user cannot enroll through the Self Help Portal.

   b. In Maximum Attempts, enter the number of attempts to enroll a user can make before the user is locked out of the enrollment process.

   c. In Template For Enrollment URL, select a template to use for the Enrollment URL. For example, Enrollment Invitation generates or sends the user an email or SMS (depending on how you configured the template) that enables the user to enroll with Device Manager.

   d. In Template for Enrollment PIN, select a template used for the enrollment PIN (if you selected the PIN-based enrollment mode). This option generates and sends the user a one-time PIN to use for secure authentication during enrollment.

5. Click Update.

6. In the Configure Enrollment Mode dialog box, click SHP to set the selected mode as the default Self Help Portal enrollment mode. If you see an error message stating “Please go to Edit and make sure this mode is bound to all notification templates first,” make sure you have selected a template in Step 4.
To check sent notifications logs

If you want to check if an enrollment notification was sent, for example, if any of your users indicated they did not receive an enrollment invitation, you can check the Device Manager notification logs report to determine if the notifications were sent.

1. In the Device Manager web console, select the Reporting tab.

2. Scroll down and click the Sent Notifications Log link to generate the report. You see the report that shows all messages that were sent, the recipients of notifications, notification attempts (successes and failures), time the messages were sent, and so on.
System Administration

XenMobile Device Manager system administration includes such tasks as importing users into Device Manager, creating users and groups, applying Role Based Access Controls (RBAC) to groups, creating and configuring device policies, deploying policy packages, reporting, viewing the dashboard, and more.
To log on to the Device Manager web console

1. Start a Device Manager administrative session by entering one of the following Web addresses in a Web browser:
   - http://<device_manager_server_name_or_IP_address>/zdm
   - https://<device_manager_server_name_or_IP_address>:<port>/zdm

2. Enter the logon credentials of the administrative user account created during Device Manager installation.

   **Note:** If you configured LDAP authentication for Device Manager, be sure to use the account credentials of a user who is a member of the administrative group.

After Device Manager validates the account, the main Device Manager window appears.
Configuring Device Manager to Connect to App Controller

If you have purchased XenMobile Enterprise Edition and are using Device Manager with App Controller to provide apps to your Worx Home users, you will need to configure the Device Manager server so it can communicate with the App Controller server. Conversely, you also need to configure the App Controller server so it can communicate with the Device Manager server.

For secure communication between the Device Manager and App Controller server, both servers should as a best practice use public trusted certs. This is because when App Controller initiates communication with Device Manager, it must validate the server cert from Device Manager, and when Device Manager initiates communication with App Controller, it needs to validate the server cert from App Controller. This handshake will fail if the issuer of the vert is not trusted on both systems.

If you select ‘Allow Secure Communication’ in the App Controller web console interface, then Device Manager will communicate with App Controller on a secure port (for example: 443). This secure communication requires public certs on both servers, and requires that the ports are open in both directions.

The communication between Device Manager and App Controller are RESTful API calls (which can be fully inspected by any inspector between App Controller and XDM) if the traffic was over port 80. The typical communication is App Controller telling Device Manager that userX needs AppY because they subscribed to it on Worx Home, or Device Manager calling App Controller to understand if App Controller exists and the user is registered on it, in order to figure out if the connecting device is in MDM-Only mode or MDM+MAM ('enterprise') mode.

For information on how to add a public cert to the Device Manager server, see Configuring an MDM Server Trust Certificate on Device Manager. For information on how to upload a public cert to the App Controller server, see xmob-appc-import-signed-cert-tsk.html

**Note:** If your device users to be able to access and install Citrix XenDesktop HDX apps, then your device users must have Citrix Receiver installed on their devices.

1. First, you will configure the Device Manager server to be able to communicate with the App Controller server. To do this, first log in to the XenMobile Device Manager web console.

2. Click Options.

3. In the Options dialog box, select Modules Configuration > AppC Webservice API, and then enter the name of the App Controller server, a shared key that you will use when you configure App Controller web console for XenMobile, and select Enable App Controller. Note: Do not click Test Connectivity yet, until you have configured the XenMobile server connection in App Controller web console.

4. Exit the Device Manager web console.
5. Next, log in to the In the App Controller management console.

6. Click Settings -> XenMobile MDM, and then configure the XenMobile Device Manager hostname and port. Also, enter the same shared key that you entered in the XenMobile Device Manager web console. Select the Require Device Manager Enrollment if you want to enforce enrollment (recommended), which requires users to enroll into XenMobile MDM. Last, select ‘Allow Secure Communication’ in the App Controller web console interface, then Device Manager will communicate with App Controller on a secure port (for example: 443). This secure communication requires public certs on both servers, and requires that the ports are open in both directions.

7. Log out of the App Controller management console.

8. Next, log back in to the XenMobile Device Manager console.

9. In the Options dialog box, select Modules Configuration > AppC Webservice API, and then click Check connection to establish communication between XenMobile Device Manager and App Controller web console.

10. When the connection has been established, click Close.
Configuring GoToAssist with XenMobile

In order to use the GoToAssist remote support features with the XenMobile Worx Home app, you will need to configure both Device Manager and App Controller with information obtained from the GoToAssist web console. Specifically, you need to configure GoToAssist and obtain the following information, which you will enter into both the Device Manager web console and App Controller management console:

- GTA Email integration key email
- GTA Chat Token

1. First, obtain your GoToAssist promo code that you received when you bought XenMobile. (If you do not have this, then contact your sales or service representative.)

2. Log in to the GoToAssist web console.


4. Create a new ‘service’ that will be used GTA integration with XenMobile.

5. Select Enable support address, which will be used when a Worx Home user wants to create a support ticket with your organization. The support address obtained here will be used when you configure the Device Manager web console and App Controller management console.

6. Next, select Monitoring > Configure -> Customer Portals and copy the GoToAssist Chat Token. This chat token will be used when you configure the Device Manager web console and App Controller management console.

7. Log in to the Device Manager web console.

8. Select Options.

9. In the Options dialog box, select Modules Configurations > GoToAssist.

10. Enter the following information:
    a. Your company's support phone number.
    b. Your company's support email address.
    c. The GoToAssist support chat token you obtained from the GoToAssist web console.
    d. The GoToAssist Integration key email obtained from the GoToAssist web console.

11. Click Close, and then log out of the Device Manager web console.

12. Next, log in to the App Controller management console.

13. Click Settings > GoToAssist.
14. In the GoToAssist section, enter the following information (same as you entered from Device Manager):
   a. Your company's support phone number.
   b. Your company's support email address.
   c. The GoToAssist support chat token you obtained from the GoToAssist web console.
   d. The GoToAssist Integration key email obtained from the GoToAssist web console.

15. Exit the App Controller management console.
Managing Devices with the Dashboard

The Device Manager Dashboard provides an interactive, high-level view of devices. Each section of the Dashboard displays a unique view of the devices you manage. You can use the Dashboard to do the following:

- **Perform actions on devices.** When you enable actions in a Dashboard widget, you can perform actions on multiple devices represented in a chart; for example, sending notifications to a set of devices on a particular platform.

- **View charts.** Charts show representations of devices in your environment, enabling you to view different groupings of the devices. For example, the Devices by Platform chart shows all of your managed devices by platform type. When you click a section of the chart by platform, the chart changes to show different platform versions for each operating system. If you want to view the devices that are running Android version 3.2 or 3, for example, you can click the slice that displays those operating system versions, and then the Devices tab appears showing only those devices running that version. The chart types you can view are as follows:

  - **Devices by platform.** Displays managed and unmanaged device platforms.

  - **Managed devices by platform.** Displays device platforms for all devices that are managed by Device Manager.

  - **Unmanaged devices by platform.** Displays devices that are not currently managed by Device Manager. Unmanaged devices appear in this chart if, for example, you perform a revoke, wipe, or selective wipe on the device; the device has an agent installed on the device the device is not enrolled, or; the device has an agent installed on it but the user profile or corporate certificate has been removed.

  - **Device by Secure Mobile Gateway status.** Displays the number of devices by Secure Mobile Gateway status: Blocked, Allowed or Unknown. You can click the bars to break down the data by platform.

  - **Devices blocked by reason.** Displays the number of devices blocked by Secure Mobile Gateway, grouped by reason for the block, such as devices that have blacklisted apps installed, devices that have been rooted, and so on. You can click the bars to break down the data by platform.

  - **Device ownership.** Displays the number of devices according to ownership, such as, corporate-owned, employee-owned, or if the unknown ownership.

  - **Android Touchdown license status.** Displays the number of devices that possess a TouchDown license.

  - **Failed package deployments.** Displays the total number of failed deployments per package. Only packages with failed deployments appear.

  - **Devices by carrier.** Drilling down on the chart by clicking the bars provides a further breakdown by platform.
To change the Dashboard chart type, click the gear icon on the lower-left of the report widget and then click a view.

- **View Dashboard alerts.** Alerts show you updates about the following device statuses:
  - New enrollments
  - Non-compliant devices
  - Inactive devices
  - Secure Mobile Gateway blocked devices
  - Number of full and selecting device wipes in the last 24 hours
  - Pending wipes
- **Send notifications.** For example, you may want to notify specific device users about an Internet virus that could affect some Android users.

You can customize the Dashboard to show exactly the information that is most relevant to your needs. You can create up to four different Dashboard views. Each dashboard configuration is saved on a per-user basis. To view the Dashboard, click Dashboard on the Device Manager web console.

### To create custom Dashboards

You can create up to four custom dashboards, in addition to the default dashboard provided by Device Manager. You can choose specific layouts and select the types of widgets to display, depending on your needs.


2. To create a new custom Dashboard layout, select a Dashboard.

3. Select a Layout Style.

4. Click a widget from the list of available widgets and then drag the widget the left to add it the Dashboard. You can edit existing Dashboard configurations by using a drag-and-drop operation to move Dashboard widgets over existing ones, thereby replacing the existing widget. You cannot remove widgets from a Dashboard; you can only add widgets.

5. Click Save and Apply.
To send notifications from the Dashboard

1. On a Dashboard widget, click Enable Actions. The Actions menu appears.

2. Click one of the bars in the graph to select the devices represented in the display. For example, if you wanted send notifications regarding a virus to all Android users, click the Android bar.

3. Click Send Notifications.

4. In the confirmation message, click OK.

5. In the notification dialog box, enter the message you want to send to users. To send quick notifications to select groups of users, you can use the Ad Hoc template.

6. Click Send.
Workflow for Managing a Device

To use Device Manager to manage a device, the typical workflow is as follows:

1. Create an LDAP connection to a user directory. Use the LDAP configuration wizard to enable a connection to a database of users, such as Active Directory, in order to import the users into Device Manager.

2. Define users and groups. You define users on the User tab. Organize your groups so that they match the configuration deployments you define. For instance, if you want to define Exchange push email for mobile users, you may want to define at least two groups:
   
   · The VIP group. Users who are allowed to download email attachments.
   
   · The Standard users group. Users who are not allowed to download the attachments.

3. Create policies. Define the policies to be pushed to the devices. You use the Servers, App tunnels, Registry, XML, and Files tabs to define resources.

4. Create deployment packages and deploy the packages. Create the deployment packages that serve as the containers of policies and apps.

5. Enroll users and devices. Enroll users so they can install the Device Manager agent software and MDM and user profiles on their devices, to ensure secure communication between client and server.
Defining Users and Groups

User account objects represent the users of the mobile devices managed by Device Manager. User accounts are associated to devices by Device Manager as part of the authentication process. Maintaining an accurate roster of users improves mobile device and service management. Groups are logical collections of users that serve as targets for management tasks, such as applying settings, implementing policies, and deploying software.

Note: Device Manager manages group of users, not individual user accounts.

User Account Information

Device Manager supports the following sources of user account information:

- LDAP directory. You can configure Device Manager to read an LDAP-compliant directory, such as Active Directory to import groups, user accounts, and related properties.

  Note: Device Manager retains the source of user accounts. As a result, certain operations are not permitted on user accounts that you source from LDAP directories.

- Manual entry. You can use group maintenance forms in Device Manager to quickly create user accounts.

- Importing a provisioning file. You can develop a file outside of Device Manager containing user accounts and properties and then import the file. Device Manager automatically creates objects and sets properties values.

User accounts appear in the user table within the main display area of the Users tab. The table depicts each user account associated with the group that you select in the Group pane. The User toolbar provides available tasks to perform on user accounts. You can manipulate the table appearance.

The groups in which a user account is a member appear in the Groups column. Note that multiple groups appear as a multi-line entry. User accounts also appear in the Devices table. The user associated with a particular device appears in the User column. The user account shown in the User column represents the user that enrolled on that device.

Group Information

The group structure in Device Manager is flexible. Users may belong to multiple groups, groups may be nested inside of other groups, and the number of groups is not limited. You can create permanent or ad-hoc groups to suit any purpose. Device Manager supports the following sources of group information:

- LDAP directory. You can configure Device Manager to read an LDAP-compliant directory, such as Active Directory to import groups, user accounts, and related properties.
Manual entry. You can use group maintenance forms in Device Manager to quickly create groups.

Groups appear in the Group pane, the area to the left on the Users tab. The pane depicts groups in a hierarchical arrangement with the number of members in each group given as a number in parenthesis after each group name. A default group is automatically created during Device Manager installation to serve as the top-level node for the group hierarchy; all other groups appear as children of this node. Groups imported from LDAP-compliant directories also appear in the group hierarchy, with the LDAP directory name as the primary node. The individual groups of the LDAP directory appear as children of the primary node.

Groups may be nested in the hierarchy without limit. Fully-qualified group names use periods as delimiters. For example, a group of name Corporate.Sales.SalesSupport.Admin implies a nesting model based on organizational structure.

**Note:** User accounts may exist at any level. Thus, on a parent node, the count of group members represents the user accounts associated with that discrete node, and not the sum of the accounts associated with the nodes children.

Groups also appear in the User table. The groups a user belongs to appear in the Groups column.
Creating an LDAP Connection to a User Directory

From the Options dialog box in Device Manager, you can perform the following actions for LDAP connections:

- Create a new LDAP connection.
- Edit an existing connection.
- Set the default LDAP connection.
- Activate or deactivate an LDAP connection.

1. To create a new LDAP connection, click New.

2. Select which type of directory (LDAP or LDAPS).

3. If you chose an LDAPS connection, enter the required parameters and then click Import.

4. After the SSL Certificate is successfully imported, click Next.

5. Define the connection parameters.

   Make sure that the Search user Service Account has the following rights granted to it:

   - READALLUSERINFORMATION
   - READALLNETWORKPERSON

   **Note:** In the lockout limit field, the default is set to zero. However, Citrix recommends using a higher value, as well as a value that is slightly lower than the lockout limit set on your LDAP server. For example, if your LDAP server is configured to a limit of five attempts before lockout, Citrix suggests that you enter a 4 or a 3 in this field.

6. Click Check to test the connection with the LDAP or LDAPS directory. If the connection check with the directory is successful, the following message appears: LDAP directory binding successful.

7. Click OK and then click Next to map the directory attributes to the Device Manager Repository database. You can leave that step as it is and Device Manager will automatically bind the default fields.

8. Click Next to define the mapping between the LDAP groups and Device Manager roles. To add a new group, press Add a group. Select a group and define the role you want to give to that LDAP group.

   **Note:** Unlike the process for creating groups within the web console in a standalone manner in which roles are given to users, here roles are given to an LDAP group.
Creating an LDAP Connection to a User Directory

9. Specify which LDAP or LDAPS directory groups are imported in the Device Manager Repository database and then click Next. A window appears summarizing the directory connection configuration.

10. Click Finish to save the parameters in the Device Manager database.
To add, edit, or delete user accounts

You manage user accounts in Device Manager User table toolbar or the context menu.

To add a user account

1. In the group pane, select a group of which the user account will be a member.
2. Click New user from the toolbar or context menu. The Create a new user window appears.
3. Type a unique name for the user and a password.
4. Select an entry from the Role drop-down list. For more information about roles, see User Accounts and Roles.
5. Optionally, on the Properties tab, set user account attributes.

To edit a user account

1. In the group pane, select the group of which the user account is a member.
2. Click the user account to edit and the click Update. The Update a user window appears.
3. Revise the user account data, then click Update to save the changes.

   Note: If you edit the properties of accounts that you source from an LDAP directory, you do not change data in the directory.

To delete a user account

1. In the group pane, select the group of which the user account is a member.
2. Click the user account to delete and click Delete on the toolbar and then click Yes to confirm the deletion.

   Important: You cannot undo this operation.

   Note: If you delete an account that you sourced from an LDAP directory, you only remove the account from the Device Manage database; you do not change the account information in the directory.
To add or delete groups

You manage groups from the Group pane toolbar or context menu. Device Manager does not have a group edit command, because the only accessible property of a group object is its name.

To add a group

1. Select the parent node of the group.
2. Click New group. The Create a new group window appears.
3. Type a name for the group and then click Create. The group name must be unique relative to its peers in the group hierarchy. In addition, groups may not be added to group nodes that you import from LDAP-compliant directories.

To delete a group

Deleting a group has no affect on user accounts. You can only remove user accounts by using the Delete User command.

1. Select the group to delete.
2. Click Delete.
3. Click Yes to confirm the operation and remove the group.

**Important:** You cannot undo this operation.
User Accounts and Roles

You manage user accounts in Device Manager by using the following commands from the User table toolbar or context menu:

- New user. Add a user account to Device Manager.
- Update. Edit a user account.
- Manage. Maintain a user account’s membership in Device Manager groups, subject to certain limitations.
- Delete. Remove a user account from Device Manager.
- Import. Read a provisioning file containing user accounts or properties to automatically create user account objects and update their attributes.

To search for a user account, on the Users tab, you use the Search tool. Type a search string into the Search field and then click the search icon.

Note: Searches are not case-sensitive; search results display matching user accounts in a separate table that does not include a "currently selected group" in the Group pane. (That is, no groups are selected.)

User Roles in Device Manager

Device Manager implements four default user roles to logically separate access to system functions, as shown in the following table. The columns represent the roles and the rows represent the system functions.

Citrix recommends that you assign the Support role to Help desk staff who require the ability to implement remote control sessions on mobile devices.

<table>
<thead>
<tr>
<th>System function</th>
<th>Administrator</th>
<th>Support</th>
<th>Provisioning</th>
<th>User</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log into administration console</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use remote support application</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use device provisioning application</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Use a mobile device</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

You can use role-based access control (RBAC) to create new user roles with permissions to access specific system functions beyond the functions defined by the default roles as shown
User Accounts and Roles

- To prevent some administrators from viewing or wiping the devices of specific users.
- To allow specific users to only run reports.
- To enable super users to have access to everything, including the ability to create and limit other user roles.

You can view details about users and groups, such as the dates you created and modified a user or group on the Reporting tab.
Configuring Custom Roles with RBAC

You can use the Role-Based Access Control feature in Device Manager to do the following:

- Create a new access control role (associate actions with roles)
- Add groups to a role
- Associate users with roles

To access the feature, in Device Manager, click Options in the upper-right corner, and then click Role Based Access Control.

To create a new access control role

You need to create an access control role in order to enable role-based access control in Device Manager.

1. In the Role Based Access Control panel, click New.
2. In the Create an admin role dialog box, enter a name for the role.
3. Select the features you want to enable for the role and then click Create.

To add groups to a role

When you create a new role, you can also associate a user group with the role as part of the role definition.

1. In the Role Based Access Control panel, select a role and then click Edit.
2. In the Role dialog box, in the Permissions list, select the feature access you want to associate with a role.
3. Under Restrict Group Access, select the group you want to have access to the role, and then click Save. The group you select and the users in the group users receive access to the features you choose.

To associate users with a role

After you create a new role, you can associate users with the role.

1. In Device Manager, click the Users tab and then in the User table, double-click a user or click New User.
Configuring Custom Roles with RBAC

2. In the New User dialog box, enter the user name and password.

3. In the Role list, click the role you want to associate with a user and then click Create.
You can use role-based access control (RBAC) to create custom roles in Device Manager, beyond the default roles. Custom roles grant permissions to user accounts to target specific functionality within Device Manager.

For example, you can create roles to allow the following capabilities:

- To give limited access to devices for administrators whom you want to only perform basic device operations and run reports. After the administrator logs on to Device Manager, only the Devices and Reports tabs appear. When a user only has Report rights, then the Device tab will not appear for that user, but the About tab will display. The About tab also will by default display for users who have no other rights at all.

- To allow an administrator to view, add, locate, edit, and lock a device.

You can associate both user and groups with roles. For example, if you import Active Directory groups into Device Manager, you can apply fine-grained access control to the Active Directory groups.

The following table describes the list of features and accessibility you can associate with a role:

<table>
<thead>
<tr>
<th>Role</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super Admin</td>
<td>Access to all functionality within Device Manager (all functionality listed in this table).</td>
</tr>
<tr>
<td>Authorised Access</td>
<td>Access to the Admin console and/or the Self Help Portal, as well as device access for remote support and remote support access:</td>
</tr>
<tr>
<td></td>
<td>- Admin Console Access</td>
</tr>
<tr>
<td></td>
<td>- Self Help Portal Access</td>
</tr>
<tr>
<td></td>
<td>- Device Access (when Remote Support is enabled)</td>
</tr>
<tr>
<td></td>
<td>- Remote Support</td>
</tr>
<tr>
<td>Dashboard</td>
<td>Access to view all of the Device Manager Dashboard and the ability to customize the Dashboard. In order to perform actions in the Dashboard, however, such as send notification, wipe/selective wipe, revoke, locate, and so on, a user must be granted those specific permissions. Also, if a user is restricted from viewing specific groups, the devices that belong to users in those blocked groups will not appear in the Dashboard.</td>
</tr>
</tbody>
</table>
| Devices | Access to the Devices tab and the ability to perform general device management tasks, such as connecting to iOS devices, importing devices, editing device properties, locating, locking/unlocking, revoking, wiping, and selectively wiping a device. Specific permissions include:  
  · Full wipe device  
  · Selective wipe device  
  · View locations - when selected, users can see location and locate/track device. Includes:  
    · Locate device  
    · Track device  
    · Lock device  
    · Unlock device  
  · Deploy to a Device - allows you to push a deployment package to a device.  
  · Edit device properties  
  · Notification to a device - gives you the ability to select a notification template, send ad-hoc notifications to a device or group of devices from the devices tab using email, SMS, or agent push notifications.  
    · Add/Delete device  
    · Devices import  
    · Revoke device  
    · View Software Inventory - when selected, user is allowed to view a device software inventory. |
| Users | Ability create users and groups. Includes the following permissions:  
  · Add/delete groups  
  · Add/delete users  
  · Edit a user's property  
  · Can manage admin users  
  · Users import - ability to import list of users from a file |
<table>
<thead>
<tr>
<th>Role Based Access Controls (RBAC) Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Enrollment</strong> Access to the Options dialog all functionality related to enrollment, including setting default enrollment modes, configuring enrollment notification servers (SMTP/SMS Gateway), modifying and creating enrollment templates, and sending enrollment notifications. Includes the following permissions:</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Policies</strong> Access to the Policies tab and all features related to defining and implementing policies, such as security and password policies, Exchange ActiveSync policies, app tunneling (Windows and Android), server groups, registry configurations (Windows), configurations, applications access (blacklist/whitelist), Sharepoint policies, and more. Includes the following permissions:</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Files</strong> Access to the Files tab and adding, deleting, and downloading files. Includes the following permissions:</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Applications</strong> Allows access to the Applications tab, where you can upload and define applications and create application categories to organize the apps you want to deploy to users' devices. Includes the following permissions:</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
## Role Based Access Controls (RBAC) Permissions

<table>
<thead>
<tr>
<th>Role</th>
<th>Description</th>
</tr>
</thead>
</table>
| Deployment    | Access to the Deployment tab and all functionality related to device deployment, such as the ability to create, edit, deploy, and delete packages. Includes the following permissions:  
  - Add/delete package  
  - Edit package  
  - Deploy packages |
| Reporting     | Access to the Reporting tab and the ability to run and view Device Manager reports.              |
| About         | Access to the About tab features:  
  - Edit and upload an APNS certificate  
  - Edit XenMobile MDM license  
  - Connections information - provides visibility into server related information, such as security parameters, JVM information, and system health. |
| Options       | The Options feature provides a user access to the Options dialog box and the following features in the Options dialog box:  
  - Role-Based Access Control  
  - LDAP  
  - Mobile Service Provider  
  - ActiveSync Gateway  
  - Network Access Control  
  - AppC WebServices API  
  - GoToAssist  
  - PKI Entity  
  - Scheduling  
  - Security  
  - General service parameters |

**Note:** If you want this role to have access to the Remote-Based Access Control feature, you need to specifically select the Remote-Based Access Control option in the dialog box.
| Restrict Group Access | Allows you to associate groups with the current role. When a group is associated with a role, users in that group can only see devices associated with that group. If a user belongs to more than one group, and some of those groups provide a range or permissions, all permissions related to all groups are merged into the role. |
To import user accounts and properties from a file

You can import user accounts and properties from a specially developed file called a provisioning file, which you can create manually.

**Note:** If you are importing users from an LDAP directory, use the domain name along with the user name in the import file. For example, specify username@domain.com. This syntax prevents additional lookups that will slow the import speed. If importing users to the Device Manager internal user directory, disable the default domain in order to speed up the import process. You can reenable the default domain after the import of internal users completes.

After a provisioning file is prepared, use the Import icon on the toolbar to read the file by following this procedure:

1. From the Users tab toolbar, click Import. The Import a provisioning file window appears.

2. In Provisioning file type, click Users or User Properties. If you click User Properties, you do not create an account.

3. In Provisioning file location, browse to the location of the file and then click Import.
Provisioning File Formats

A provisioning file that you create manually and use to import user accounts and properties to Device Manager needs to have the following format:

For a user provisioning file of a .csv file type, the field separator is the ‘;’. The fields are the following:

user;password;role;group1;group2

Note: Because ‘;’ is used as the separator character, it needs to be escaped if present in string values -> ‘\;’

An example of a user provisioning file content is as follows:

user01;pwd\;01;USER;myGroup.users01;myGroup.users02;myGroup.users.use rs01

in which:

· User: user01

· Password: pwd;01

· Role: USER

Note: Role can only be one of the following: USER, ADMIN, SUPPORT, or DEVICE_PROVISIONING.

· Groups:

  · myGroup.users02

  · myGroup.users02

  · myGroup.users.users01

Note: The ‘.’ character is used as a separator to create group hierarchy, and so this character is forbidden in the groups name.

An example of the file format to provision user attributes is as follows:

user;propertyName1;properyValue1;propertyName2;properyValue2

Note: Because ‘;’ is used as the separator character, it needs to be escaped if present in string values -> ‘\;’

An example of a user attributes provisioning file is as follows:

user01;propertyN;propertyV\;test\;1\;2;prop 2;prop2 value

in which:
Provisioning File Formats

- User: user01
- Property 1:
  - name: propertyN
  - value: propertyV;test;1;2
  
    **Note:** Property attributes must be lower case. The database is case-sensitive
- Property 2:
  - name: prop 2
  - value: prop2 value
Managing Devices

You can manage devices by using the following:

- Tagging devices to identify ownership of the device. You can tag devices with a script or by using the Device Manager web console.

- Adding devices to Device Manager either manually or by using the Device Provisioning tool.

- Locking and unlocking devices by using the Device Manager web console.

- Revoking device certificates to prevent devices from accessing Device Manager.

- Wiping information from devices that includes removing some or all data on the device.
To add a device to Device Manager manually

The Device Manager server repository database stores a list of mobile devices. Each mobile device is defined by a unique serial number and/or IMEI. There are a number of methods to populate Device Manager with your devices:

- Adding devices manually.

- Import a list of devices from a file by using Device Provisioning tool (Windows Mobile and Symbian devices only) or Device Auto Discovery (only available with the Secure Device option).

1. Click New device and then select the device type.
To import a list of devices by using a file

Develop a text file according to the following format by using a utility application such as a text editor, spreadsheet application, or note taker.

<table>
<thead>
<tr>
<th>Element</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number</td>
<td>Device serial number (required if IMEI is not given)</td>
</tr>
<tr>
<td>IMEI</td>
<td>Device IMEI identifier (required if serial number is not given)</td>
</tr>
<tr>
<td>Operating System Family</td>
<td>Required to be WINDOWS, ANDROID, or iOS.</td>
</tr>
<tr>
<td>Property name 1</td>
<td>Optional</td>
</tr>
<tr>
<td>Property value 1</td>
<td>Optional</td>
</tr>
<tr>
<td>Property name (n)</td>
<td>Optional</td>
</tr>
<tr>
<td>Property value (n)</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Many mobile operators or device manufacturers provide lists of authorized mobile devices, and you can utilize these to avoid having to enter a long list of mobile devices manually. Device Manager supports an import file format that is common to all three of the supported device types.

Note the following:

- File charset must be UTF-8/
- Semi-colon (;) is used as the field delimiter so it must be escaped if it is present in the data.
- For iOS device import, Serial Number is mandatory. Serial Number is the identifier for iOS devices.

For example:

1050BF3F517301081610065510590391;15244201625379901;WINDOWS;propertyN;propertyV;test\;1\;2;prop 2;prop2 value
2050BF3F517301081610065510590392;25244201625379902;ANDROID;propertyN;propertyV$*&&ééétest
3050BF3F517301081610065510590393;35244201625379903;iOS;test;
4050BF3F517301081610065510590393;;iOS;test;
55244201625379903;ANDROID;test.testé;value;}
To import a task file

1. Click the Import tab.

2. Browse to the corresponding provisioning file.

3. Click Import.
Viewing the Device Properties

When you click a device name in Device Manager and click Edit, you can view device overview information for a device type. The tabs that appear may differ slightly depending on the device.

The main tabs that appear and the information they contain are as follows:

- **General.** On this tab, you can view device properties, such as the software inventory, the device serial number, IMEI, as well as the Strong ID if the Secure Device option is available in the license installed on the server. You can also display the status of the Device Lock and Device Wipe commands:
  - The statement No device lock/wipe, if no command was sent.
  - A description and the date and time at which the command was sent or carried out.

- **Properties.** The hardware inventory appears on this tab. The list is updated automatically each time the device connects to Device Manager. For devices that use the Secure Device Option, additional tabs appear, such as Certificates and Master Keys.

- **Software.** The software inventory appears on this tab. The list includes all applications and software packages installed on a device, such as package name, author, size, installation date, and version of the software. You must request an inventory if you want to display the applications deployed through Device Manager as well as user-installed apps. To request an inventory, you need to configure a deployment from the Deployment tab. Under Resources to be deployed, select Software Inventory.

  **Note:** For Windows Mobile devices exclusively, only software programs available in the Add/Delete program menu on the device appear on the Software tab.

- **iOS Profiles.** You can view the profiles for an iOS device on the iOS Profiles tab. Profiles may include web clips, mobile device management (MDM) configurations, access permissions, and more.

  **Note:** When working with iOS configuration profiles generated with Apple’s iOS Configuration Utility (IPCU), such as profiles for Exchange ActiveSync, WiFi, and VPN access with a certificate, Device Manager cannot prompt the device unless you include the certificate password in the profile when you create the certificate. You must include the certificate password in the IPCU steps, and then use Device Manager to import the profiles with the certificates.

- **Certificates.**

- **Deployment.** You can view a complete real-time view of package deployment, on a per-device basis, on the Deployment tab. You can view of all packages assigned to a device, and the status of the deployment.

  **Note:** The status of pending is the same as remaining. The status means that the package has not yet been deployed.

- **Connection.** The Connections tab displays the users who have authenticated against a device. It lists the user name, and last two authentication times.
Viewing the Device Properties

- MDM Status. On this tab, you can review the mobile device management (MDM) status for iOS devices. The information that appears is as follows:

  **MDM status:**
  - **INACTIVE.** The server does not expect the device to connect to it any time soon, nor does it consider it necessary.
  - **ENQUEUED.** The server is attempting to communicate with the device, but a push notification has yet to be sent to the Apple Push Notification service (APNs).
  - **ACTIVE.** The server is either currently handling a device request, or it expects the device to reply to a previously sent command.
  - **PENDING.** The server is waiting for a connection from the device.

**Last push initiation.** The time of the most recent push notification initiated by Device Manager.

**Last notification completion.** The time of the most recent completed push notification to the device.

**Note:** The message “Completion of a Push notification attempt” means the notification payload was successfully sent to the server running APNs and the server did not reply with an error (which would indicate syntax errors and so on).

**Last reply device time.** The time that a device connected to Device Manager following a push notification.
Viewing Device Management Status

For each device you manage, Device Manager provides a variety of management status and property information, such as device management status, whether or not the device has been jailbroken, device operating system and hardware information, serial number and IMEI/MEID number, user of the device, device phone number, and so on.

Three of the most commonly used and important statuses for your device indicate whether or not a device is managed or not: Jailbroken/Rooted, SMG Status, and Managed.

The following table describes the status information and icons that you see on the Devices tab in Device Manager:

<table>
<thead>
<tr>
<th>Status</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jailbroken/Rooted</td>
<td>A green light means that the device is NOT jailbroken (iOS) or rooted (Android).</td>
</tr>
<tr>
<td></td>
<td>A red light means that the device has been jailbroken or rooted.</td>
</tr>
<tr>
<td>Secure Mobile Gateway Status</td>
<td>A green light means that Secure Mobile Gateway recognizes the device as legitimate and will allow it to access your Exchange email infrastructure.</td>
</tr>
<tr>
<td></td>
<td>A red light means that Secure Mobile Gateway recognizes the device as a potential threat to your email Exchange email infrastructure and is blocking it.</td>
</tr>
<tr>
<td></td>
<td>A gray light means that your instance of Device Manager does not have Secure Mobile Gateway installed and configured.</td>
</tr>
</tbody>
</table>
| Device Managed | A green light means that the device is managed by Device Manager, which means that the device has the XenMobile agent installed on it and that it is enrolled (and can communicate) with the server running Device Manager. 

**Note:** In some cases, a device will appear as "managed" even though it does not have the XenMobile agent installed. This means that the device has likely been recognized by Device Manager through an ActiveSync connection. For example, if you import users into Device Manager who own a Blackberry or Palm device, or if they connect to their email server through Active Sync, their devices will appear in Device Manager as "managed." Even though these devices cannot have a Device Manager agent installed, their communication with Device Manager is limited, and they cannot have policies deployed to them, it is possible to issue an ActiveSync or Blackberry wipe to them. 

A red light means that the device is not currently being managed by Device Manager for the following possible reasons: 

- If you perform a revoke, wipe, or selective wipe on a device. 
- If the device has an agent installed on it but it was never enrolled. 
- If the device has an agent installed on it but it was never enrolled. |
| Anonymous | Under the User column, a status of 'Anonymous' can occur if a user authentication fails (wrong credentials). 

When this happens, Windows Mobile, Symbian and Windows 8 devices will switch to anonymous mode. It can also happen if the user can't be used to authenticate from a device any more. 

iOS and Android devices authenticate using a client certificate, so those devices will only become 'Anonymous' if the user is deleted or disabled in Active Directory. |
To search for and edit device properties

From the Devices tab in the Device Manager web console, you can search for a device in the list. You can also edit the device properties to add additional properties.

To search for a device

The Search option under the Devices tab is a free-form search field, in which you can search for a device by typing in information you know about a device and you can narrow your search within certain criteria as well.

1. Click the search icon and then specify one or more of the following criteria:
   - The name of one of the device's users
   - The device serial number
   - The device IMEI
   - The model of the device
   - Device platform
   - Operating system version

   **Note:** For each search criteria, you can enter the first letters or numbers of the item you are looking for.

2. To narrow the search to specific criteria, in the Search list, select one or more of the following check boxes:
   - IMEI/MEID
   - User
   - Model
   - Platform
   - OS version
   - Serial number

To restore the complete list of devices, click x next to the Search field.
To edit the device properties

After you have added one or more devices into the repository database, you can populate additional comprehensive device data into the repository database. This ability allows administrators to maintain a detailed hardware inventory of their field devices within Device Manager. This process mirrors that of adding additional user information, minimizing training requirements.

1. Click the Devices tab.

2. Highlight the device to which you want to add additional hardware information and then click Edit.

3. Click the Properties tab and then click New Property.

4. Select either one of the included fields, or select Other to create a custom data field. This field is free form, and can contain up to a maximum of 256 characters.
To show or hide device statuses

Under System Configuration in the Device Manager web console, you can change the parameters of how the devices status appears. In the Devices column, you can also choose which columns to show or hide.

To show or hide device statuses

The following procedure describes how to show or hide the device status for jailbroken or rooted, Secure Mobile Gateway, and Device Manager management statuses.

1. In Device Manager, click Options.
2. In the Options dialog box, click General.
3. Under General Parameters, you can click to enable or disable the following statuses:
   - Highlight “Jailbroken/Rooted” column
   - Highlight “SMG Status” column
   - Highlight ”Managed” column
   - Enable device triangulation
   - Enable WebEAS for iOS

To add or remove device status columns

1. Click the Devices tab.
2. Click the arrow in a status column to show a list of the possible columns that you can display. Each selected item appears in the Devices table.
3. Clear a check box to hide a status column.
Locking a Device Remotely

If the device is lost, but you are not sure it was stolen, you can remotely "lock" the device. To do so, select the device in Device Manager and then on the Security menu, click Lock.

For Android and Windows Mobile devices, the system will then generate a PIN code that will be set in the device if the user had not set a PIN code already. To access the device, the user will have to type that PIN code.

When the device is found again, you can remove the lock by using the Cancel the lock option.
Selectively Wiping a Device

You can perform a selective wipe in Device Manager if you only want to clear corporate data from the device while retaining personal information and selected settings. A selective wipe removes the mobile device management (MDM) profiles. All packages pushed by Device Manager to the device are also removed. The device can be re-enrolled at a future time.

Select Selective Wipe command from the Devices tab > Security menu > Selective wipe. Selective Cancel Wipe to undo the operation request.

Selective Wipe for iOS and Android Devices

Performing a selective wipe from the if you only want to clear corporate data from the device while retaining personal information and selected settings. The MDM profiles and all packages pushed by Device Manager to the hand held are removed. The device can be re-enrolled at a future time.

Note: Selectively wiping an Android devices does not completely disconnect the device from Device Manager and a user’s corporate network. In order to break the connection between the device and the corporate network, you also need to revoke the Android device.

Selective Wipe for Windows 8 Devices

When you perform a selective wipe on a Windows 8 device from Device manager, it will remove all contents from the currently logged on user’s profile folder.

Selective Wipe for Windows Phone 8 Devices

When you selective wipe a Windows Phone 8 device using Device Manager, the following is removed from the device:

- The enterprise token that allows apps to be installed on the device by Device Manager.
- All Device Manager certificates.
- All Device Manager configurations that have been deployed to the device.
If a device is stolen or lost, you can send a request to have all data on a device be erased. For Android devices, this also includes the option to include any memory cards.

To fully wipe a device, from the Devices tab inside the XenMobile Device Manager web console, select Security > Full Wipe.

**Note:** Erasing a device may not complete in full if the “current holder” of the device has time to turn the device off before the content of the memory card is completely deleted. As such, they may still have access to data on the device.

If the wipe of the device is not done and it is retrieved, you can cancel the wipe command by selecting the Cancel wipe menu item.

For Android devices, you can choose to wipe only the device, which removes any internally stored data, or choose to wipe the device, plus any externally connected storage data (memory cards).

For Windows Phone 8 Devices, a full wipe removes all MDM information plus all user data, including all personal content such as apps, emails, contacts, and media files.

For Windows Mobile devices that are not running Windows Mobile 6 or later, after wiping, it may be required to send the device back to the manufacturer to reload the original operating system and/or software.
Tagging User Devices Automatically

You can tag your users’ devices as either corporate-owned or employee-owned to keep track of your company’s Bring Your Own Device (BYOD) program, either automatically with a script, or manually by using the Device Manager web console. To enable employee and corporate device tagging, you will need to download a Microsoft PHP, add device IDs to a CSV file, and execute the given XenMobile scripts that will automate the device tagging process. After setting up the device tagging, you will schedule the script as a repeating Windows Task to run every minute.

**Note:** For on-premise deployments, the tagDevices.php script is located at C:\Program Files (x86)\Citrix\XenMobile Device Manager\samples\WebServices.
To set up device tagging

1. In a browser, go to the Windows PHP download site at http://windows.php.net/download/.

2. Download the installer package named php 5.3 (VC9 x86 Thread Safe (2012-Feb-02 21:56:19).

3. Install the package on your local system at c:\php5.

4. Copy the two files named tagDevice.php and devices.csv to c:\temp (this PHP script is host, location and platform agnostic).

5. Open the tagDevice.php file in a text editor and replace the default information (highlighted) with the following parameters:

   - For an on-site Device Manager implementation:

     ```php
     $soap_url = "<servername>/zdm/services/EveryWanDevice?wsdl"
     $client = new SoapClient(null, array(
        'location' => $soap_url,
        'url' => "<servername>",
        'login' => "demo",
        'password'=> "XXXXX"));
     ```

     For example:

     ```php
     $soap_url = "mdm.zenprise.com/zdm/services/EveryWanDevice?wsdl"
     $client = new SoapClient(null, array(
        'location' => $soap_url,
        'url' => "mdm.zenprise.com",
        'login' => "demo",
        'password'=> "XXXXX"));
     ```

     where mdm.zenprise.com is the name of the Device Manager server and zdm is the Device Manager instance name.

   - For a cloud deployment implementation:

     ```php
     $soap_url = "<instance>.zc.zenprise.com/<instance>/services/EveryWanDevice?wsdl"
     $client = new SoapClient(null, array(
        'location' => $soap_url,
        'url' => "<instance>.zc.zenprise.com",
        'login' => "demo",
        'password'=> "XXXXX"));
     ```

     For example:

     ```php
     $soap_url = "abc.zc.zenprise.com/abc/services/EveryWanDevice?wsdl"
     $client = new SoapClient(null, array(
        'location' => $soap_url,
        'url' => "abc.zc.zenprise.com",
        'login' => "demo",
        'password'=> "XXXXX"));
     ```
6. Edit the devices.csv file and add the serial numbers of all corporate devices, on separate lines.

7. Open a DOS command prompt and cd to c:\temp and run the following command tagDevice.php as follows:

```bash
c:\temp>c:\php5\php.exe tagDevice.php
device:7R043870A4S is a personal asset
device:82835PLWY7K is a personal asset
device:88025X9PA4T is a personal asset
device:880277VSA4S is a personal asset
device:99000052027603 is a personal asset
device:A1000013555FD9 is a personal asset
device:A1000013BB2613 is a personal asset
device:A1000017B0A311 is a personal asset
device:C329030326CC33E is a corporate asset
device:GB0262YCETV is a personal asset
device:GB0289L3ETV is a personal asset
c:\temp>
```

To configure a device tagging script to run as a repeating task

1. Create a file named tagDevice.cmd under c:\temp (where you previously had copied tagDevice.php and devices.csv) and add the following line: cd c:\temp && c:\php5\php.exe tagDevice.php

2. Create an MS Scheduled task to execute this command once every minute (/MO 1). For example: c:\> schtasks /create /TN tagDevice c:\temp\tagDevice.cmd /MO 1

3. Query the tasks to verify that it exists by executing the following command: c:\ schtasks /query /TN tagDevice

4. To delete the task, execute this command: c:\ schtasks /delete /TN tagDevice
Tagging User Devices Manually

There are three ways you can manually tag a device:

- Tag the device during the invitation-based enrollment process (iOS-only).
- Tag the device during the Self Help Portal enrollment process.
- Tag the device by adding device ownership as a device property (any device).

When you enroll an iOS device, you have the option of tagging the device as either corporate- or employee-owned. When using the Self Help Portal to self-enroll a device, you can also tag the device as either corporate- or employee-owned. You can also tag a device manually by adding a property to the device from the Devices tab in Device Manager, creating the property named Device Ownership and choosing either Corporate or Employee.
Working with Apps

You can add apps and files to Device Manager that you want to deploy to Android, and Windows devices. You can add proprietary apps apps you have developed innterally for your users and then depeloy those apps to the Worx Store in a deployment package. You can also add app defintions of publically available apps, so your users can access them from the iTunes, Google Play, or Windows Phone app stores and install them on their devices.

Apps you deploy appear to the iOS and Android device users in the Worx Home Store. Windows Phone 8 users access their apps from the Worx Home app.
Adding iOS Apps

You can add iOS apps to Device Manager and make them available to your users. You can deploy apps to devices using a deployment package. You can make iOS apps available either through the Connect app in the Applications folder, or create an web clip application store to deploy to your iOS users’ home screen.

You can add iOS apps in two ways:

- Internally. Upload the application to the Device Manager database as an iOS .ipk file
- Externally. Create an application definition that references the App data through a URL to the Apple iTunes app store.
To add an internal iOS app

If you have internally developed iOS apps (.ipa) or iOS apps that you have licensed to distribute, you can upload those apps directly to the Device Manager database and then deploy those apps to users’ devices.

1. In the Device Manager web console, select the Applications tab.

2. Click New > New app.

3. In the Import an application into the XenMobile MDM database dialog box, click Choose File.

4. In the iOS app parameters section, enter the following information:

   a. Select the Remove App when MDM profile is removed (Application push only) is you want the app to be removed from any devices you deploy it to if the XenMobile MDM profile is removed from the iOS device.

   b. Select the Prevent backup App data (Application push only) option if you want to prevent the device user from backing up the app to an external device or application.

5. Click Import.
To add an external iOS app

For those iOS apps that are must have, or that you would like to recommend to your iOS users, you can define an iOS app definition and then push the app to your users’ devices. When your users open the Connect app on their device and tap the Applications folder, they can download the app to their devices.

In order to add an external iOS app to Device Manager, you will need the complete URL to the app from the iTunes app store.

1. In the Device Manager web console, select the Applications tab.
2. From the New menu select New > External iOS App.
3. In the Add an external iOS application dialog box, enter the following information:
   a. Specify the URL with a link to the Apple App Store.
   b. Click Go to validate the URL link and retrieve application information.
   c. Next, you can optionally select one or both of these app security policies (under the app description):
      - Remove App When MDM Profile is Removed. To ensure that certain any external apps (those not developed by your organization) are only installed on devices that are managed by your IT department, you can choose to remove a pushed app on iOS device if the user removes their MDM profile.
      - Prevent App Data Backup. Before you push an iOS 5 app to an iOS device, you can select the Prevent Backup of App Data setting, which will prevent allowing the user to backup a specified app either on their computer (via iTunes) or through iCloud.
4. If the app is licensed by the Apple Volume Purchase Program (VPP), then you will see a second tab in the dialog labeled VPP Licenses. To import your VPP license file for this app, select the VPP Licenses tab. (If the app you are defining is not licensed by VPP, then you can skip to step 9.)
5. Click Import a License File.
6. In the Import a License File dialog box, click Choose File to select your VPP license file for the app.
7. Next, from the Country drop-down list, select a country code for the country where the app was developed or localized to. For example, if the app was developed for a French audience, then choose France.
8. Next, click Read a license file to import the file.
9. Click Confirm Import.
10. Click Add. The external iOS app definition is added to Device Manager. You can add as many apps or app definitions as you want to push to your users’ devices. These apps can be pushed to users’ devices when add them to a deployment package.
To add an external iOS app
Citrix Worx Store for iOS Apps (MDM-only)

If you are using XenMobile MDM edition (not the Enterprise solution), you can deploy apps to your users' iOS devices using the Citrix Worx Store for iOS, preconfigured in XenMobile Device Manager as an iOS web clip, and included as part of the iOS base package that gets deployed when a user enrolls into Device Manager. For more information about the iOS base package, see iOS base package.

When you install Device Manager for the first time, you need to add iOS apps to Device Manager, add apps to the Worx Store Deployment Package and then deploy the package to users.
To add apps to Worx Store for iOS

To add apps to the Citrix Worx Store app store for iOS, you need to add iOS apps to Device Manager, and then add those apps to a deployment package and deploy that package to iOS users.

1. In the Device Manager web console, click the Deployment tab, and then click New Package > New iOS Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the Self-service package and then click Next.

3. In the Groups of users window, select the group you created earlier and then click Next.

4. Under Enterprise Application Store, select the apps you want to add to the app store and then click the right arrow to add them.

5. Click Next.

6. In the Deployment schedule window, select the If not deployed Start Now option and then click Next.

7. In the Deployment rules page, click Next.

8. Click Finish.

9. To deploy the Citrix Worx Store app store for iOS, from the Deployment tab, click Deploy. To verify, check the device os an iOS user you deployed the package to and look for the Self-serve app on the device Home screen.
To brand the Worx Store for iOS

You can change the default image used for the Worx store, for example if you want to provide a corporate image or branded logo. You brand the Worx store by creating a branding policy in which you upload your desired image, and then deploy that policy to your users’ devices.

**Note:** Before you begin, make sure you have your custom image ready and accessible.

1. From inside the Device Manager web console, select the Policies tab.
2. From the left side of the console, under MDM Policies, select Branding.
3. Click New Policy.
4. In the Enterprise App Store Branding dialog box, enter a name for the policy.
5. Next, click Browse to select an image to use for the branding.
6. Click Create.
7. Now, to deploy this package to your users’ iOS devices, you need to create a deployment package and deploy it to your user’s devices. For more information, see To create and deploy a deployment package.
To create a deployment policy for iOS apps

1. Select the Deployment tab, click New Package and then click New iOS Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the package, such as iOS App Store, and then click Next.

3. On the Groups of users window, select the group you created earlier and then click Next.

4. On the Resources to be deployed window, under Available Resources, scroll to the Enterprise Application Store, select the apps you want to add, click the right arrow button and then click Next.

5. On the Deployment schedule window, select the If not deployed Start Now option and then click Next.


8. In the packages list, click Deploy.
To push applications to iOS devices

You can push both external Apple iOS apps or apps that you developed internally within your organization to devices from Device Manager. You can select apps you want to push from the Files tab and then deploy the apps in a package.

Device Manager provides two security policy settings that you can apply to applications before you push them:

- **Prevent App Backup.** To ensure that certain external apps are only installed on devices that you manage, you can choose to remove a pushed app on an iOS device if the user removes their Device Manager profile.

- **Remove App When MDM Profile is Removed.** Before you push an iOS 5 app to an iOS device, you can select the Prevent Backup of App Data setting, which will prevent the user from backing up a specified app either on the user’s computer (via iTunes) or through iCloud.

**Note:** Before you can push an iOS app to a device, the app file must already have been imported into Device Manager.

1. On the Files tab, select either an internal or external app and then click Edit or double-click the app.

2. In the app properties dialog box, select:
   
   - if you want the app to be removed if the user deletes the device's Device Manager profile.

   - if you want to prevent the app from being backed up by the user.

3. Click Update.


5. On the Package Name page of the Create New Package wizard, enter a name for the iOS app and then click Next.

6. On the Groups of users page, select the users whom you want to receive the app on their devices when you push the app and then click Next.

7. On the Resources to be deployed page, in Available Resources, select the app you want to push, click the right arrow to add the app to the package and then click Next.

8. On the Deployment schedule page, configure to push the app Now or at a specified time in the future and then click Next.

9. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment and then click Next.

10. On the Package summary page, review the package deployment configuration and then click Finish.
To push applications to iOS devices

11. To deploy the package, select the package and then click Deploy. Connected devices receive the package as soon as scheduling rules are met. Reconnecting devices receive the package as they reconnect, subject to other rule criteria. When you push external apps to a store, the user is prompted to enter the user’s Apple ID credentials in order to install the app.
To remove an app from an iOS device

You can easily remove applications from iOS devices by creating an app removal policy and then deploying that policy to a device. An app removal policy is based upon specifying an application's app bundle ID in the policy. In order to determine an application's bundle ID, you must push a software inventory policy on to a device. Or, you can use the Apple's Xcode IDE application tool called Bundle identifier to find an app's bundle ID.

**Note:** Removing an application will only work for applications distributed by Device Manager.

1. On the Policies tab, click iOS > Configurations.


3. In the App Removal dialog box, select an App bundle ID for the app you want to remove.

4. Click Create. You created the new app removal policy.

5. To deploy the policy as a package, click the Deployment tab.

6. On the Package Name page of the Create New Package wizard, enter a name for the app removal policy and then click Next.

7. On the Groups of users page, select the users from whose devices you want to remove the app and then click Next.

8. On the Resources to be deployed page, in Available Resources, select the app removal policy you want to use for the package, click the right arrow button to add the resource to the package and then click Next.

9. On the Deployment schedule page, configure to push the app Now or at a specified time in the future and then click Next.

10. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment and then click Next.

11. On the Package summary page, review the app removal package configuration and then click Finish.

12. To deploy the package and remove the app, click the package and then click Deploy. Connected devices receive the package as soon as scheduling rules are met.
Distributing iOS Volume Purchase Program Apps

The Apple Volume Purchase Program allows you to purchase iOS apps and books in volume and distribute them to your employees, either as free apps or apps for purchase by using Device Manager.

Distributing apps purchased through the Volume Purchase Program requires following general steps:

- Purchase apps by using your Volume Purchase Program account on the Apple Volume Purchase Program website.
- Download the Volume Purchase Program app purchase license spreadsheet containing the app license redemption codes.
- Add the app information and import the license spreadsheet into the Device Manager file repository.
- Build an app package and deploy it to your users' devices.

Before you begin, make sure you have the following information about the Volume Purchase Program apps you want to distribute with Device Manager:

- App Store Web address for each purchased Volume Purchase Program app you want to distribute.
- Licensing spreadsheet with redemption codes for the Volume Purchase Program apps that will import into Device Manager. You can import license spreadsheets one time, or multiple times if you purchase new licenses at a later date.

After you import Volume Purchase Program license codes, the codes are initially considered to be “Unused.” Each code is reserved and will switch to “Pending” status during deployment as they are sent to devices. The device can determine the following:

- The code is not necessary (for example, the app was already purchased by the specified iTunes account), in which case the code status will be switched back to “Unused.”
- The device can determine that the code was invalid (for example, the code was already used for a purchase), in which case the status switches to “Invalid.”

**Note:** If an app installation fails because the code was invalid, the code is not sent to the device until the next deployment.

- The code is applied successfully to the purchase of the application, in which case the status goes to “Used.”
To distribute Volume Purchase Program apps

1. Click the Files tab and then on the New menu, click New external iOS app.

2. In the new app dialog box, enter the Web address of the Volume Purchase Program iOS app. This is the Web address you used to purchase the app at the Apple app store.

3. Click Go Device Manager locates the Web address for the app and then populates the dialog box with the app details.

4. Click the VPP Licenses tab and then click Import a license spreadsheet.

   Note: You can also update Volume Purchase Program license redemption codes by adding a new license spreadsheets with new licenses at a later date. For example, if you first buy 100 licenses for an app and then eventually have more target devices to deploy to, you can purchase more licenses at the Apple Volume Purchase Program site and then add them to the existing Volume Purchase Program app definition in Device Manager.

5. Browse to the location on your system where you have saved the license spreadsheet and then click OK.

6. Click Add to complete the file configuration.

7. Next, you will create a deployment package so you can push the Volume Purchase Program apps to your users' devices. Select the Deployment tab.


9. On the Package Name page of the Create New Package wizard, enter a name for the iOS Volume Purchase Program app and then click Next.

10. On the Groups of users page, select the users on whose devices you want to receive the Volume Purchase Program app when you push the app and then click Next.

11. On the Resources to be deployed page, in Available Resources, select the Volume Purchase Program app from the External iOS list, click the right arrow button to add the app to the package and then click Next.

   Note: With Volume Purchase Program app deployments, you can only choose to push these apps as External iOS apps, and not as Enterprise App Store apps.

12. On the Deployment schedule page, configure to push the app Now or at a specified time in the future and then click Next.

13. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment and then click Next.
14. On the Package summary page, review the iOS Volume Purchase Program app packaged deployment and then click Finish.

15. To deploy the iOS Volume Purchase Program app package, select the package and then click Deploy. Connected devices receive the package as soon as scheduling rules are met. Reconnecting devices receive the package as they reconnect subject to other rule criteria. When you push external apps to a store, the device user is prompted to enter their Apple ID credentials in order to install the app. You will see a message after installation that the user account on the device will not be charged for the app.

After you deploy the app, the purchased app becomes the property of the iTunes account entered on the device. The owner of the account used to install the app on the device can later install the application on the device of their choosing without having to pay for it.
To run a code inventory report on Volume Purchase Program apps

The Apple Volume Purchase Program code inventory report in Device Manager provides a detailed list of all of your Volume Purchase Program app purchases, the number of purchased licenses for each app, whether or not the license is being used by a device, the associated device ID, and more.

- Click the Reporting tab and then click AVPP code inventory to run the report.

The code summary report results appear in a new tab.
To update new versions of custom iOS apps

A custom app is an app that is not available on iTunes. When a new version of a custom app is available, you can update the app by adding the new .ipa file to the Files tab in Device Manager. The next time the device connects to Device Manager, the app is updated to the new version.

1. On the Files tab, click the iOS app you want to update and then click Edit.

2. To upload an iOS application with a .ipa extension, click Choose File and then browse for the app.

3. Click Update. The new version of the app is pushed to the device the next time the device connects with Device Manager.
Adding Android Apps

You can add Android apps to Device Manager and make them available to your users. You can deploy apps to devices using a deployment package. You can make Android apps available either through the Connect app in the Applications folder.

You can add Android apps in two ways:

- Internally. Upload the application to the Device Manager database as an Android .apk file
- Externally. Create an application definition that references the App data through a URL to the Google Play or Amazon app store.
To add an internal Android app

If you have internally developed Android apps (.apk) or Android apps that you have licensed to distribute, you can upload those apps directly to the Device Manager database and then deploy those apps.

1. In the Device Manager web console, select the Applications tab.

2. Click New > New app.

3. In the Import an application into the XenMobile MDM database dialog box, click Choose File.

4. Select the file and click Open.

5. In the APK parameters section, enter the following information:
   a. Select the Execute APK File option if you want the app to launch immediately after it is installed on the device.
   b. Select After Installation if you want to prevent deleting the installation file from the device when the installation is done.
   c. In the Destination folder section, select the folder where you want to upload the installation file. Only Flash Storage and Device Manager Installation folder are available as a path prefix for Android devices.
   d. In the If the file already exists section, you can specify what to do if the file already exists. You can copy it if the files are different or do not overwrite the existing one.

6. When you are finished, click Import. Once imported, the app can be added to deployment packages and pushed to Android devices.
To add an external Android app

For those Android apps that are must have, or that you would like to recommend to your Android users, you can define an Android app definition and then push the app to your users’ devices. When your users open the Connect app on their device and tap the Applications folder, they can download the app to their devices. In order to add an external Android app to Device Manager, you will need the complete URL to the app from the Google Play or Amazon app store.

1. In the Device Manager web console, select the Applications tab.
2. From the New menu select New > External APK App.
3. In the Add an external Android application dialog box, enter the following information:
   a. In the Application store drop-down, select either Google Play or Amazon and then specify the URL with a link to the app store.
   b. Click Go to validate the URL link and retrieve application information.
4. If for some reason the app URL is not recognized, you can click the Credentials button to authenticate with the Google Play store so your managed devices will be recognized and you can add external Android apps to Device Manager. You will need the device ID from an Android phone that is managed by XenMobile Device Manager.
   a. To obtain an Android device ID from a managed Android phone, type *#*#8255##*#* on the Android device phone number pad.
   b. In the Add an external Android application dialog box, click Credentials.
   c. In the Android Market Access Credentials dialog box, enter your Google username and password. This can be the account ID from any valid Google account.
   d. Enter the Android phone device ID you obtained in step a.
   e. Select the Save in database option to store the authentication with Device Manager.
   f. Click Save.
5. Click Add. The external Android app definition is added to Device Manager. You can add as many apps or app definitions as you want to push to your users’ devices. These apps can be pushed to users’ devices when add them to a deployment package.
Citrix Worx Store for Android Apps (MDM-only)

You can provide your users a list of recommended or required apps on their Android devices. You can add external (free or for cost) apps hosted on Google Play or Amazon Application Store as well as internal, in-house apps that your team has developed. All apps appear inside the Citrix Worx Store for Android apps on your users’ Android devices.

Populating the Citrix an Enterprise Application Store for Android requires performing the following tasks:

- Add the custom-made or external apps from the Android Google Play or Amazon app store to Device Manager.
- Create and push a deployment package containing the apps to the device.
- Brand the Citrix Worx store with your company logo or image (optional)

Users view the apps you deploy on their Android devices by opening the Citrix MDM Connect and then tapping the Apps icon.
To add Android apps to Device Manager

To add external apps to Device Manager, you will need the app Web addresses. For example, you can distribute the following apps to your users:

- Nitrodesk TouchDown For Smartphones (if you have a phone).

- Nitrodesk TouchDown HD for Tablets (if you have an Android tablet).

- Project Viewer.
  https://play.google.com/store/apps/details?id=cintelic.project.pro&hl=en

To add internal Android apps, you only need to upload Android app files with the .apk extension.

1. In Device Manager, select the Files tab, click New and then click External APK App.

2. Click Credentials to add your Google Play Market access credentials. Device Manager uses the credentials to access the proper version of the app you will distribute to your employees.

3. Click Save.

4. In the Add an external APK application dialog box, in URL, enter the app Web address and then click Go button to find the app data.

5. Click Add.

6. Repeat Steps 1 through 5 for the other apps. The apps appear on the Files tab in Device Manager.

7. To upload internal apps to Device Manager, click New and then select New App or File.

8. In the Import a file to the Device Manager database dialog box, click Choose File and then browse to select the Android app file (.apk) to import.

9. Select from the following options:

   - Execute APK file. Select this option to execute the installation automatically when the file transfer is done.

   - After installation. Select this option to avoid deleting the installation file from the device when the installation is done.

   - Destination folder. Write down the folder where the file should be uploaded. Only Flash Storage and Device Manager Installation folder are available as path prefix for Android devices.
To add Android apps to Device Manager

- If the file already exists. Copy it if the files are different or do not overwrite the existing one. You can also decide to register a comment if needed.

10. Click Import.
To brand the Worx Store for Android

You can change the default image used for the Worx store, for example if you want to provide a corporate image or branded logo. You brand the Worx store by creating a branding policy in which you upload your desired image, and then deploy that policy to your users’ devices.

**Note:** Before you begin, make sure you have your custom image ready and accessible.

1. From inside the Device Manager web console, select the Policies tab.
2. From the left side of the console, under MDM Policies, select Branding.
3. Click New Policy.
4. In the Enterprise App Stor Branding dialog box, enter a name for the policy.
5. Next, click Choose File to select an image to use for the branding.
6. Click Create.
7. Now, to deploy this package to your users’ Android devices, you need to create a deployment package and deploy it to your user’s devices. For more information, see To create and deploy a deployment package.
To create a deployment package for Android apps

1. Select the Deployment tab, click New Package and then click New Android Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the package, such as Android App Store, and then click Next.

3. On the Groups of users window, select the group you created earlier and then click Next.

4. On the Resources to be deployed window, under Available Resources, scroll to the Enterprise App Store -, select the check boxes for the two external and two internal apps you want to add, click the right arrow button and then click Next.

5. On the Deployment schedule window, select the If not deployed Start Now option and then click Next.


8. In the packages list, click Deploy.
Adding Windows Phone 8 Apps

You can add Windows Phone 8 apps to Device Manager and make them available to your users. You can deploy apps to devices using a deployment package.

**Note:** Makes sure that before you deploy apps to your users, that the app has been signed by the Microsoft app signing tool (XapSignTool.exe) with a valid app certificate. For more information, see [Configuring Apps for Windows Phone 8 Wrox Home](#).

You can add Windows Phone 8 apps in two ways:

- Internally. Upload the application to the Device Manager database as an Windows .xap file.
- Externally. Create an application definition that references the App data through a URL to the Windows Phone app store.
To add an internal Windows Phone 8 app

If you have internally developed Windows Phone 8 apps (.xap) or Windows Phone 8 apps that you have licensed to distribute, you can upload those apps directly to the Device Manager database and then deploy those apps.

1. In the Device Manager web console, select the Applications tab.

2. Click New > New app.

3. In the Import an application into the XenMobile MDM database dialog box, click Choose File.

4. Select the file and click Open.

5. In the XAP parameters section, enter a description, and then click Import.
To add an external Windows Phone 8 app

For those Windows Phone 8 apps that are either must have, or that you would like to recommend to your users, you can define a Windows Phone 8 app definition and then push the app to your users’ devices. When your users open the Company Store on their device, they can install the app on their devices. In order to add an external Windows Phone 8 app to Device Manager, you will need the complete URL to the app from the Windows Phone app store.

1. In the Device Manager web console, select the Applications tab.

2. From the New menu select New > External Windows Phone App.

3. In the Add an external Windows application dialog box, enter the URL to the app from the Windows Phone app store.

4. Click Go to validate the URL link and retrieve application information.

5. Click Add. The external Windows Phone app definition is added to Device Manager. You can add as many apps or app definitions as you want to push to your users’ devices. These apps can be pushed to users’ devices when add them to a deployment package.
Configuring Apps for Windows Phone 8

Worx Home

Using Device Manager, you can deploy apps to your Windows Phone 8 users by creating an Enterprise Hub policy in Device Manager and then deploying it to your users. On their devices, this policy will appear as the Windows Worx Home app. Once deployed, you can then provide Windows Phone 8 apps for your users using deployment packages.

In order to provide Windows Phone 8 apps from the Windows Mobile app store or from your own development teams, you need to perform the following setup tasks as described on the Microsoft Windows Phone 8 development center web site.

**Note:** Make sure that you follow these setup tasks in the order presented, and ensure that you build the Enterprise Hub and add it to the Windows Phone 8 Base Package in Device Manager before you enroll your Window Phone 8 users, or they will not receive the Worx Home app store and will have to re-enroll to receive it.

The general setup tasks are as follows:

1. Register a company account on Windows Phone Dev Center and acquire an enterprise certificate from Symantec. Visit the Symantec Enterprise Mobile Code Signing Certificate Web site, and complete the required steps to acquire an enterprise mobile code signing certificate.

2. You will receive an application enrollment token (AET) with the .aetx extension. This file will be used in the policy created in Device Manager that you will deploy to your Windows Phone 8 devices.

3. Obtain the Citrix developed Worx Home app store app on the XenMobile Device Manager server at the following location: \Citrix\XenMobile Device Manager\tomcat\webapps\zdm\CitrixWorxHome.xap.

4. Sign the Citrix Enterprise Hub app (CitrixWorxHome.xap) as well as any internally developed apps you want to distribute, using the Microsoft app signing tool (XapSignTool.exe).

5. Using the Device Manager web console, create a Windows Phone 8 Enterprise Hub policy and add the package to the Windows Phone 8 base package located under the Deployment tab.

6. Enroll your Windows Phone 8 users.

7. Optionally, you can add Windows Phone 8 apps you want to distribute to the base package package, or create a separate package to deploy the apps. Apps can be from the Windows app store, or internal apps you develop in-house. Note that for all internal apps, they must be signed with the certificate before you deploy them. For information on how to add Windows Phone 8 apps to device manager, see Adding Windows Phone 8 Apps.
Adding Windows Mobile Apps

If you select a Windows Mobile app (.cab) to be uploaded, several options appear, as described below.

**Note:** This is possible only with signed applications. The installation will silently fail otherwise.

- **Execute CAB file:** select this option to execute automatically the installation when the file transfer is done.

- **Silent installation:** select this option to silently install the application, without prompting the end user. Frequently, for reasons of code signatures, messages may be generated asking the device user to confirm installation of applications. Likewise, by default under Windows CE, messages ask where applications should be installed. Device Manager allows applications to be installed in silent mode without the device user having to reply to confirmation messages.

- **After installation:** select this option to avoid deleting the installation file from the device when the installation is done.

- **Destination folder:** Write down the folder where the file should be uploaded.

- **Specify what to do if the file already exists:** copy it if the files are different or do not overwrite the existing one.

- **You can also decide to register a comment if needed.**
Configuring Applications Access Policies

When you deploy a software inventory package to a device, Device Manager maintains the list of apps. You can work from those lists to configure Applications Access Policies, also known as application blacklists and whitelists to manage your users’ access to applications on their devices.

You can also use the Applications Access Policies in the following ways:

- As triggers for Automated Actions. For example, if Device Manager detects that a device has an unapproved app installed, you can configure an Automated Action that remotely wipes a device, or sends a notification to the user that the user’s device is out of compliance with the organization’s policy.

- To serve as device status flags for the Secure Mobile Gateway rules. For example, if Device Manager detects that a device has an unapproved app installed, you can configure the Secure Mobile Gateway rules to block the device from receiving email from the organization. For more information, see Secure Mobile Gateway Policies and Rules.

Applications Access Policies Types

You can create the following types of Applications Access Policies:

- **Forbidden** (blacklist). A list of apps that users cannot install on their devices. If even one app on device matches an app in the Forbidden list in Device Manager, the device is considered to be in violation of the policy.

- **Suggested** (whitelist). A list of apps that you suggest to users. Users can have one or more of the apps from the list installed and still be in compliance with the policy. However, if users install an app that is not listed in the policy, the user’s device is in violation of the policy.

- **Required** (whitelist). A list of apps that must be installed on the device to be in compliance with the policy. Users must install all of the apps on the list. If users do not install any of the apps in the list, the device is in violation of the policy.

App Definitions

You have the option in Device Manager of using the App bundle ID and App package name when you define iOS and Android apps in your policies. Device Manager can identify apps more reliably, however, when you use these values.

In iOS, an App bundle ID is traditionally a reverse-domain-name style string used when a developer creates a new app. For example, for Angry Birds (www.rovio.com/), the App bundle ID on iOS is ‘com.rovio.angrybirds’. On Android, an App package naming convention is similar to iOS, in which the developer identifies the app with a reverse-domain-name style string. The last part of the name is the name of the App package, often with the file
To configure an Applications Access Policy

1. In the Device Manager web console, click the Policies tab.


3. Click New Applications Access Policy.

4. In the Add a new Applications Access Policy dialog box, enter a name for the policy, such as, Forbidden iOS Apps and then optionally enter a description.

5. In Access policy, click one of the following options:

   - Required (whitelist). Defines a list of apps that users are required to install on their device to be in compliance with the policy. If any of the apps is not installed, the device is in violation of the policy.

   - Suggested (whitelist). Defines a list of apps that are suggested to users. Users can have one or more of the apps from the list installed and still be in compliance with the policy. However, if the user installs any apps that are not listed in the policy, the device is in violation of the policy.

   - Forbidden (blacklist). Defines a list of apps that users should not install on their devices. If any apps on device match an app in the this list, the device is in violation of the policy.

6. In OS type, select the device platform you want to associate with the policy.

7. Click New app.

8. In the Add a new application dialog box, enter the name of an app that you would like to add to the Applications Access Policy list. When you add an app, you can optionally enter the app bundle ID and app package name for iOS and Android. If you configure these fields, Device Manager uses the values to identify the app.

9. Click Create. This will create the application in the list. The app appears in the list in the Add a new application dialog box.

10. Click Create again to create the Application Access Policy. Once created, you can add this policy to a deployment package and deploy to the devices you want to manage.
Application Tunnels

Device Manager Application Tunnels (App Tunnels) are designed to increase service continuity and data transfer reliability for your mobile apps. App Tunnels are used to define proxy parameters between the client component of any mobile device app and the application server component.

Device Manager tunneling acts as a stream buffer to overcome inherent network issues, such as irregular latency or network hopping. Tunneling also provides checkpoint restart capabilities, which is critical when bouncing between cellular data points. Furthermore, Device Manager automatically applies on the fly data compression and AES encryption to all data traffic transiting within each tunnel.

You can assign a tunnel channel dedicated to each mobile app and monitor the apps. For each App Tunnel you define, Device Manager transmits and monitors the data streams in a separate tunnel.

App Tunnels provide the following benefits:

- Security through encryption of data traffic.
- Efficiency through compression of data traffic (can help reduce strain on your device data plan as well as battery usage).
- Reliability through buffering of data traffic. For example, if a device loses connectivity or switches from WiFi to 3G, App Tunnels make sure data traffic is buffered until the connection is reestablished.

**Note:** Each application requires its own tunnel.

This section includes procedures for creating App Tunnels in Device Manager and for creating a remote support App Tunnel specifically for the Remote Support help desk application.
To add an Application Tunnel

1. In the Device Manager web console, click the Policies tab and then under MDM Policies, click the device type for which you want to add an Application Tunnel (App Tunnel).

2. Click Tunnels and then click New tunnel.

3. In the Create a tunnel dialog box, in Name, enter the tunnel name. Citrix recommends the format Application_Name.

4. Select the Remote Support check box if the tunnel will be used for the Remote Support application. If you select this option, some of the options in the dialog box become unavailable. To complete the remote support tunnel configuration, see To create a remote support App Tunnel.

5. Under Connection configuration, in Connection initiated by, click Device if the connection is client-initiated or click Server if the connection is server-initiated. With the exception of Remote Support, App Tunnels are typically client-initiated.

6. In Protocol, click Generic TCP or Active FTP as the tunnel protocol.

7. In Max. connections per device, set the maximum connections, per device, per tunnel. (1 is recommended.)

8. Optionally, set the connection timeout, in seconds. This option allows for App Tunnels to be closed cleanly, even if the app fails.

9. Optionally, choose to use SSL encryption connection between the server running Device Manager and the desktop running the Remote Support application.

10. Optionally, in Secure Connection, select the Use SSL connection check box to block the traffic through that tunnel when the devices are in a roaming situation.

11. Under Application device parameters, click one of the following options to define the mobile application traffic redirection:

   · Through application settings. If you choose this option, you must set 127.0.0.1 in the application server field on the mobile device.

   · Using a local alias. The application on the mobile device will connect to the alias you enter; the alias will be resolved to localhost and intercepted by Device Manager Client Agent. An alias can be any name; for example: my.crm application, exchange server, and so on.

   · An IP address range. Specify a range of IP address targets for which the mobile application will try to connect to in order to make Device Manager to tunnel the connection. For example:

     · From: 0.0.0.0 to 255.255.255.255. In that case, all the traffic from the mobile device is redirected through Device Manager.
To add an Application Tunnel

- From: 88.10.10.10 to 88.10.10.10. In that case, only the traffic toward 88.10.10.10 is redirected through Device Manager.

12. In Client port, enter the port used by the application on the mobile device. This option is required.

13. In Application server parameters, enter the application IP address or server name, and the server port number. These options are required. In most cases, this is the same value as for Client port.

14. Click Create.

**Note:** To properly use an App Tunnel, you need to configure the device-based apps to connect to the Device Manager server rather than to their own server. Usually, 127.0.0.1 (localhost) is specified as the server address. However, some apps may not allow this type of configuration, or it may be preferable not to change the configuration of applications already deployed. In such cases, check the Specify a local alias box and enter the server's name. This name will be redirected automatically to 127.0.0.1 on the mobile devices.
To update or delete an App Tunnel

You can change the configuration settings of an existing tunnel in Device Manager but you cannot change the name of the tunnel.

1. In the Device Manager web console, click the Policies tab and then under MDM Policies, click the device type for which you want to update or delete the app tunnel.

2. In the list of tunnels in the center pane, select the check box for the tunnel you want to edit or delete.

3. Click Edit to change the settings or click Delete to remove the App Tunnel.

4. In the Edit a tunnel dialog box, change the settings and then click Update.
To create a remote support App Tunnel

You need to create a remote support Application Tunnel (App Tunnel) to support the Remote Support help desk application, which allows for the remote control of mobile devices over-the-air through Device Manager.

1. In the Device Manager web console, click the Policies tab and then under MDM Policies, click to expand the device type for which you want to configure a remote support App Tunnel.

2. Click Tunnels and then click New tunnel.

3. In the Create a tunnel dialog box, in Name, enter a name for the remote support app tunnel.


5. Optionally, under Connection configuration, in Connection time-out, select the Define check box and then enter a value in seconds to indicate the interval in which the connection to the Remote Support application should time out.

6. In Secure Connection, select the Use SSL connection check box if you want to configure a secure connection between the Device Manager server and the Remote Support application.

7. In While roaming, select the Block cellular connections passing by check box if you want to block the tunnel while roaming.

   **Note:** WiFi and USB connections are not blocked.

8. Click Create.
Adding Files

You can add script files Device Manager that perform certain functions for your users, or you can add document files that you want your Android device users to be able to access on their devices. You can also specify when you add the file the directory you want the file to be stored at on the device; for example if you want your Android users to receive a company document or .pdf file, you can deploy the file to the device and let your users know where the file is located.

You can add the following file types on the Files tab in Device Manager:

- Text-based files (.xml, .html, .py, and so on)
- Any other file such as documents, pictures, spreadsheets, or presentations
- Mortscript files (files with .mscr extension) on Windows Mobile devices
To add a file to Device Manager

To deploy a file to a device, you first need to upload the file into the Device Manager repository database. When you add the files to deployment packages, you can deploy the files to users' devices. You can add file types, such as documents, images, videos, presentations, and .pdfs.

Note: You cannot add files to iOS devices.

1. In the Device Manager web console, click the Files tab and then click New file.

2. In the Import a file to the XenMobile MDM database dialog box, browse to the file on your computer, click Open and then click Import

You can now deploy the file to a device in a deployment package.
Uploading Other Files

If you select a file type that has no ".cab" or ".mscr" extension, you will be presented with several options.

- Macro must be replaced. Select this option if you want to search and replace the macros inside the scripts.
- Destination folder. Write down the folder where the file should be uploaded.
- Specify what to do if the file already exists. Copy it if the files are different or do not overwrite the existing one.
- Specify if you want to set the Read-Only option.
- You can also decide to register a comment if needed.
Configuring Macro Substitution for Uploading Files

If you want to upload files and Mortscript script files, select the Macro must be replaced box. When you select this check box, it indicates that the script file must be checked before deployment to substitute macros with dynamic values that depend on the user and device.

An example of a macro in a Mortscript script is as follows:

```
result = Question( "Hello world, mail = %{ user.mail? | protect('\'', '\''') | encode('UTF-8') }%!", "It's a start!", "YesNo")
```

The macro is located between the `%{...}%` tags. The server does not detect in advance the file format or its encoding. Instead, the files are analyzed in binary format. The “%”, “{”, ... are checked in ASCII format. Those are compatible with most characters encoding (but not UTF-16).

The macros are made of several parts separated by the pipe character: xxx | yyy | zzz:

- The first element indicates a property: `user.xxx` for a user property, or `device.xxx` for a device property. You can use the `?` character to indicate to use an empty string if the property is not defined.

- The next elements are filters used to encode, transform and mostly protect the string so that it can enter the file context.

In this example, the elements are: `user.mail? | protect('"', '"') | encode('UTF-8')`

- `user.mail?`: insert the “mail” property of the user. Use an empty string if it is empty or not defined.

- `protect('"', '"')`: protect the quote character by adding another quote before it. This protection is specific to Mortscript that requires to double that character when one wants to define one. For instance: “This is a quote “” in Mortscript”.

- `encode('UTF-8')`: encode the string in UTF-8.

The last part is important since only a binary element can be inserted by Device Manager:

- `user.mail?`: give a character string.

- `protect('"', '"')`: takes a string in input and outputs a string.

- `encode('UTF-8')`: takes a string in input and outputs a binary array.

Note: Use other encodings if necessary, like CP1252 in France if you want to use accentuated characters.
This other available functions are:

- **S64Encode**: \([\text{binary} \Rightarrow \text{text}]\) encodes a binary in S64.
- **S64Decode**: \([\text{text} \Rightarrow \text{binary}]\) decodes a S64 string to binary.
- **B64Encode**: \([\text{binary} \Rightarrow \text{text}]\) encodes a binary in Base64.
- **B64Decode**: \([\text{text} \Rightarrow \text{binary}]\) decodes a Base64 string to binary.
- **encode(encoding)**: \([\text{text} \Rightarrow \text{binary}]\) encodes a string with a specific encoding.
- **decode(encoding)**: \([\text{binary} \Rightarrow \text{text}]\) decodes a binary array to a string.
- **protect(searched_character, protection_character)**: \([\text{text} \Rightarrow \text{text}]\) insert the protection_character before the searched_character.
- **transform(searched_string, replacement_string)**: \([\text{text} \Rightarrow \text{text}]\) replace searched_string by replacement_string.

You can use the following user properties in the scripts:

- cn
- company
- companyname
- property_country
- department
- description
- displayname
- distinguishedname
- facsimiletelephonenumber
- givenname
- homecity
- homecountry
- homefax
- homephone
- homestate
- homestreetaddress
- homezip
When users connect to an LDAP directory, most of these properties are completed automatically. You can also manually enter the properties in the user properties dialog.

You can use the following device properties:

- `ew_version`
- `ew_revision`
- `cpu_clock_speed`
- `sim_id`
- `memory`
- `freedisk`
- `tel_number`
- `system_oem`
- `system_platform`
Configuring Macro Substitution for Uploading Files

- cpu_type
- system_os_version
- system_os_build
- memory_available
- total_disk_space
- system_language
- user_language
- screen_width
- screen_height
- screen_nb_colors
- main_battery_percent
- backup_battery_percent
- battery_charging
- external_storage1_name
- external_storage1_total_space
- external_storage1_free_space
- external_storage2_name
- external_storage2_total_space
- external_storage2_free_space
- user_defined_1
- user_defined_2
- user_defined_3

You can also use any custom property defined for the device or user.
Examples of Simple Mortscripts

Ask for Reboot Script Example

```mortscript
Result=Question("Your device needs to be rebooted. Do you want to reboot now?", "Hello %user.name? | protect("", ") | encode('UTF-8') ")!", "YesNo")

If ( Result=YES)
    Reset
EndIf

This script opens a dialog box with the Yes and No buttons. It asks for the user to reboot now or later. If Yes is pressed, the device will reboot. If No is pressed, nothing happens except that the dialog is closed.

The title of this dialog displays the name of the user, as stored in a custom property of the user.
```

Data Upload Script

Here is an example of a Mortscript script used to upload a file to a FTP server:

```mortscript
```

This simple script will upload the file test.zip, located in the directory “My Documents” of the device, to the server ftp.mydomain.com, in the directory “incoming”. The file will be renamed file.zip. It will use the login “test”, and password “test”, and use the passive mode of the ftp protocol.

The synopsis of the FtpUpload function is the following:

```mortscript
```

**Note:** In this Device Manager release, the FtpUpload function is not yet part of the standard Mortscript program but is only available in the Device Manager release of MortScript.
To upload a MortScript file

You can add MortScript (.mscr) files to Device Manager to deploy to Windows Mobile devices. MortScript is a batch scripting language that allows you to perform basic functions, such as opening or closing apps, running processes, creating directories, establishing or closing network connections, and other basic device functions.

1. Click the Files tab and then click New file.

2. In the Import a file to the XenMobile MDM database dialog box, browse to the MortScript file on your computer and then click Open.

3. Enter the following script parameters:
   a. Execute script. Select this option to execute the script automatically when the file transfer is done.
   b. Macro must be replaced. Select this option if you want to search and replace the macros inside the scripts.
   c. Specify what to do if the file already exists. If the files are different, you can choose make a copy or to not overwrite the existing one.

   **Note:** You must encode MortScript files by using ANSI character set if possible. Unicode is also supported with proper prefixes.

4. Click Import.
Creating Policies

You create and configure Device Management policies on the Policies tab in the web console that you can push or made available to devices. You need to put the policies in a package for further deployment.

The way you configure the devices depends on the device operating system. To create a policy, on the left-hand menu, select the desired configuration option for a given platform. Then, click New to create the new policies or configurations.
Scheduling Connections to Device Manager

Scheduling provides essential control over devices that are subject to compliance rules. The schedule feature directs the device to automatically connect to the server running Device Manager at predetermined intervals. During these connections, a policy audit automatically occurs and missing or modified policies are automatically reapplied. Additionally, scheduling ensures that Device Manager has the most up-to-date device information available.

Note: Flexible scheduling is available on Android and Windows Mobile devices only. iOS devices use a predetermined schedule defined by the iOS operating system.

The Scheduling Wizard is located in the Device Manager web console under the Policies tab. Scheduling provides Registry keys for managing scheduled connections between a device and the server. This is useful for devices that require the ability to connect back for data synchronization between a Line-of-Business, ERP, or CRM-type system.

To define an hourly range in the scheduling table, you can either click in a square or you can drag and drop with your mouse to define a range. (First left-click on a square and then, keeping the button pressed, move the cursor over another square and release the button.)
Managing SharePoint Configurations

Citrix data loss prevention (DLP) solution enables access from your mobile workforce to your SharePoint content. You can apply access control rules to content to prevent unauthorized usage depending on document classification. In XenMobile, you can use the DLP-SharePoint/Encrypted Email Attachment Viewing feature to manage SharePoint configurations. If the SharePoint configuration item is unavailable, your license does not include the SharePoint feature. To check your license features, view the About tab in the web console.

If you are planning to use the Device Manager SharePoint access management feature, make sure your deployment meets the following Windows requirements:

- SharePoint 2010 or Office 365.
- Windows 2008 R2 - SharePoint 2010 SP1 is required or KB976127.
- Windows 2008 - Rest API calls will fail unless KB976217 is also installed.

**Note:** Make sure that your SharePoint folders on the SharePoint server do not use special characters such as commas (,), semicolons (;), or periods (.), or those folders will not appear on your users' devices.
To configure a SharePoint resource configuration site in Device Manager

When you configure a SharePoint resource configuration site, you define the SharePoint server settings and specific directories (folders) that you want to expose to the device user.

**Note:** Make sure that your SharePoint folders on the SharePoint server do not use special characters such as a comma(,), semicolon(;), or period(.), or they will not appear on your users’ devices.

1. In Device Manager, click the Policies tab.
3. Click New Configuration.
4. In the Create a resource configuration dialog box, on the Site/Folder Config tab, enter a name for the SharePoint site and then configure the following:
   - Name. The name of the resource definition.
   - Description. A free text description describing the resource.
   - Site. Enter the SharePoint site Web address.
   - Doc Library/Folder. Enter the list of path relative to the base site that you want to publish.
   - Include Sub-folders. Enables the access to sub folders of the above defined path.
   - Document Control. Check all the document controls that are applied to the doc libraries.
5. In the Options dialog box, configure extra options you would like to apply to the SharePoint document folder on your users’ iOS devices, such as if the documents should be wiped from the device if the device is jailbroken, encryption and annotation of documents, and so on and then click Create.
To configure a SharePoint document control policy

You can enable your users to securely access corporate SharePoint content. You apply access control rules, on the content in order to prevent unauthorized usage or actions, depending on your company policy and document sensitivity. You can view this content on the Connect agent on a device in the Documents -> Shared Docs folder. You create a SharePoint control policy to define explicitly what a device user can and cannot do with documents in their secure document container on their devices, such as whether or not documents can be printed, if a user can copy and paste to and from documents, if document check in/check out is allowed, and so on.

1. Under XenMobile Policies, under SharePoint, click Control Policy.
2. Click New Control Policy.
3. In the New Control Policy dialog box, on the General tab, enter a name for the policy.
4. On the Document Control tab, you can set the control policies for all the documents in the folders specified in your SharePoint resource configuration.
5. You can define the following controls that will be applied to the documents by selecting them. Any options left unselected will not be allowed by document users.

- Allowed features
  - Document synchronization. Allow the document to be synchronized to the device. If not checked, the document is only accessible online.

    Note: If you want to be able to annotate PDF files on your device, you need to make sure this option is selected, since the PDF annotation tools only work with locally synced documents.
  - Copy/Paste of content. Allow copy/paste of document content.
  - Email link to document. Enable users to send a link to this document via email.
  - Email document. Allow users to send this document via email.
  - Print document. Allow users to print this document.
  - Document check in. Allow users to check in this document from SharePoint.
  - Document check out. Allow users to check out this document from SharePoint.
  - Open document in another application. Allow users to open this document in a third-party application on the device. If not checked, only the internal viewer can be used.
  - Time expiration
To configure a SharePoint document control policy

- Expires on a date. Specify a date after which the document is not accessible. If on the device, it will be deleted.

- Expires after x Days. Specify the duration of validity of the document. After the specified period, the document is not accessible.

- Authentication expiration

  - Specify an authentication timeout. If the user does not authenticate regularly to SharePoint, the documents become inaccessible.

6. Click Create.
Configuring SharePoint on Android Devices

To configure a SharePoint data loss prevention (DLP) connection for Android, you need to do the following:

- Create an application tunnel that the SharePoint server will use to communicate securely with the device; use a client port.
- Create a SharePoint resource configuration that configures the SharePoint site server address; be sure to use the client port configured in the application tunnel.
- Create a SharePoint policy to configure the security and access parameters for the SharePoint site.
- Deploy the policy to the device.

To create the application tunnel

1. On the Policies tab, click Android, click Tunnels and then click New Tunnel.
2. In the Create a Tunnel dialog box, enter the following app tunnel parameters:
   - Name. Give the app tunnel a name that indicates it is going to be for a SharePoint connection.
   - Application Device Parameters Client Port. The port number that will be used by the XenMobile client application on the device.
     
     **Note:** You will need to use this same port when you configure the SharePoint resource.
   - Application Server Parameters
     - IP address or server name. Address of the SharePoint server.
     - Server Port. SharePoint server port.
3. Click Create.

To create a new SharePoint control policy

A SharePoint control policy defines a set of actions that the user will be able to execute on documents. Document Control allows you to define all the features that will be applied to the documents:

1. On the lower-left, click Control Policies and then click New Control Policy.
2. In the Create new control policy dialog box, on the General tab, enter a name such as “Android Employee SharePoint Documents.”

3. On the Document Control tab, configure the following settings:
   - Allowed features
     - Document synchronization. Allow the document to be synchronized to the device. If cleared, the document is only accessible online.
       
       **Note:** If you want to be able to annotate PDF files on your device, you need to make sure this option is selected, since the PDF annotation tools only work with locally synced documents.
     - Copy/Paste of content. Allow copy/paste of document content.
     - Email link to document. Enable the user to send a link to this document via email.
     - Email document. Allow the user to send this document via email.
     - Print document. Allow the user to print this document.
     - Document check in. Allow the user to check in this document from SharePoint.
     - Document check out. Allow the user to check out this document from SharePoint.
     - Open document in another application. Allow the user to open this document in a third-party application on the device. If not selected, only the internal viewer can be used.
   - Time expiration
     - Expires on a date. Specify a date after which the document is not be accessible. If on the device, it will be deleted.
     - Expires after x Days. Specify the duration of validity of the document. After the specified period, the document is not accessible.
   - Authentication expiration
     - Specify an authentication timeout. If the user does not authenticate regularly to SharePoint, the documents become inaccessible.

4. Click the Tag Mapping tab to enable deeper integration to your SharePoint libraries by applying document controls based on already defined tags on your SharePoint documents.

5. Click Create.

---

**To create a SharePoint resource configuration**

A SharePoint resource configuration defines a SharePoint document library access and the control policies that are tied to its documents.
1. Click the Policies tab and then from the left side, click SharePoint Resource Configuration.

2. Click New Configuration.

3. In the Create a Resource Configuration dialog box, enter a name and description of the SharePoint server connection.

4. Enter the SharePoint server name plus the port number you configured in your application tunnel for the Client Port (2500, for example).

5. Under Document Control, select the Control Policy you created. Other options:
   - Include Sub-folders. Enables the access to sub-folders of the above defined path.
   - Document Control. Check all the document controls that are applied to the document libraries.

6. Click Create. The new SharePoint server appears as a resource in Device Manager. Now the SharePoint server and resource you configured is ready to be accessed by users.

To deploy the SharePoint resource to your device

To enable your client users to access the content in this SharePoint site, you need to create a deployment package that contains the SharePoint resource and then push that deployment to your device. Once on your device, you can launch the client application and access the documents folder to view the documents contained on the SharePoint server. Now, you will create a deployment package and push the new Android SharePoint resource to your Android device.

1. Click the Deployment tab, click New Package and then click New Android Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the SharePoint package (such as Android SharePoint Package) and then click Next.

3. In the Groups of users window, select a group to which you want to deploy the SharePoint package to and then click Next.

4. In the Resources to be deployed window, under Available Resources, scroll to the SharePoint folder, select the SharePoint Configuration you created in the last step and then click the right arrow to add the resource to the deployment package.

5. Scroll in the Available Resources list and then on the Tunnels folder, click the application tunnel you created for your Android SharePoint configuration.

6. Click the right arrow to add the resource to the deployment package and then click Next.

7. In the Deployment schedule window, select the If not deployed Start Now option and then click Next.


10. From the Packages list, click Deploy.

When the deployment has finished, select the deployment package, and then click Details to see information about the success of the package deployment. When the package shows as deployed, you can check the success of your deployment. Select the deployment package, open the Connect client on the Android device and then tap the Documents folder. From here, users can open documents from the SharePoint site.
Managing iOS Configurations

You can create a variety of policy types and configurations for your iOS devices to help manage user and company data security, including passcode policies, general iOS restrictions policies, App Tunnel configuration policies so your users can securely access your company intranet, email policies so your users can seamlessly connect to corporate email accounts, app distribution policies so you can make useful apps available to your users, app removal policies to revoke unauthorized or out of date apps, and much more.
To configure iOS security policies

The following procedures describe how to configure three basic security policies in Device Manager for iOS users: passcode policies, general restrictions policies, and an App Tunnel policy to enable users to securely access their organization’s intranet.
To create an iOS passcode policy

1. On the Policies tab, under iOS, click Configurations.
2. In the New Configuration menu, click Profiles and Settings > Passcode.
3. In Passcode, on the General tab, enter a name for the policy and then configure the policy removal settings.
4. On the Policy tab, configure your iOS passcode policy according to the standards of your IT department. The passcode policy options are as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Require a code on the device</td>
<td>Enables passcode protection on the device. If cleared, the device does not require a passcode on the device (unless the device user sets it manually).</td>
</tr>
<tr>
<td>Allow simple values</td>
<td>Allows the use of a simple passcode, which is defined as a passcode containing repeated characters, or increasing (bottom up) or decreasing (top down) characters (such as 123 or CBA).</td>
</tr>
<tr>
<td>Require alphanumeric values</td>
<td>Requires that at least one character of the passcode is a letter.</td>
</tr>
<tr>
<td>Minimum length codes</td>
<td>Allows you to set the minimum overall length (in characters) required for the passcode.</td>
</tr>
<tr>
<td>Allowed minimum non-alphanumeric characters</td>
<td>Allows you to set the minimum amount of numerical characters required of the passcode.</td>
</tr>
<tr>
<td>Maximum passcode age (1-730 days, or none)</td>
<td>Allows you to specify the number of days for which the passcode can remain unchanged. After the set number of days, the user is forced to change the passcode before the device is unlocked.</td>
</tr>
<tr>
<td>Auto lock (1-5, 10 or 15 minutes or none)</td>
<td>Allows you to specify the number of minutes for which the device can be idle (without being unlocked by the user) before it gets locked by the system. When this limit is reached, the device is locked and the user must enter the passcode.</td>
</tr>
<tr>
<td>Codes History (1 to 50 codes or none)</td>
<td>Allows you to specify that when the user changes the passcode, it has to be unique within the last N number of entries in the history.</td>
</tr>
</tbody>
</table>
To configure iOS security policies

| Grace period before device lock | Allows you to set the maximum grace period, in minutes, to unlock the phone without entering a passcode. Default is 0, (no grace period), which requires a passcode immediately. |
| Maximum failed attempts         | Allows you to specify the number of allowed failed attempts to enter the passcode at the device's lock screen. When this number is exceeded, the device is locked and must be connected to its designated iTunes in order to be unlocked. |

5. Click Create. The new policy appears in the Policies list.

To create an iOS restrictions policy

iOS restrictions enable you to control a wide variety of options and actions on an iOS device, such as restricting voice dialing and camera usage, limiting Siri usage and YouTube, document syncing to iCloud, and more. Where indicated, some features are only supported on iOS 6 and require that the device be placed into Supervised mode with the Apple Configurator.

1. On the Policies tab, under iOS, click Configurations.

2. In the New Configuration menu, click Profiles and Settings > Restrictions

3. In Restrictions Configuration Creation, on the General tab, enter a name for the policy.

4. On the Restrictions tab, enter the following information:

   Tip: Any option for which you select Allow means that the user can perform the operation or use the feature. For example:

   · Allow installing apps. If selected, the App Store is enabled and its icon is available from the Home screen. Users can install or update their applications. If cleared, the App Store is disabled and its icon is removed from the Home screen. Users cannot install or update their applications.

   · Allow use of camera. If selected, the user will be able to use the camera on their iOS device. If cleared, the user cannot use the camera on their iOS device.

   Note: The following restrictions work only on iOS 6 devices and require that the device is placed in Supervised mode with the Apple Configurator.

   · Allow Game Center. If selected, the Game Center will be available on the iOS 6 device. If cleared, the Game Center is inaccessible and the icon removed from the device desktop.

   · Allow UI configuration profile installation. If selected, users can install configuration profiles and certificates on their devices.

   · Allow iMessage. If selected, users can use iMessage.

5. On the Applications tab, enter restrictions you would like to set for default iOS apps.
To configure iOS security policies

6. On the iCloud tab, enter restrictions you want to configure for Apple iCloud.
7. On the Security tab, enter any security restrictions.
8. On the Ratings tab, enter settings for the type of content you want to allow on iOS devices.
9. Click Create. The new policy appears in the Policies list.

To create an iOS app tunnel

When you create an app tunnel, you ensure that all data flows securely for your applications through a server you specify, such as a server in the internal network. App tunnels are used to define proxy parameters between the client component of any mobile device applications and the application server component.

1. On the Policies tab, under iOS, click Configurations.
2. Under iOS, click the tunnels icon and then click New Tunnel.
3. In the New Tunnel dialog box, enter the following information:
   - Name. Enter a name for the tunnel.
   - Maximum connections, per device.
   - Connection timeout. Allows for app tunnels to be closed cleanly, even if the application fails to do so.
   - Client port. Enter the port used by the application on the mobile device.
   - Application server parameters. Enter an IP address or application server host name.
   - Server port. Enter a server port number. In most cases, this is the same value as for the Client port.
4. Click Create.
To configure network policies for iOS

You can configure WiFi and VPN policies in Device Manager for iOS devices so your users can securely access the internal network in your organization and use resources wirelessly.

To create an iOS WiFi policy

1. On the Policies tab, under iOS, click Configurations.
2. On the New Configuration menu click Profiles and Settings > WiFi.
3. In the WiFi Configuration dialog box, on the General tab, enter a name for the policy.
4. On the WiFi tab, enter the WiFi SSID and encryption type.
5. On the Protocols tab, enter the protocol type used by your the WiFi in your organization.
6. On the Authentication tab, enter the user name and password to be used to log on to the wireless network in your organization.
7. On the Trust tab, if you are using custom trusted certificates for authentication, upload the certificates.
8. On the Proxy tab, if you are using a proxy server to route traffic through the WiFi, configure the proxy settings.
9. Click Create.

To create an iOS VPN policy

1. On the Policies tab, under iOS, click Configurations.
2. On the New Configuration menu click Profiles and Settings > VPN.
3. In the VPN Configuration dialog box, on the General tab, enter a name for the policy.
4. On the VPN tab, enter the VPN configuration for your company VPN network.
5. On the Proxy tab, if your VPN network uses a proxy server, configure the proxy server.
6. Click Create.
To create an app lock policy for iOS 6 devices

App lock, or Kiosk Mode, enables you to set an iOS device to only run a single application. When pushed to a device, the policy runs only the app that is specified, disables the home button, returns the device to the specified application automatically upon wake or reboot.

Note: This feature works only on iOS 6 devices and requires that the device to be placed into Supervised mode with the Apple Configurator.

1. On the General tab, enter the Identifier of the policy, the display name, and a description.

2. In Allow profile removal operation, click on of the following options:
   - Always: This option allows the profile to always be removable.
   - Authentication: Allows you to enter a required password that is used when profile is removed. Requires a password
   - Never: Prevents the profile from ever being removed.

3. Select the Allows you to select a specific date check box to specify a date you want to remove the profile.

4. Select the Duration until removal (in days) check box to enable you to set a period of time after which the profile will automatically be removed.

5. On the App Lock - Configuration tab, enter the app bundle identifier of the app you want to use and then click Create.
To create a profile removal policy for iOS

If you want to remove an application profile for iOS from a user's device, you need to create a profile removal policy in Device Manager and then deploy the policy to the device.

1. On the Policies tab, click iOS > Configurations.


3. In Add a profile removal operation, enter the Profile ID of the app profile. This is found in the profile's General tab. You can find the profile ID on the profile General tab.

4. Click Create.

Deploy the policy to a device.
To configure automatic profile removal for iOS 6 devices

For iOS 6 devices, you can configure automatic profile removal in Device Manager. You can configure profiles to be removed automatically at a specified date, to be removed manually by the user with password authentication, or never to be removed.

In the iOS 6 profile dialog (e.g., APN configuration creation), at the bottom of the General tab, you can configure the automatic profile removal settings:

1. Click New Configuration and then click Profiles and Settings.

2. Select a profile type; for example, APN.

3. In the iOS 6 profile dialog box, on the General tab, configure the automatic profile removal settings as follows:
   a. In Allow profile removal operation, select one of the following options:
      - Always. Allows the profile to always be removable.
      - Authentication. Allows you to enter a required password that is used when profile is removed. Requires a password.
      - Never. Prevents the profile from ever being removed.
   b. Select the Automatic Removal Date check box if you want to select a specific date on which to remove the profile.
   c. Select the Duration until removal (in days) check box to specify a set a period of time after which the profile will automatically be removed.
4. Click Create.
To configure geo-tracking on iOS devices

If the iOS devices you manage in Device Manager have a location services policy applied and you configure geo-tracking, you can view the locations of the device over the time you configured in the location services policy. Geo-tracking enables you to track an iOS device over periods of up to six hours at a time. You can view the geographical location of a device and its movement and you can view the device location on Google Maps. If you want to specify individual parameters for your GPS tracking (as opposed to just activating it), you need to deploy a geo-tracking policy. If you choose to keep the default values, you can enable tracking immediately.

1. Click the Policies tab and then click iOS > Configurations.

2. On the New Configuration menu, click Profiles and Settings and then click Location Services.

3. In the Locations Services - Configuration creation dialog box, enter the following information:
   - Name. Enter a name for the location services policy.
   - Description. Provide an optional description for the policy.
   - Location fix timeout. Enter the time Device Manager waits before timing out if the device location cannot be fixed. If nothing is set, Device Manager attempts to locate the device according the Poll interval you specify.
   - Tracking duration. Enter the period of time that the device will be tracked after an Enable Tracking command is sent to the device. The maximum setting is six hours.
   - Poll interval. Enter a value for how often Device Manager will attempt to fix a location on the device. If the device cannot be located, the attempt to locate the device will time out according to the Location fix timeout setting.
   - Accuracy. Set the accuracy of the location point from the device.
   - Report if location services are disabled. Select to enable the device to report that GPS is disabled to Device Manager and Device Manager will display the status of the device.

   **Note:** You must enable the setting if you are using Automated Actions to trigger an action based upon a location-based trigger, such as Location Perimeter Breach or Location Services Disabled. Also, the device must be contacted for its location to trigger the action.

4. Click Create.

5. To enable tracking of the device according to this configuration, you need to deploy the package to the devices you want to track. Next, deploy the package to the iOS device users to implement the policy.

6. To track the device, do one of the following:
To configure geo-tracking on iOS devices

- On the Devices tab, select the device, right click, and select Security -> Enable Tracking.
- Select the device and from the Security button click Enable Tracking.

To view a device's geo-tracking

1. On the Devices tab, select the device you want to view and then click Edit.

2. Click the Geo-tracking tab. Each point on the map indicates when Device Manager fixed the location of the device. A green point indicates the first location point when tracking started. A red point indicates the last device location point captured before tracking ended. You can mouse over each point to see more detailed geographical information. To see a longer range of tracking points, for example, if the device was tracked several times, you change the Display Points From date range and then click Filter.
To create a geo-fencing policy for iOS devices

Geo-fencing in Device Manager allows you to define a geographic perimeter for an iOS device. You can then perform a selective or full wipe upon the breach of the perimeter you set. The policy also notifies Device Manager and the device user when the device has moved beyond the defined radius of the policy. You have the option of setting a delay before the device is wiped, which can give the user time to return to the allowed GPS location perimeter.

1. On the Policies tab, under iOS, click Configurations.

2. In New Configuration, click Profiles and Settings, and then click Location Services.

3. In the Location Services - Configuration Creation dialog box, on the General tab, enter a name for the new profile configuration and configure the following settings:
   - Name. Enter a name for the location services policy.
   - Description. Provide an optional description for the policy.
   - Location fix timeout. Enter the time Device Manager will wait before timing out if device location cannot be fixed. If nothing is set, Device Manager attempts to locate the device according the defined Poll interval.
   - Tracking duration. The period of time that the device will be tracked once an Enable Tracking command has been sent to the device. Maximum is six hours.
   - Poll interval. Enter a value for how often Device Manager will attempt to fix a location on the device. If the device cannot be located, the attempt to locate will time out according to the Location fix timeout setting above.
   - Accuracy. Enter the accuracy of the location point from the device.
   - Report if location services are disabled. Select if you want the device to report that GPS is disabled to Device Manager and the server will show the status of the device.

4. Click the Geo-fencing tab and then set the following parameters:
   - Radius. Select this option to define the radius of the geo-fence. The default value represents the smallest allowable radius for this feature, which is approximately 164 feet, or 50 meters. Enter a small value; for example, 150 feet.
   - Center Point Latitude. For example, 37.787454.
   - Center Point Longitude. For example, 122.402952.
   - Device Notification on perimeter breach. Select this option so that the device user is notified when the device has breached (has gone outside of) the defined perimeter radius.
To create a geo-fencing policy for iOS devices

- Delay on Wipe. Enter 2 minutes as the time allowed before the device becomes wiped of its corporate data and apps.

5. Click Create.

After you create the policy, you need to deploy it to your iOS devices. In the Devices tab, when you click Deploy, the following actions take place:

- All deployment packages targeting the device are deployed.
- Device inventory, properties, and usage data is refreshed.
To store iOS user password

If you want to ensure that iOS users have their passwords stored for ongoing authentication, even if they log out of the Connect agent, you can configure that setting in the Options dialog box in the Device Manager console.

When enabled, Device Manager securely stores a user’s password that may be used for ongoing authentication with the Device Manager server, such as if the user logs out of the agent.

When disabled, Device Manager will not store users’ passwords and will use a certificate for all ongoing authentication with Device Manager. Note that when this setting is enabled, you may still allow users to register and authenticate with a domain password since an enrollment invitation will override this setting when other enrollment modes are configured.

1. Click on the Options dialog box in the Device Manager console.

2. In the left pane, click iOS.

3. Click Store User password settings.
To place a device in supervised mode by using the Apple Configurator

In order to use the Apple Configurator, you will need an Apple computer running OS X 10.7.2 or later.

Some iOS 6 features required that you place your iOS 6 device into supervised mode by using the Apple Configurator.

**Important:** Placing a device into supervised mode will install the selected version of iOS on the device, completely wiping the device of any previously stored user data or apps.

1. Install the Apple Configurator from iTunes.
2. Connect the iOS device to your Apple computer.
3. Start the Apple Configurator. The Configurator shows that you have a device to prepare for supervision.
4. To prepare the device for supervision:
   a. Switch the Supervision control to On. Citrix recommends that you choose this setting if you intend to maintain control of the device on an ongoing basis by reapplying a configuration regularly.
   b. Optionally, provide a name for the device.
   c. In iOS, click Latest, for the latest version of iOS you want to install.
5. When you are ready to prepare the device for supervision, click Prepare.

After you prepare the device, you can now enroll the device into Devce Manager and start deploying policies to manage the iOS device.
Managing Android Configurations

You can create a variety of policy types and configurations for your Android devices to help manage user access company data security, including App Tunnel configuration policies so your users can securely access your company intranet, TouchDown Exchange email configurations so users can seamlessly connect to corporate email accounts, app monitoring policies to block unauthorized apps that violate company policy, and a selection of Samsung SAFE device configurations.
Configuring Policies for Android Devices

You can configure various policies for Android devices in Device Manager so you can more easily manage and ensure consistency across Android device deployments. You can configure the following settings:

- Basic options
- Agent uninstallation
- Password policies (including Encryption for Android 3.0)
- WiFi configurations
- GPRS access point network configurations
- TouchDown email policies
- Security Certificates
- Configurations specific to Samsung SAFE devices
- HTC Exchange ActiveSync configurations

To manage the configuration settings for an Android device, click Configurations in the Policies > Android section in the Device Manager web console and then click New configuration to open the wizards menu. On the wizards menu, you can choose the setting you want to configure.
To configure basic options for Android devices

You can configure some of the agent parameters for Android devices in Device Manager.

1. Click XenMobile Options from the New Configurations menu.

2. In the XenMobile Options dialog box, enter a name to the configuration and optionally enter a comment.

3. Select the Hide traybar icon check box, if you want the tray icon to be visible in the traybar.

4. In Connection time-out, set the connection time-out for the device's connection to the Device Manager server, in seconds. If the device does not connect, cancel the connection attempt.

5. In Keep-alive interval, set the frequency that the device will ping the server, in order to keep the connection alive.

6. Specify the degree to which the device user will be notified of support actions initiated remotely.

7. Click Save.
To uninstall Device Manager on an Android device

1. On the New Configurations menu, click Uninstall XenMobile.
2. Enter a name for the configuration and optionally enter a comment.
3. Select the Uninstall XenMobile from devices check box and then click Save.
To create an Android credential policy

You can create an Android credential to enable integrated authentication with your PKI configuration in Device Manager, such as a PKI entity, a keystore, an a credential provider, or a server certificate. For more information on configuring PKI integration with Device Manager, see About XenMobile PKI.

You can configure an security certificate policy to use for WiFi configurations, TouchDown email configurations, Samsung Exchange ActiveSync configurations, Samsung VPN configurations, and Android SharePoint configurations.

1. In the Device Manager web console, click the Policies tab, click to expand Android and then click Configurations.

2. In the Add a credential dialog box, enter the following information:
   
   a. Credential name. Provide a unique name for the credential.
   
   b. Description. Optionally, you can type a description for the credential.
   
   c. Credential Type. Select a credential type according to the PKI configuration you have set up for Device Manager, such as a certificate, a keystore, a server certificate, or a credential provider.
   
   d. Credential file path, Server certificate, or Credential provider. Select the path or the name of the credential you are adding to the policy. If you are using a Keystore file, then you need to provider the keystore password.

3. Click Add. Now you can access this credential in Android WiFi configurations, TouchDown email configurations, Samsung Exchange ActiveSync configurations, Samsung VPN configurations, and Android SharePoint configurations.
To configure a policy to schedule connections for Android devices

1. In Device Manager, click the Policies tab and then under MDM Policies, click Android.

2. Click Configurations, click New Configuration and then click Scheduling.

3. In the Scheduling dialog box, enter a name for the configuration and optionally, a comment.

4. Under Scheduling configuration parameters, click one of the following options:
   - Do not define a connection schedule. The device does not reconnect unless the user clicks Connection in Device Manager.
   - Keep the connection permanently live. If the connection is permanent, Device Manager on the mobile device attempts to reconnect automatically to the server running Device Manager after a network connection loss. In addition, Device Manager monitors the connection by transmitting control packets at regular intervals.
     
     **Note:** This configuration consumes more battery charge and generates more network traffic. However, the setting ensures that all commands are executed in real time and completed immediately when they are sent to the device. For more information, see How Scheduling Policies Affect Android Battery Usage.

   - Force a connection every. When you click this option, you also enter a value in minutes. Device Manager on the device forces a connection to the server running Device Manager at every interval you configure.

   - Define a permanent and/or occasional connection schedule within a given time range. Device Manager keeps the connection live during the time range that you configure by selecting one or both of the following check boxes:
     
     - Keep connection alive during this time period. Device Manager on the device attempts to reconnect to the server running Device Manager after a network connection loss and monitors the connection by transmitting control packets at regular intervals. In the time line, click to select either specific times or time ranges for each day of the week when you want the Device Manager server to connect to the device. Each time segment is in 30-minute increments according to a 24-hour clock. For example, if you want Device Manager to connect between 3 A.M. and 4 A.M. every night of the week, you click the two squares between 3 and 4 for each day of the week, Monday through Sunday.

     - Force one connection during the time range below. The connection automatically shuts down after updates have taken effect. This option forces a scheduled, one-time connection to the server, in particular to check for the availability of new deployments. To avoid a connection peak at the beginning of the selected range, the relevant devices connect randomly during the defined range. Device Manager on the device only reconnects after a network
To configure a policy to schedule connections for Android devices

**Note:** Each of the preceding options includes an option to synchronize the schedule to the local device clock time rather than to Coordinated Universal Time (UTC).

5. Click Create.
How Scheduling Policies Affect Android Battery Usage

When you create a scheduling policy in Device Manager for Android devices, the way you create the scheduling policy can affect battery usage. For example, compared to a device that does not have an XenMobile client agent running on the device, the following may occur:

- If you create a scheduling policy that is set to permanently alive, with app monitoring enabled and basic MDM policy, tests reveal an additional 4 percent battery drain per hour.

- If you create a scheduling policy that is set to permanently always alive, without app monitoring, but with basic MDM policies, tests reveal a 2.5 percent battery drain per hour.

<table>
<thead>
<tr>
<th>Policy and connection conditions</th>
<th>Effect on Android device battery over time</th>
</tr>
</thead>
<tbody>
<tr>
<td>You do not configure an app control policy, but configure the following policies: password policy, TouchDown Email policy, and scheduling policy</td>
<td>8-9 percent over 5 hours equaling approximately 1.6 percent battery drain per hour</td>
</tr>
<tr>
<td>You configure an app control policy, but do not configure a scheduling policy</td>
<td>10-12 percent over 5 hours equaling approximately 2.4 percent battery drain per hour</td>
</tr>
</tbody>
</table>
| You do not configure an app control policy, but you configure the following policies:  
  - Password policy  
  - Software inventory policy  
  - WiFi configuration policy  
  - Credentials policy  
  - SharePoint configuration  
  - TouchDown Email policy  
  - Scheduling policy of permanently alive with a default connection timer set | 10-13 percent over 5 hours equaling approximately 2.6 percent battery drain per hour |
You configure the following policies:

- App control policy (a type of blacklist policy)
- Password policy
- Software inventory policy
- WiFi configuration policy
- Credentials policy
- SharePoint configuration
- TouchDown Email policy
- Scheduling policy of permanently alive with a default connection timer

For more information, see Configuring App Monitoring for Android Apps and To configure a policy to schedule connections for Android devices.
To define password requirements and enable encryption on Android device

You define the requirements for Android device passwords and enable encryption on Android 3.0 devices on the Password Policy window in Device Manager.

1. On the New Configurations menu, click Password Policy.

2. In the Password policy configuration creation dialog box, enter name for the policy and optionally, a description.

3. To establish a password policy, click the Password policy tab.

4. Select Require a code on the device and then complete the configuration parameters.

5. To enable an encryption policy for Android 3.0 devices, click the Encryption tab.

6. Select Enable device storage encryption.

   **Note:** This option is available for Android 3.0 and later. The Android 3.0 encryption operation will prompt the user to accept the action. It also requires the device to be plugged in and the device will not be usable for up to an hour while the encryption operation takes place. This is a function of the Android 3.0 encryption capability.

7. For Samsung SAFE Devices, you have the option of setting a single password for multiples users on a device. Select the Use same password across all users check box to enable this option.

8. Click Create.
To configure WiFi settings for Android devices

You can use the WiFi configuration wizard in Device Manager to deploy Wi-Fi configurations to users. The users will not be aware of details, such as the WEP encryption key. Fill in the required fields according to your configuration and specifically the fields as follows.

1. On the Create a WiFi configuration page, in Configuration name, enter an name for the policy, and optionally enter a description.

2. In Authentication, click one of the following options:
   - Open
   - Shared
   - WPA
   - WPA-PSK
   - WPA2
   - WPA2-PSK
   - 802.1x EAP (WPA Enterprise)

   For the 802.1x EAP configuration, you can specify user identity through the Device Manager macro named ${user.username} to auto-populate in the configuration. Citrix recommends that you leave the password field blank, so the device user can enter the WiFi password from their device.

3. Click Create.
To declare a GPRS access point in an Android device

You can use the APN (Assess Point Name) configuration creation wizard to declare a specific General Packet Radio Service (GPRS) access point for Android devices, like an enterprise private APN.

1. On the General tab, enter a name and optionally a description.

2. Specify the APN resource, account credentials and type of authentication.

3. Optionally, specify proxy settings and then click Create.
Configuring App Monitoring for Android Apps

Android app monitoring in Device Manager provides a secure application-browsing environment on Android devices. You can define blacklisted or whitelisted applications and take action on applications, such as preventing the applications from opening or, in real time, selectively allowing applications to run.

You can define blacklisted or whitelisted applications in an XML file that you package and push to Android devices. Sample XML files are available for reference under `<installation directory>/XenMobile Device Manager/samples/appmon/`. For example, the default Android app monitoring policy XML file is located at: `<installation directory>/XenMobile Device Manager/samples/appmon/appControlPolicyConfiguration.xml`. The configuration tags that you can include in the XML file are as follows:

- `<whitelist>` and `<blacklist>`. These tags define applications to be blocked or allowed by package name. Some sample native application package names are as follows:
  - Camera. com.android.camera
  - Browser. com.android.browser
  - Email. com.android.email or com.htc.android.mail

- `<appblockmessage>`. This tag allows customized message to appear as part of the block screen to a user and when a blacklisted or non-whitelisted application opens.

- `<appcontrolpolicylogo>`. This tag allows you to add a custom image to your app block display message when a user is prevented from installing an app. When this element is set to true, the custom logo appears. You must name the custom image appControlPolicyLogo.png and upload the file to Device Manager and then deploy the image file to the device on which you want to display the image.

- `<enforceblacklist>` and `<enforcewhitelist>`. These tags enforce applications through `<blacklist>` or `<whitelist>` tags. In case both these tags are set to true, applications defined in a whitelist XML file take precedence, and the blacklisted applications are ignored.

- `<prevent_uninstall>`. This tag allows you to block a user from uninstalling the Citrix Mobile Connect app from their device. When set to true, a user cannot uninstall the app from their device.

  **Note**: If you set this option to true, you will not be able to uninstall any other apps from the device.

- `<password>`. This tag allows a device to access blacklisted or non-whitelisted applications by using an administrator-defined passcode. There are no restrictions on the length or type of characters in the passcode. You can choose to not include this tag as part of the XML file. As a result, the user cannot enter the passcode in a text box. Instead, block screen appears with a custom company logo file (optional), customized text that you define by using the `<appblockmessage>` tag, and a button that users tap to
· `<dorestart>`. This tag defines if the application control service should be running or not running on the device. If set to false, the service does not run on the device.

**Multiple Configuration Files**

You can define multiple Android app monitoring policy files. For example, you can create a blacklist or a whitelist policy for different groups in your organization, such as a policy for your engineering group, a separate policy for your finance group, sales group, and so on. In order to create multiple app list configuration files, you need to retain the string `appControlPolicyConfiguration` in the file name. You can, however, modify the other part of the file name to help indicate the purpose of the file. For example:

· `appControlPolicyConfigurationOff.xml`. An app monitoring policy in which certain apps cannot run on the device, such as the camera.

· `appControlPolicyConfigurationDisable.xml`. An app monitoring policy in which certain apps are blacklisted and cannot be installed on the phone.

· `appControlPolicyConfigurationEnable.xml`. An app monitoring policy in which certain apps are whitelisted and can run on the device.
Example XML Syntax for Blacklisting and Whitelisting Policies

The following code samples illustrate how to use App Monitoring to create application whitelists and blacklists for your Android devices. Blacklisting app use case. Block an native email app on Android devices that are running operating systems Version 3.0 and earlier.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<appcontrol>
  <appcatalog>
    <whitelist>
      <name>org.mozilla.firefox</name>
    </whitelist>
    <blacklist>
      <name>com.android.email</name>
    </blacklist>
  </appcatalog>
  <enforceblacklist>true</enforceblacklist>
  <enforcewhitelist>false</enforcewhitelist>
  <dorestart>true</dorestart>
  <password>P@ssw0rd</password>
</appcontrol>
```

Whitelisting app use case. Only allow a XenMobile app to run on the Android device, and block all other applications.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<appcontrol>
  <appcatalog>
    <whitelist>
      <name>com.citrix</name>
      <name>com.android.launcher</name>
      <name>com.android.launcher2</name>
      <name>com.htc.launcher</name>
      <name>com.android.email</name>
    </whitelist>
    <blacklist>
      <name>com.android.email</name>
    </blacklist>
  </appcatalog>
  <enforceblacklist>true</enforceblacklist>
  <enforcewhitelist>false</enforcewhitelist>
  <dorestart>true</dorestart>
  <password>P@ssw0rd</password>
</appcontrol>
```
To add a logo to a customized block screen on an Android device

In Device Manager, you can customize the block screen that appears on an Android device by using the `<appblockmessage>` XML tag defined in an App Monitoring policy. The screen can also include a company logo.

1. Save the logo file as `appControlPolicyLogo.png` on your computer.

2. In the Import a file to the Device Manager database dialog box, import the logo file and then save the file to a destination folder on the device.

   **Note:** Make sure you use the following format to name the destination folder: `%XenMobile folder%\files`.

3. Add the following line to your `appControlPolicyConfiguration.xml` (<installation directory>/XenMobile Device Manager/samples/appmon/appControlPolicyConfiguration.xml) file after the end of `<appblockmessage>` tags: `<appcontrolpolicylogo>true</appcontrolpolicylogo>`

4. Create a deployment package that includes the application monitoring policy XML file, as well as an optional company logo file.
Common Issues with the App Monitoring Policy Implementation

With the App Monitoring feature, you might encounter the following issues:

If you notice that XenMobile is not blacklisting an application you have defined as forbidden, you can try the following tasks to remedy the situation:

- Check the XML file name; it should be appControlPolicyConfiguration.xml.
- Make sure the package containing appControlPolicyConfiguration.xml policy is deployed to the device, and the device is connected to the server.
- Check the package name for the blacklisted application. Use XenMobile Remote Support to verify native application package names under “Task Manager”.
- Validate your appControlPolicyConfiguration.xml file XML syntax with a validator, such as XML Validation.

If you can verify the preceding information, but the issue persists, open a support case and attach the XML file as well as device logs. You can share device logs by using alogcat, a free Android marketplace application.

If you notice that your Company Logo is not included as part of the block screen, verify that logo PNG file is saved as appControlPolicyLogo.png and is saved under %XenMobile folder%\files.

If you need to reset an application passcode, modify the <password> XML tag value to include the new passcode.

If you are not sure if the App Mon service is running, please note that the service is not running by default. You must push the XML policy file (appControlPolicyConfiguration.xml) to the device.

If you need to revoke device access to blacklisted applications, you can modify the <password> XML tag value to include the new passcode. The user needs to obtain and enter the new passcode.
Configuring Touchdown for Android Devices

Device Manager leverages NitroDesk’s TouchDown technology to enable you to push Exchange email configurations and security policies on Android devices using the ActiveSync protocol. It provides device administrator the ability to install TouchDown software on Android devices, configure device email settings, and apply corporate security policies to Android devices managed by Device Manager. Before you configure policies, do the following prerequisites:

- **Download NitroDesk TouchDown binary from the following locations:**
  - http://nitrodesk.com/tddownloads/nitroid-droid.apk (Smartphones with Android 2.x or 4.x)

You can visit the Android Marketplace and download the TouchDown app, or, you can download the software from your server running Device Manager. To download the TouchDown software to your Android device, make sure that you have either an internal or external SD card. Also, make sure that you enable the following setting on your Android device before attempting the download: Settings → Applications → Unknown Sources

**Obtain a license.** Contact XenMobile and provide them with your ActiveSync URL to get an encrypted license. For example, the ActiveSync URL for XenMobile is webmail.xenmobile.com. NitroDesk will provide the following information for licensing:

- Server: webmail.zenprise.com
- LICENSE KEY: ABCDEF
- SERVER KEY (for ECE Licensing): BC89OjU5MTR2

Once you have obtained your TouchDown license, you will need to add that license key to any Android TouchDown policies you create in Device Manager.

If you want to test TouchDown before purchasing a license, when you install the Touchdown application on the device, a one-month free trial license is included. With this trial license you can still send down the Touchdown policies and settings from the server running XenMobile. The only difference between the paid and free license is that when using the free trial license the mail sync is manual, so the user must initiate the sync to receive email on the device. When using the full paid license, all mail will automatically sync to the device.
Policy Combinations for Touchdown on Android Devices

The following policies combinations are common and useful ways to manage your Android devices with TouchDown.

License TouchDown App and Add Encryption Policy

License Key

RequireDeviceEncryption = true
RequireStorageCardEncryption = true

Require Passcode for TD App

DevicePasswordEnabled = true
MaxInactivityTimeDeviceLock =
MinDevicePasswordLength =
SuppressApplicationPIN = true

Prevent Attachments Download to SD Card

AllowStorageCard = false

Roaming and Custom Signature

RequireManualSyncWhenRoaming = true
SetSignature = Zenprise Protected Tablet

Update Device Type

DeviceTypeString = TouchDown
XenMobile-Certified TouchDown Policies

The following is a list of XenMobile-certified TouchDown policies that you can use with your Android device. Device Manager provides several other policies that are available but not officially certified.

Email Data Encryption Policies

The following two TouchDown policies are required to enable secure email data encryption:

- **RequireStorageCardEncryption** = true. If True, email attachments downloaded to an SD card will be encrypted. Also, the policy disallows moving a TouchDown profile/database to the SD card. Note that attachments prior to this policy will continue to remain in plain text, and all attachments after this policy is activated will be encrypted on the SD card.

- **RequireDeviceEncryption** = true. If True encrypts Contacts, Calendar and Email content; i.e., header as well as body, but not attachments.

TouchDown License Policy

**LicenseKey** = <String>. String value that specifies license key for the TouchDown application.

Individual Security Policies

- **SuppressApplicationPIN** = true. If True, the application will not show a PIN prompt and if you do not want the Exchange ActiveSync PIN to be enforced by TouchDown. This is useful if Device Manager decides to enforce a device level PIN. If set to False, then TouchDown will prompt for pin/passcode only once. To change that behavior, set this policy to false and add the policy named **MaxInactivityTimeDeviceLock**, which prompts the user for a pin/passcode after a period of inactivity.

- **MaxInactivityTimeDeviceLock**. Integer value (in seconds) that defines maximum inactivity time period before device auto locks.

- **DevicePasswordEnabled** = true. If this field is not present, TouchDown will honor the PIN policies that Exchange ActiveSync sends. If this field is present then True = Enable PIN prompting. False = TouchDown will not prompt for PIN (even if EAS policies require it). Please make sure to add the policy named **MinDevicePasswordLength** along with this policy.

- **MinDevicePasswordLength** = 1,2,3…14. Integer value that defines minimum password length for device passwords. Please make sure to add policy:DevicePasswordEnabled along with this policy.

- **AlphaNumericDevicePasswordRequired** = true|false. True, if you want TouchDown application to enforce alphanumeric codes for device passwords. Make sure to add policies:DevicePasswordEnabled and MinDevicePasswordLength along with this policy.
**XenMobile-Certified TouchDown Policies**

- *AllowSimpleDevicePassword* = true|false. If True, allows simple device passwords. Make sure to add policies:DevicePasswordEnabled and MinDevicePasswordLength along with this policy.

- *AllowStorageCard* = true|false. If false, prevents downloading of attachments to the SD card. Also, disallows moving of TouchDown profile/database to the SD card.

- *AttachmentsEnabled* = true|false. True, ability to send/receive email attachments via TouchDown

- *RequireManualSyncWhenRoaming* = true|false (to reduce data roaming charges). If True, ability to manually sync email when device is roaming thereby limiting data roaming charges.

- *DisableCopyToPhoneBook* = true (for data loss prevention purposes). If True, this will cause TouchDown to never copy contacts to the device phone book.
To configure a TouchDown policy to install and configure Exchange email accounts

You can use TouchDown in Device Manager to install and configure Exchange email accounts for users for your Android device users.

**Note:** For each TouchDown policy that you create, be sure to add the TouchDown license key to the policy, or the policy will not work.

1. **Click the Policies tab and then under Android, click Configurations.**

2. **Click New Configuration and then click TouchDown Email.**

3. **In the Add a TouchDown Email configuration dialog box, enter a configuration name for the policy (such as TouchDown Email) and then enter your Exchange email parameters.**

   **Important:** While deploying this policy, XenMobile behaves as though the NitroDesk TouchDown application is already being installed from Android Marketplace. You can leave the password field blank, which will prompt the user to enter a password. Or, you can enter the variable `%EWPASSWORD%` which will auto-configure email on the Android device without prompting the device user for the password when they launch their email client.

4. **Add the license key to the policy, so you can be sure you are deploying valid software.**

   **Click the Policies and Applications Settings tab, click New Configuration and then click Policy.**

5. **In the New Configuration dialog box, in Name, click LicenseKey and then, in the Value, enter the TouchDown license string.**

6. **Click Create and then click Add.**

After you deploy the policy, the user needs to log in the Android agent and authenticate the users’ credentials in order activate the policy.
To create a deployment package for the TouchDown policy

In order to push the TouchDown email configuration policy to your Android devices, you need to create, configure, and run a deployment package in Device Manager to the devices you want to use the policy.

1. Click the Deployment tab, click New Package and then click New Android Package.

2. In the Create New Package wizard, define and deploy the TouchDown email package. On the Name page, enter a name for the email policy, such as TouchDown Email.

3. On the Groups page, select a group or groups to be the recipient of this package. Or, you can choose to create an anonymous deployment. Any users unable to authenticate themselves to the server can be connected in anonymous mode and still receive packages.

4. On the Resources to be deployed page, in the Policies list, select the TouchDown Email policy you want to deploy and then click the right arrow to add the policy to the package.

5. In Installation Files, select the appropriate Android and TouchDown software to add to the package, depending on if you are deploying to an Android tablet or a phone.

6. In the Deployment schedule page, choose a time to run the deployment, or click Now to run the deployment immediately.

7. On the Deployment rules page,

8. When you have configured the deployment package and are ready to deploy, click Finish.

After you deploy the policy, the user needs to log in on the Android agent to authenticate the user’s credential in order activate the policy.
To initiate a selective wipe of email data by using a TouchDown API

You can initiate a selective wipe of email data (emails and attachments) on an Android device by using a TouchDown application programming interface (API). You can initiate an elective wipe on the Security tab. Status updates are available on a per-device basis on the General tab.

If a device user saved an email attachment to a location outside of the TouchDown default attachments folder, TouchDown won’t be able to detect the action. XenMobile won’t, therefore, delete the data as part of the selective wipe. This is a limitation with the use of the TouchDown API.
Configuring Deployment Rules for an Android Device Size

You can use Device Manager deployment rules to differentiate between a smartphone and a tablet based on the size of the Android device, and then deploy the policies based on size of the target device. The screen size rules enable you to apply specific policies based on whether or not the device is a tablet or a smartphone. Because some deployment resources are tablet-specific, using the screen size property will ensure accurate deployments of tablet- or phone-specific policies.

You can create the rules, for example, if you want to deploy a TouchDown Android policy on all Android tablets except the Amazon Kindle, and you want to ensure that these policies do not get deployed to any smartphones that may happen to be running the same version of Android that the tablets are running. Conversely, you may want to deploy a similar Android package, but for smartphones.

You set the rules in the Edit package wizard, in Deployment rules, on the Simple or Advanced tabs.
List of TouchDown Policies for Android Devices

AllowHTMLEmail
Type: Boolean
If True, TouchDown will allow the device to receive email that uses HTML format.

AllowSimpleDevicePassword
Type: Boolean
If True, allows simple device passwords.
Please be sure to add the following policies in combination with this policy:
- DevicePasswordEnabled
- MinDevicePasswordLength

AllowStorageCard
Type: Boolean
If False, prevents downloading of attachments to a device's SD card. Also, this policy disallows moving a TouchDown profile/database to an SD card.

AlphaNumericDevicePasswordRequired
Type: Boolean
If True, TouchDown will enforce alphanumeric codes for device passwords. Please be sure to add the following policies in combination with this policy:
- DevicePasswordEnabled
- MinDevicePasswordLength

AttachmentsEnabled
Type: Boolean
If True, allows you to send/receive email attachments via TouchDown.

DevicePasswordEnabled
Type: Boolean
If this field is not present, TouchDown will honor the PIN policies that EAS sends. If this field is present, and if you set to True, PIN prompting is enabled and a PIN will be required to access the device. If False, TouchDown will not prompt for a PIN, even if the
DevicePasswordExpirationDays

Type: Integer

Value that defines when a device's password is about to expire, measured in days. 0 = no expiration.

DevicePasswordHistoryCount

Type: Integer

Value that defines device password where 0 = no history.

DisableCalendarWidget

Type: Boolean

If True, the Calendar widget will not show any data.

DisableChangeSignature

Type: Boolean

If true, TouchDown disallows user from changing email signature line.

DisableCleanup

Type: Boolean

If True, the user will be prevented from being able to wipe configuration settings on the device.

DisableCopyPaste

Type: Boolean

If True, users will not be able to copy data from email or paste data into email when composing messages.

DisableCopyToPhoneBook

Type: Boolean

If True, this will prevent the user from ever being able to copy contacts to the device phone book.

DisableDatabaseBackup

Type: Boolean

If True, the user cannot backup data to an SD card.

DisableEasyPINRecovery

Type: Boolean
If True, the user cannot use PIN Reset by entering a Microsoft Exchange account password.

**DisableEmailWidget**

Type: Boolean

If True, email widget will not display any data.

**DisableExportTo3rdPartyWidgets**

Type: Boolean

If True, device cannot export data to external content provider widgets.

**DisableReconfiguration**

Type: Boolean

Reconfiguration of device is disabled except through the MDM client.

**DisableSettingsBackup**

Type: Boolean

If True, user cannot back up device settings to an SD card.

**DisableSpeechNotification**

Type: Boolean

If True, notifications will not be read out loud.

**DisableTaskWidget**

Type: Boolean

If True, task widgets will not display any data.

**DisableUniversalWidget**

Type: Boolean

If True, Universal widget will not display any data.

**HideCalendarInfoOnNotificationBar**

Type: Boolean

If True, notifications will not show calendar data indicating which appointment is scheduled.

**HideEmailInfoOnNotificationBar**

Type: Boolean
List of TouchDown Policies for Android Devices

- **HideTaskInfoOnNotificationBar**
  - Type: Boolean
  - If True, notifications will not show Task data.

- **hideWidgetDataWhenLocked**
  - Type: Boolean
  - If True, PIN lock will hide data in widgets.

- **License Key**
  - Type: String
  - String value that specifies license key that enables running the TouchDown application.
  - **Note:** Configuring the LicenseKey policy is required in order to use TouchDown Android policies in Device Manager.

- **MaxAttachmentSize**
  - Type: Integer
  - Integer value that defines maximum size of attachments.

- **MaxCalendarAgeFilter**
  - Type: Integer
  - Integer value specifying maximum range of past events to sync.
  - Valid values are as follows:
    - $0 = unlimited$, $4 = 2$ weeks, $5 = 1$ month, $6 = 3$ months, $7 = months$
  - Note that this will not impact the currently set values by the user if the current values are more restrictive than this value.

- **MaxDevicePasswordFailedAttempts**
  - Type: Integer
  - Integer value that defines maximum failed attempts to enter a correct device passcode before locking the user from accessing the device.

- **MaxEmailAgeFilter**
  - Type: Integer
  - Integer value specifying maximum range of past emails to sync.

- **MaxEmailBodyTruncationSize**
List of TouchDown Policies for Android Devices

**Type: Integer**

Integer values that determines the maximum sized of an email body before it is truncated.

**Valid values:**
- 0 - No Body is fetched
- 1-4k
- 2-5k
- 3-7k
- 4-10k
- 5-20k
- 6-50k
- 7-100k
- 8 - unlimited

Raw integral values representing the size in bytes may also be used. For example, if you set to 3000 (above 8), TouchDown will limit to the closest kilobyte unit shown above. Also note, this ONLY limits the upper limit the user chooses, and does not enforce the exact value. For example, if you set the value to 7, the user can then choose to limit to any value less than 100k.

**MaxInactivityTimeDeviceLock**

**Type: Integer**

Integer value (in seconds) that defines maximum inactivity time period before device auto locks.

**MinDevicePasswordComplexCharacters**

**Type: Integer**

Specifies the number of complex characters required in a device password.

**MinDevicePasswordLength**

**Type: Integer**

Defines minimum password length for device passwords.

Please make sure to add the DevicePasswordEnabled policy along with this policy.

**PhoneBookCopyFields**

**Type: Integer**

Comma-separated list of fields that can be copied to phone book.
The following fields can be entered in this string, delimited by commas, without any spaces:

- org
- photo
- note
- title
- location
- dept
- wphone
- wphone2
- hphone
- hphone2
- mphone
- ofax
- hfax
- assistantphone
- radiophone
- carphone
- pager
- compphone
- email1
- email2
- email3
- homeaddress
- workaddress
- otheraddress

**RequireDeviceEncryption**

Type: Boolean

If True, encrypts Contacts, Calendar and Email content, such as header as well as body, but not attachments.
List of TouchDown Policies for Android Devices

**RequireStorageCardEncryption**

Type: Boolean

If True, email attachments downloaded to the SD card will be encrypted. Also, True disallows moving of TouchDown profile/database to the SD card.

Please note that attachments prior to this policy will continue to remain in plain text, and after this policy is activated all attachments will be encrypted on the SD card.

**SetPlainTextSignature**

Type: String

String values that specify the signature on the application to be used with plain text email.

**SetSignature**

Type: String

String value that sets the signature on the application.

**SetSuppressions**

Type: String

String value that specifies a list of suppression codes to apply to TouchDown. To prevent TouchDown from displaying certain options to the end user. The list of codes should be comma separated, with at least one comma in the string.

**SuppressApplicationPIN**

Type: Boolean

Set to True if you do not want the application to show a PIN prompt, and you do not want the Exchange ActiveSync (EAS) PIN to be enforced by TouchDown. This is useful if the MDM decides to enforce a device level PIN. If False, TouchDown will prompt for pin/passcode only once.

To change that behavior, set this policy to False and add the policy named MaxInactivityTimeDeviceLock, which prompts the user for a pin/passcode after a period of inactivity.
List of TouchDown Application Settings for Android Devices

**AlwaysBCCSelf**

Type: Boolean

If True, sends a copy (BCC) of all outgoing emails to the configured email address.

**AppointmentRemindersAtNonPeakTime**

Type: Boolean

If True, reminds user of all appointments even if the appointment occurs during off hours or if the reminder is set to occur during off hours.

**CalendarAllDayInWidget**

Type: Boolean

If True, this option will show all-day events in the TouchDown Calendar Widget.

**CalendarCustomWeekView**

Type: Boolean

This option gives two additional options:

- Week starts on
- Week ends on

Using these options, the user can change the Week starts on and Week ends on options to select the start and end dates for the week.

Selecting a custom week start and end days will change the way the week view is shown. It will not affect the month view unless your Week start day is before the weekend day (Monday to Saturday).

**CalendarDefaultPrivacy**

Type: String

Automatically places the same privacy status for each new event unless otherwise specified by the user.

**CalendarDefaultReminder**

Type: Integer
Automatically places the same reminder length for each new event unless otherwise specified by the user.

**CalendarDefaultStatus**

Type: String

Automatically places the same availability status for each new event unless otherwise specified by the user.

**CalendarEnableResources**

Type: Boolean

If True, gives the ability to specify a resource field when creating new meetings. The user may use the resources field to specify non-attendees such as conference room resources or equipment which are available using an email address.

**CalendarFirstWeekday**

Type: Integer

 Specifies the first day of the week to show in the calendar.

**CalendarLastWeekday**

Type: Integer

 Specifies the last day of the week to show in the calendar, where 1 - 7 represents Sunday - Saturday. For example, 1 = Monday, 2 = Tuesday, and so on.

**CalendarLightTheme**

Type: Boolean

If True, the day and week Views will be shown with a light theme.

**CalendarOverdueTasksInAgenda**

Type: Boolean

If True, shows overdue tasks in the agenda.

**CalendarShowUpcomingOnly**

Type: Boolean

If True, in the TouchDown Agenda view only current appointments that have not already passed for the current day are shown.

**CalendarSyncHistory**

Defines date range of appointments to synchronize.

Values:
CalendarTasksInAgenda

Type: Boolean

If True, shows the calendar tasks in the agenda.

CalendarWorkEnd

Type: String

Species the end of the work day.

CalendarWorkStart

Type: String

Specifies the start of the work day.

CalnedarZoom

Type: Integer

Indicates zoom size for showing the day and week views in larger size and fonts. A good recommended zoom size for high resolution devices is 150%.

CleanSDCardonRemoteWipe

Type: Boolean

Removes data from SD card when a remote wipe command is issued.

- If True, will delete the entire SD card on remote wipe.
- If False, remote wipe will delete only the TouchDown folder.

CopyToPhoneNameFormat

 Defines how to copy TouchDown Exchange contacts to the phone book as First Last name or as Last First name. Values:

- 0 = First Middle Initial Last
- 1 = Last First Middle Initial
- 2 = File As

DeferServerUpdates
List of TouchDown Application Settings for Android Devices

Type: Boolean

Selected changes are deferred and batched to the server. This is selected by default and improves response time of the application as well as reduce the number of server updates.

DeviceTypeString

Type: String

Default is Android. Once this value is set, it should not be changed.

DisableSmartreplies

Type: Boolean

If True, Smart Replies are turned off. This option should only be selected if the server does not allow SmartReplies and SmartForwards. If forwards and replies are not working, then turn this option ON to determine if it works.

DisableTabletMode

Type: Boolean

If True, disables tablet mode even if it has detected that the user is working on a tablet. This option is specifically for tablet users who prefer the classic TouchDown view.

EmailAfterDeleteGoto

This option lets the user select the behavior when viewing a message and selecting to delete the message. Options include:

- Email List. Go to the email list.
- Next Email. Open the next email in the list. If none, go back to the email list view
- Previous Email. Go to the previous email in the list. If none, go back to the email list view.

EmailAlwaysExpandFolders

Type: Boolean

If True, then when the user opens Choose Folders or taps the email folder bar to change folders, the folder tree will always appear uncollapsed until the user manually collapses them.

EmailBodyStyle

Type: String

Specifies different fonts, sizes, colors and styles to be used when composing new messages in HTML mode.

EmailConfirmDeletes

Type: Boolean
If True, prompts user with a message each time the user deletes an email to confirm that the email should be deleted.

**EmailDownloadSize**

Defines the download size of the email messages from the server during synchronization. Zimbra users should set this to a value less than or equal to 10 KB.

- 1 = 4KB
- 2 = 5KB
- 3 = 7KB
- 4 = 10KB
- 5 = 20 KB
- 6 = 50 KB
- 7 = 100 KB
- 8 = Full
- 10 = No body

**EmailFetchEmbeddedImages**

Type: Boolean

If True, if using ActiveSync connection mode and HTML emails are enabled, embedded images within emails will automatically be downloaded and displayed. Note that this may cause some refreshing of the email message after each image is fetched and shown.

**EmailHighlightSender**

Type: Boolean

If True, makes the name of email sender of any email item larger and bold (as opposed to the subject).

**EmailHighlightUnread**

Type: Boolean

If True, any read items in the email list will appear grey, without subject or sender in bold, leaving only unread emails fully lit and bold.

**EmailMoveToAny**

Type: Boolean

If True, when the user selects to move email messages to other folders, the user is able to move messages to folders that have not been selected for synchronization. If this is False, then the user can only move emails to folders that have already synchronized.

**EmailMultiSelectors**
List of TouchDown Application Settings for Android Devices

Type: Boolean

If True, each email message in the email list view will show a circle on the right side. The user can place a check mark against each message by tapping the circle. Once selected the user could perform operations like Delete, Mark as Read, Mark as Unread and Move to Folder on all the selected items at once by tapping the Menu button on the device and selecting the option from the menu that opens.

EmailPreviewAttachments

Type: Boolean

If True, view a sample thumbnail of email attachments after download but before attachments are opened with an attachment viewer.

EmailSearchAsYouType

Type: Boolean

If True, when the user searches for messages using the Menu/ Search option in the email list view, the messages are filtered according to the search string as typed. If this is False, the user must tap the green arrow next to the search string to perform the search.

EmailShowSummary

Type: Boolean

If True, displays an email summary.

EmailSyncHistory

Type: Integer

Defines a date range of emails to synchronize. Default is 14 days.

EmailTextViewSize

Select the text size to use when viewing email messages. This can be set to 1 of 5 levels: smallest, smaller, normal, larger or largest.

EmailToolBarMode

Select how to display the toolbar. Values:

- 0 = Always show
- 1 = Hide
- 2 = Toggle on shake

EnableHTMLEmail

Type: Boolean

If True, TouchDown will attempt to download and display emails in HTML format. If False, emails will be retrieved as plain text.
**ExcludeAttachmentsFromGallery**

Type: Boolean

If True, ensures that media files are not scanned by the Android Gallery application when it scans the SD card for media files.

**FilteredTasksOnHomeScreenAndWidgets**

Type: Boolean

If True, displays tasks on the home screen window and on the task widget when they are viewed on the TouchDown Tasks Screen.

**HonorBackgroundDataSetting**

Type: Boolean

If True, honors the user’s preference in the Android operating system if user has decided to turn off Background Data in device settings under the Accounts & Sync heading.

**IncludePhoneContactsInPickList**

Type: Boolean

If True, lists contacts from the Android Phone Book as contact options for new email or SMS items.

**ManualSyncWhenRoaming**

Type: Boolean

If True, supresses push and polling when on a roaming network.

**NoDeleteOnServer**

Type: Boolean

If True, deleting emails on the device will not remove them from the server.

**NoMarkReadOnServer**

Type: Boolean

If True, reading emails or marking them as read/unread on the device will not mark them as read/unread on the server.

**NormalizePhoneNumber**

Type: Boolean

If True, changes contact phone numbers as follows:

- X and x, and extension will be replaced by a ; (semicolon)
NotifyAppointments
Type: Boolean
If True, shows a notification for reminders.

NotifyFailedPolling
Type: Boolean
If True, sends a notification when a periodic data refresh fails.

NotifyNewEmail
Type: Boolean
If True, sends a notification when new messages are received.

NotifyPasswordFailure
Type: Boolean
If True, sends a notification when an entered password is incorrect.

NotifySuccessfulPolling
Type: Boolean
If True, sends a notification when a successful data refresh is received.

OffPeakPollInterval
Type: Integer
Defines off-peak polling interval. Any integer >=0, which specifies the polling minutes if polling is enabled during off peak hours.

PollAtOffPeak
Type: Integer
If True, TouchDown will periodically poll for changes even during off peak times.

PollingFrequency
Type: Integer
Defines the frequency to check for changes from the server. An ideal value is 15 minutes. Keep in mind that smaller polling intervals can increase battery drain. (Note: This only applies if Push is not enabled.)

PushEnabled
List of TouchDown Application Settings for Android Devices

Type: Boolean
If True, push email is enabled.

ReminderRepeat
Type: Integer
Allows you to set interval of reminder repeats. Values:
- 0 = No repeat reminders
- X>0 = repeat after X minutes
- X<0 = Repeat X minutes before appointment

ShowEmailsOnStartup
Type: Boolean
If True, TouchDown will always open and display your email list.

Suppressions
Type: Integer
Comma-separated codes which will specify which fields to suppress.

UpdateContactChangesToPhone
Type: Boolean
If True, updates contact information on the device when detected on the server.
To update a new version of a custom Android app

Before you can update a custom app to a new version, the app must meet the following requirements:

- The new app package name must be the same as the previous version.
- The app version number must be later than the previous version.

When a new version of a custom Android app (not available on Google Play) is available, you can update the app by adding the new .ipa file to the Files tab. The next time the device connects to the server running Device Manager, the app will be updated to the new version.

**Note:** The app file name can be the same or different and doesn't affect the new version update.

1. On the Files tab, select the iOS app you want to update and then click Edit.

2. To upload an iOS application with a .apk extension, click Choose File and then browse for the app.

3. Click Update.
Configuring Exchange ActiveSync Policies for HTC devices

Device Manager supports Exchange ActiveSync policy configurations for Android HTC devices. Device Manager supports HTC API version 0.5.0.

To access the HTC Exchange ActiveSync policy, in the Device Manager web console, select Policies tab. Under Android, click Configurations and then click New Configuration > HTC Exchange ActiveSync Configuration.

HTC Exchange ActiveSync Control Configuration

The HTC Exchange Active Sync configuration allows you to remotely configure Exchange Email settings, such as server configuration and advanced mail server settings (SSL, synchronize contacts, synchronize calendar, make default email account).

**Note:** In order to push an ActiveSync policy to an HTC device, you need be running the Citrix Connect agent on an HTC device.

In the Create an HTC Exchange ActiveSync configuration dialog box, you configure settings on the following tabs:

- Configuration Name. Type a name for the Exchange ActiveSync email configuration policy so it is easily identifiable in Device Manager.

- Description. Type an optional description.

- Configuration Display Name. Type a unique name for the email account configuration as it will appear on the device.

- Server Address name. Server address of the Exchange ActiveSync server.

- User ID. Type the email account user name.

- Password. Type the email account user password.

- Domain. Type the domain for the Exchange ActiveSync server.

- Email address. Type the user's email address.

**Note:** In this field, you can use Device Manager system macros ${user.username} and ${user.mail}, which will automatically look up specific users and their email accounts based on the format listed.
Configuring Policies for Samsung SAFE Devices

Device Manager supports policy configurations for Samsung SAFE devices so that you can successfully manage your Samsung Android devices. Device Manager Samsung SAFE configurations are compatible with Samsung API Levels Version 2 and 3.

You can access all of the new Samsung configurations on the Policies tab. Under Android, click Configurations and then click New Configuration.

Restrictions Configuration

The Android Restrictions policy allows you to Allow or Disallow the following on Samsung device configurations:

- Common Apps/App Store. YouTube, Browser, Google Play Marketplace, Non-Google Play App Install
- Hardware controls. Factory reset, backup, OTA, clipboard, camera, power off, screenshot capture, SD card, and so on.
- Network settings. Bluetooth, BT tethering, WiFi, WiFi tethering, cellular data, roaming, and so on.
- USB settings. Debugging, mass storage, tethering, and so on.

In the New Samsung restriction configuration dialog box, you configure settings on the following tabs:

- General. On this tab, you enter a name and description for the configuration.
- Applications. This tab allows you to block or allow specific apps and app marketplaces. When you select an option, the app or app store will be allowed on the device. You clear the option if you do not want the device user to be able to access these apps or app stores.
- Hardware controls. This tab allows you to block or allow user control of specific hardware settings on the device. When you select an option, the device user will be able to change the hardware settings. You clear the option if you do not want the device user to be able to change these settings.
- Network. This tab allows you to block or allow user control of specific network settings on the device. When you select an option, the device user will be able to change the network settings. You clear the option if you do not want the device user to be able to change these settings.
- USB. This tab allows you to block or allow user control of specific USB controls on the device. When you select an option, the device user will be able to change the USB controls. You clear the option if you do not want the device user to be able to change these settings.

Exchange ActiveSync Control Configuration
The Samsung Exchange Active Sync control configuration allows you to remotely configure Exchange Email settings, such as server configuration and advanced mail server settings (SSL, synchronize contacts, synchronize calendar, make default email account).

**Note:** In order to push an ActiveSync policy to a SAFE device, you need a SAFE device that is running the XenMobile for Samsung agent.

In the Edit a Samsung Exchange ActiveSync configuration dialog box, you configure settings on the following tabs:

- General. On this tab, you can define your Exchange Active Sync configuration you want your Samsung devices to use.
- Advanced. On the Advanced tab, you can select or clear the following Exchange Active Sync settings:
  - Use SSL
  - Is default account
  - Synchronize contacts
  - Synchronize calendar

**Firewall Configuration**

The Samsung Firewall configuration allows you to remotely configure firewall settings for your Samsung devices.

In the Edit Samsung firewall configuration dialog box, you configure settings on the following tabs:

- General. On this tab, you enter a name and description for the configuration.
- Allow/Deny Hosts (Blacklisting/Whitelisting). The Allow/Deny Hosts tab allows you to enter IP addresses or host names you want to either allow (whitelist) or block (blacklisting) from allowing the device to access.
- Proxy Configuration. Allows you to remotely configure proxy settings for the device.
- Re-route Configuration. Allows you to configure proxy reroute configurations for your devices.

**App Uninstall Configuration**

The App uninstall configuration allows you to block or permit specific apps from being uninstalled from a Samsung device. You can choose from a prepopulated list of apps derived from all software inventories taken from all managed Samsung devices.

In the Edit Samsung App Uninstall Restriction dialog box, you configure settings on the following tabs:

- General. On this tab, you enter a name and description for the configuration.
- Application. The list of apps that you are blocking or allowing to be uninstalled from an Android device. Click New Application to add a new app to the list.

**Password Policy**
The Password Policy configuration allows you to optionally enforce a single device to use the same password for any user accessing the device plus complex flexible password parameters: numeric/alpha numeric restrictions, length requirements, expiration, wipe device after X number of failed attempts, plus encryption for device storage.

In the Password policy configuration update dialog box, you configure settings on the following tabs:

- **General.** On this tab, you enter a name and description for the configuration.

- **Password complexity.** This tab gives you great flexibility in configuring password complexity parameters for Android devices.

- **Encryption.** You can choose to enable encryption on the Android devices storage.

  **Note:** Available for Android 3.0 and later. The Android 3.0 encryption operation will prompt the user to accept the action. It also requires the device to be plugged in and the device will not be usable for up to an hour while the encryption operation takes place. This is a function of the Android 3.0 encryption capability.

- **Samsung SAFE.** This setting allows you to set a single password for multiple users on a device.

**Silent App Un-Installation Configuration**

The Silent App UnInstall configuration allows you to initiate software un-installation without requiring user intervention. You can choose from the list of apps derived from all software inventory of all managed Samsung devices in your network. When you deploy the policy, the apps selected will be uninstalled quietly and seamlessly.

**Selective Wipe for SAFE Devices**

You can use Selective Wipe on Samsung SAFE devices to remove email data, document data, and application data.
Managing Samsung Knox Configurations

Device Manager provides the ability to manage the Samsung Knox Container on Samsung Knox Devices (Samsung S4 devices that support the Knox API version 1.0) by deploying the following XenMobile MDM Policies:

- To create a Samsung Knox Exchange ActiveSync configuration
- To create a Samsung Knox password policy
- To create a Samsung Knox browser configuration
- To create a Samsung Knox enterprise VPN configuration
- To create a Samsung Knox app restriction policy
- To perform a selective wipe on a Samsung Knox device
- To create a Samsung Knox app install policy
- To create a Samsung Knox app uninstall configuration
- To create a Samsung Knox Remote Support policy
To create a Samsung Knox Exchange ActiveSync configuration

The Samsung Knox Exchange ActiveSync configuration allows you to remotely configure Exchange Email settings, such as server configuration and advanced mail server settings (SSL, synchronize contacts, synchronize calendar, make default email account).

1. In the Device Manager web console, select the Policies tab.


3. In the Create an HTC Exchange ActiveSync configuration dialog box, General tab., enter the following information:
   
a. Configuration Name. Type a name for the Exchange ActiveSync email configuration policy so it is easily identifiable in Device Manager.

b. Comment. Type an optional comment for the policy.

c. Server Address name. Server address of the Exchange ActiveSync server.

d. User ID. Type the email account user name.

e. Password. Type the email account user password.

f. Domain. Type the domain for the Exchange ActiveSync server.

g. Email address. Type the user's email address. In this field, you can use Device Manager system macros ${user.username} and ${user.mail}, which will automatically look up specific users and their email accounts based on the format listed.

h. Identify Credential. If an identity server has been configured with Device Manager, then you can select the identify certificate or credential type here.

4. Next, select the Advanced tab and select the advanced settings you want to be activated for the policy: Use SSL, Is default account, Synchronize contacts (between device client and server), Synchronize calendar.

5. Click Create.
To create a Samsung Knox password policy

The Samsung Knox password policy provides the ability to configure device passcode policy according to the standards of your IT department.

1. In the Device Manager web console, select Policies tab.

2. Under Android, click Configurations and then click New Configuration > Samsung -> Samsung Knox Password Policy.

3. In the New Samsung Knox Password Policy Configuration dialog box, General tab, enter a name and optional description for the policy.

4. On the Policies tab, select which password requirements you want to enforce. You can configure such settings as the maximum number of characters allowed, maximum allowed failed attempts (before user is locked out of their Samsung device), minimum number of complete characters (non-alphanumeric), expiration (in days) before password expires, and so on. When an option is selected, then the policy is enforced.

5. In the Forbidden Strings tab, click New Forbidden String, and then select in the field to enter a string you want to prohibit from being used in a password. For example, you may want to prevent common unsecure password strings that are easy to guess, such as 'password' or 'welcome' or '123' and so on.

6. Click Create.
To create a Samsung Knox browser configuration

The Samsung Knox browser configuration allows you to control behavior of the Knox browser on the Samsung device, such as blocking the browser from being used, enabling or disabling JavaScript, disable cookies or pops ups, disable auto-fill, and forcing the browser fraud warning.

1. In the Device Manager web console, select Policies tab.

2. Under Android, click Configurations and then click New Configuration > Samsung -> Browser Configuration (Available for Samsung Knox).

3. In the New Samsung Browser Configuration dialog box, select the security settings you want to enforce on the device browser:
   a. Disable Browser
   b. Disable Popup
   c. Disable JavaScript
   d. Disable Cookies
   e. Disable Autofill
   f. Force Fraud Warning

4. Click Create.
To create a Samsung Knox enterprise VPN configuration

The Samsung Knox enterprise VPN configuration allows you to specify corporate VPN settings so apps launched from inside the Knox secure container (such as the browser) use a secure connection. The Samsung KNOX container is an on-demand FIPS-certified VPN client called per-app VPN. Per-app VPN allows you to configure, provision, and manage the use of VPN on a per-application basis. So you can create the Samsung Knox VPN strong IPSec VPN encryption, including support for Suite B cryptography.

1. In the Device Manager web console, select Policies tab.

2. Under Android, click Configurations and then click New Configuration > Samsung -> Enterprise VPN.

3. In the New Samsung Knox Enterprise VPN configuration dialog box, General tab, enter a name and option description for the policy.

4. In the VPN tab, enter the following information:
   a. Connection name
   b. Hostname
   c. Enable backup server. (If configured on the VPN server.) If a backup server is connected, then complete the backup server configuration.
   d. Backup VPN server name
   e. Username
   f. Password
   g. Groupname
   h. IPsec group Id type
   i. IKE version
   j. Authentication Method
   k. Identity Credential
   l. CA Certificate

5. Select the Others tab and select any additional VPN parameters you want to configure for the connection.

6. Select the Forward Routes tab and click Forward Routes to enter a new forwarding routes if your corporate VPN server supports multiple route tables.
To create a Samsung Knox enterprise VPN configuration

7. Click Create.
To create a Samsung Knox app restriction policy

The Samsung Knox app restriction policy allows you to configure app blacklists of apps you want to block from being installed in the Knox Container. You can also specify on a per-app basis those apps you want to allow ers to install as we, (whitelist).

1. In the Device Manager web console, select Policies tab.

2. Under Android, click Configurations and then click New Configuration > Samsung -> Samsung Knox App Restriction.

3. In the New Samsung Knox App. Restriction Policy dialog box, General tab, enter a name and optional description for the policy.

4. Select the Applications tab, and then click New Application.

5. In the New user an an application to deny or allow dialog box, enter an application name, or from the drop down list select an app that has already been added to Device Manager.

6. Once you have a select an app, choose either Deny or Allow.

7. Click Create.

8. You can add as many apps as you wish. When you are done adding apps, click Create to create the policy.
To create a Samsung Knox app uninstall configuration

The Samsung Knox app uninstall configuration policy allows you to perform silent app removal from the Knox Container.

1. In the Device Manager web console, select Policies tab.

2. Under Android, click Configurations and then click New Configuration > General -> App Uninstall.

3. In the App Uninstall dialog box, enter a name for the configuration and optional description.

4. In the Applications to be uninstalled section, enter or select the name of the app you want to uninstall.

5. Click OK.
To create a Samsung Knox Remote Support policy

You can create a Samsung Knox remote support policy if you have purchased and install the XenMobile MDM Remote Support product. You will need to perform the following setup in order to enable remote support for Samsung Knox devices:

• Install XenMobile Remote Support application in your environment.

• Configure a Remote Support app tunnel.

• Configure a Samsung Knox remote support policy (this topic).

• Deploy both the Remote Support app tunnel and the Samsung remote support policy to a user’s device.

There are two kinds of remote support you can enable for a Samsung Knox device:

• Basic Support. This allows you to view diagnostic information about the device such as system information, processes that are running, task manager (memory and CPU usage), installed software folder contents, and so forth.

• Advanced Support. This option allows you remote control over the device’s screen, including control with colors, in either the main window, or in a separate, floating window, establishment of a Voice-over-IP session (VoIP) between the helpdesk and the user, configuration of settings, and establishment of a chat session between the helpdesk and the user.

1. In the Device Manager web console, select the Policies tab.


3. In the New Samsung Premium Remote Support configuration dialog box, enter a name for the policy.

4. Select either Basic Support or Advanced Support.

5. Click Create.
To perform a selective wipe on a Samsung Knox device

A selective wipe will remove all XenMobile policies and packages that have been deployed to the device (including the Samsung Knox container), as well as any corporate data, while retaining personal information and selected settings. The device can be re-enrolled at a future time.

**Note:** Selectively wiping an Android devices does not completely disconnect the device from Device Manager and a user's corporate network. In order to break the connection between the device and the corporate network, you also need to revoke the Android device.

1. From inside the Device Manager, select the Devices tab.
2. From the Devices tab, select the Samsung Knox device you want to selectively wipe.
3. From the Security menu, select Selective Wipe.
4. Confirm that you want to selectively wipe the device.
Managing Windows Phone 8 Configurations

You can create a variety of policy types and configurations for your Windows Phone 8 devices to help manage user and company data, including Windows Phone app distribution through the Enterprise Hub Company store, storage policies (to encrypt stored data and the ability to prevent storage card usage), password policies, Exchange ActiveSync email policies so your users can seamlessly connect to corporate email accounts, as well as your own custom policies.
To create an Enterprise Hub policy for Windows Phone 8 devices

Before create and deploy the Enterprise Hub policy for Windows Phone 8 devices, make sure you have obtained your AET (.aetx file )signing certificate from Symantec and that you have obtained and signed the Citrix Company Hub app (CitrixCompanyStore.xap) using the Microsoft app signing tool (XapSignTool.exe). For more information, see Configuring Apps for Windows Phone 8 Worx Home.

Note: Before you deploy the Enterprise Hub policy to devices, ensure that your users have been enrolled into Device Manager, or the policy will not work.

1. In the Device Manager web console, select the Policies tab.

2. Under MDM Policies on the left, select Windows Phone 8 > Configurations > New Configuration > Enterprise Hub.

3. In the Create an AET cert and/or enterprise app store configuration dialog box, enter a name for the policy.

4. Next, click on the Choose file buttons to upload the AET file and the signed Citrix CitrixCompanyStore.xap to the policy.

5. Click Create. To deploy the package to your Windows Phone 8 devices, you need to add this policy to a deployment package and deploy it a specified user group. For more information on creating and deploying policies in a deployment package, see To create and deploy a deployment package.
To access apps on the Windows Phone 8 Work Home App Store

1. When apps have been deployed through XenMobile to a user’s Windows Phone 8 device, the apps can be access through the Windows Phone 8 Worx Home app store. To access the app store, on your Windows Start Screen, tap the Worx Home tile.

2. In the Worx Home login screen, enter your Device Manager (or for XenMobile user name and password (the same one uses when you enrolled), and then tap the arrow button. (Authentication is only required once.)

3. In the Available Apps screen, you can scroll to the app you want to install. To install an app, tap it and install it.
To create a Windows Phone 8 app deployment package

In order for your Windows Phone 8 users to receive apps through the Company Store, you need to add those apps to Device Manager and deploy them to your users. In addition, you need to make sure you deploy the Enterprise Hub policy as well. You can deploy both the apps and the Enterprise Hub policy using a Device Manager deployment package.

1. Select the Deployment tab, click New Package and then click New Windows Phone 8 Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the package, such as Android App Store, and then click Next.

3. On the Groups of users window, select the group you created earlier and then click Next.

4. On the Resources to be deployed window, under Available Resources, scroll to the Enterprise App Store and select the Windows Phone 8 apps you want to add, click the right arrow button and then click Next. If you haven’t yet deployed the Enterprise Hub policy, you can add it here.

5. On the Deployment schedule window, select the If not deployed Start Now option and then click Next.


8. In the packages list, click Deploy.
To perform a selective wipe on a Windows Phone 8 device

When you selective wipe a Windows Phone 8 device using Device Manager, the following is removed from the device:

- The enterprise token that allows apps to be installed on the device by Device Manager.
- All Device Manager certificates.
- All Device Manager configurations that have been deployed to the device.

1. From inside the Device Manager web console, select the Devices tab.

2. To find the Windows Phone 8 device you want to selectively wipe, sort the list using the OS filter.

3. Select the check mark next to the device you want to selectively wipe.

To configure Windows Phone 8 Exchange ActiveSync policies

You can use this policy to preconfigure and deploy your corporate Exchange ActiveSync configuration to your Windows Phone 8 device users. Note, however, that the policy does now allow you to set the user password. The device user will need to set that parameter from the device once the policy is pushed.

1. On the Policies tab, under Windows Phone 8, click Configurations.

2. In the New Configuration menu, click New Configuration > Exchange ActiveSync.

3. In the Create a new Exchange ActiveSync configuration dialog box, on the General tab, enter the following information:
   a. Configuration Name. Type a name for the policy.
   b. Description. Type an optional description.

4. In the Email parameters section, enter the following information:
   a. Account name. Name of the Exchange ActiveSync account.
   b. User name. Type the account user name.
   c. Domain. Type the domain for the Exchange ActiveSync server.
   d. Email address. Type the user’s email address. In this field, you can use Device Manager system macros ${user.username} and ${user.mail}, which will automatically look up specific users and their email accounts based on the format listed.
   e. Server name. Type the name of the Exchange ActiveSync server.

5. Click the Advanced tab and enter the following information:
   a. Synchronization Frequency. Select the frequency with which you want the email account on the device to sync with the Exchange server. This setting specifies how often any new data from the server will be sent to the device.
   b. Synchronization Items. Select which items you want to be synced, such as email, contacts, calendar, and so on.
   c. Loggin. Specify the level of detail for logging of Exchange activity (or no loggin).

6. Click Create.
To configure Windows Phone 8 password policies

1. On the Policies tab, under Windows Phone 8, click Configurations.

2. In the New Configuration menu, click new Configuration > Password Policy.

3. In the Create a password policy dialog box, on the General, enter a name for the policy and a brief description.

4. In the Policies section, configure your Windows Phone 8 password policy according to the standards of your IT department. The password policy options are as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Require a password on the device</strong></td>
<td>Enables password protection on the device. If cleared, the device does not require a password on the device (unless the device user sets it manually).</td>
</tr>
<tr>
<td><strong>Allow simple password</strong></td>
<td>Allows the use of a simple password, which is one consisting only of repeated “2222” or sequential “abcd” characters.</td>
</tr>
<tr>
<td><strong>Password complexity</strong></td>
<td></td>
</tr>
<tr>
<td>· Alphanumeric: Requires that at least one character of the password is a letter.</td>
<td></td>
</tr>
<tr>
<td>· Alphanumeric or Numeric: Requires that the password contain either at least one letter or one number (but not both).</td>
<td></td>
</tr>
<tr>
<td>· Alphanumeric, Numeric, or none: Password can contain both alphanumeric and numeric characters.</td>
<td></td>
</tr>
<tr>
<td><strong>Minimum password length</strong></td>
<td>Allows you to set the minimum overall length (in characters) required for the password.</td>
</tr>
<tr>
<td><strong>Minimum password complex characters</strong></td>
<td>Allows you to the number of characters that are required to be present in the password. The characters are defined as: lower case alphabetical characters, upper case alphabetical characters, numbers, non-alphanumeric characters. For example, if the value is 2, a password with both upper case and lower case alphabetical characters would be sufficient, as would a password with lower case alphabetical characters and numbers.</td>
</tr>
<tr>
<td><strong>Password expiration (in days)</strong></td>
<td>Allows you to specify the number of days for which the password can remain unchanged. After the set number of days, the user is forced to change the password before the device is unlocked.</td>
</tr>
<tr>
<td><strong>Password history</strong></td>
<td>Allows you to specify the number of previously used passwords to store. When a user creates a new password, the user can’t reuse a stored password that was previously used.</td>
</tr>
</tbody>
</table>
### Inactivity before device lock (in minutes)

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Allows you to specify the length of time that the phone can be inactive before the password is required to reactivate it. You can specify any interval between 30 seconds and 1 hour. The default is 15 minutes. The format of the setting is hh:mm:ss; for example, 15:00 = 15 minutes.</td>
<td></td>
</tr>
</tbody>
</table>

5. Click Create. The new policy appears in the Policies list.
To create Windows Phone 8 storage policies

1. In the Device Manager web console, click the Policies tab, click to expand Windows Phone 8 and then click Configurations.

2. In the New Configuration menu, click Storage Policy.

3. In the Create a storage policy configuration dialog box, General section, type a name for the policy.

4. Next, select one or both of the policy options:
   a. Require Device Encryption. Select if you want all data stored on the device to be encrypted. Selecting this option ensures that no one will be able to access the data without the PIN code, even if the device is cracked and the chip is removed.

   b. Disable Storage Card. Select if you want to prevent a user from storing data on an external storage card.

5. Click Create.
To configure Windows Phone 8 custom policies

1. On the Policies tab, under Windows Phone 8, click Configurations.

2. In the New Configuration menu, click New Configuration > Custom Policy.

3. Enter your own custom XML configuration for Windows Phone 8, and then click Validate.
Creating Windows 8 Tablet Registry Configurations

You can create Windows 8 Tablet registry configurations to allow for a single point for device registry management. A set of registry keys can only exist in a configuration. You can create different settings and then deploy them selectively to some or all of the mobile devices under management. Once you create the registry settings, you can then deploy them to your Windows 8 devices using a Device Manager deployment package.
To create a Windows 8 tablet registry configuration

1. In the Device Manager web console, on the Policies tab, under Windows 8 Tablet, click Registry Configurations.

2. Click New > Configuration.

3. In the New registry configuration dialog box, type a name and then click Create. This will create a default, blank registry value set, for which you can create custom registry entries to suit your requirements.

4. To create a new registry key, select a registry folder and click New > Key. Type a name for the new key and then click Create.

5. To enter a registry key value, click New Value.

6. In the Create a new registry value dialog box, enter the following information:
   a. Name. Type a name for the registry key.
   b. Type. Choose the registry type, such as String, DWORD, Expandable String, or Executable.
   c. Value.

7. Click Create.
Managing Windows Mobile Configurations

You can create several types of device management policies and configurations for your Windows Mobile devices such as App Tunnels for secure connections to your corporate network at the app level, registry setting configurations, server settings policies and custom XML configuration policies.
About the Windows Mobile Server Collection

The Servers collection in Device Manager is used to both administer existing and to add new servers to your Device Manager deployment. A default server configuration is created during the Device Manager installation.

In the Server Groups collection for Windows Mobile, you can add new access points to the server and create backup servers. You can also configure server groups. You can use server groups to do the following:

- Create one or more backup servers (valid only if strong authentication is not enabled for the product).
- Define several access points for connection to the server running Device Manager.
- Provide logical grouping for multiple deployment locations.
To add a new server in the Windows Mobile server collection

To create other servers running Device Manager, you first need to create a new server group. To do so, click New group. After you create at least one group, you can then create a new server.

If a device cannot connect to the selected server, the device will attempt to connect through other servers in the same group, one after the other, following the defined order, through to the default server. To change the order in which servers are listed, you right-click a server and then click Options (or click Down).

1. Click New server.
2. Enter a display name for the server.
3. Enter the IP address or fully qualified host name (FQDN) of the server.
4. Enter the host port.
5. Choose optional settings of SSL, use a proxy server, or use as the default server.
To edit or delete a server

1. Click the server whose settings you want to modify and then click Update.
2. Modify the settings and then click Update.

To delete a server

1. Click the server whose settings you want to modify and then click Delete.
2. When prompted, click to confirmation the deletion.
To configure device IP address ranges

By design, devices connect to the default server running Device Manager at the provided host name or IP address. For situations in which you have a LAN, WiFi, or USB connection, you can specify IP address ranges. If the Windows Mobile device has an IP address within this range, the device will connect to the server running Device Manager. Specifying IP address ranges is useful when uploading new software because you can lower data charges.

1. Click an existing server create a new server.
2. Click Update.
3. Click the Device IP ranges tab.
4. Click New IP address range.
5. Enter the starting and ending IP addresses and then click Create. When you select a server, you can also update the server settings, such as the IP address, default server, and IP address ranges.

You must also deploy a server group to a user group on the Deployment tab.
About the Registry Collection for Windows Mobile

The Registry collection is used to configure the Windows Mobile or Windows CE registry base of the mobile devices, thus allowing for a single point for device registry management. Device Manager includes a series of step-by-step wizards, allowing for rapid setup and deployment of registry configurations.

In addition to configuring both new and existing registry values, either for the operating system or installed third-party applications, you can manage the Device Manager client configuration options. This allows for multiple client backup settings, and control over network connectivity, and is included with a dedicated wizard.

A set of registry keys can only exist in a configuration. You can create different settings and then deploy them selectively to some or all of the mobile devices in the fleet. You can create a new registry configuration either manually or via the wizard. The configuration wizard includes pre-configured options for the following applications:

- Device Manager options
- Uninstall Device Manager client from a device
- Scheduling
- Connect on SMS reception / Connect on call
- MS Exchange configuration for MS Outlook
- Security rules

**Note:** This feature is available only with the Device Manager Secure Device option.

- Configure devices when roaming
Device Manager uses registry keys to store its own data in each mobile device. You can configure these options by using the Device Manager Options wizard.

Device Manager configuration backup. Device Manager client settings can be stored on the removable memory card in the mobile device. If a mobile device has to be hard reset, it will automatically retrieve the settings required to reconstruct the configuration, such as Device Manager agent, registry keys, Device Manager-related security certificates, and network configuration. For devices with more than one memory card, you can also configure a backup to a specific card.

Connect to these networks. Device Manager is authorized, if necessary, to activate connections as defined in Network Management on the mobile device:

- User-defined Office
- User-defined Internet
- Built-in Office (My Work Network connection)
- Built-in Internet (My ISP connection)

The operating mode depends on the status of the mobile device's current network connection at any given time:

<table>
<thead>
<tr>
<th>Authorized and Active connection</th>
<th>If the server running Device Manager is accessible through this network, the Device Manager agent connects to the server running Device Manager.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unauthorized but Active connection</td>
<td>The mode is the same as for Authorized and Active connection.</td>
</tr>
<tr>
<td>Authorized and Inactive connection</td>
<td>The Device Manager agent will activate the connection and then connect to the server running Device Manager.</td>
</tr>
<tr>
<td>Unauthorized but Active connection</td>
<td>The Device Manager agent will not attempt to activate the network connection.</td>
</tr>
</tbody>
</table>

- Device Manager icon. This option hides or displays the Device Manager icon in the mobile device's traybar.

- Connection time-out. This option sets the connection time-out for the device's connection to the server running Device Manager, in seconds. If the device does not connect, cancel the connection attempt.

- Keep-alive interval. This option sets the frequency that the device will ping the server, in order to keep the connection alive.
• Ask the user before allowing remote control. When a connection is established with the helpdesk, the remote device prompts the user to allow the helpdesk to take remote control over the mobile device through a confirmation dialog box.

• Ask the user before allowing file transfer by the remote control tool. File transfers from a device to the server can be configured for anonymous mode on the Device Manager web console, with user confirmation of the request or with only presentation of a message to inform the user.
To uninstall Device Manager from a Windows Mobile device

It is preferable to create a special group, such as UninstallGroup, on the Users tab to uninstall Device Manager and then create a package of Device Manager option registry keys containing the uninstall option. You can then deploy this package to UninstallGroup on the Deployment tab. Thereafter, you can add a user to the UninstallGroup in order to uninstall Device Manager from the remote device.

1. Select Uninstall XenMobile from devices.

2. On the Deployment tab, deploy this configuration to selected users.
Configuring a Connection to Device Manager on SMS Reception or Call

This feature allows for Windows Mobile devices to be forced to connect back to the server running Device Manager when either a call or SMS from a preconfigured number is received by the device. To enhance security, a keyword must be included within the SMS message. This is particularly useful if a device is lost or stolen, and needs to be remotely disabled or wiped.

To use this feature, in the Connect On SMS reception / Connect on Call dialog box, select to either connect to server when receiving a SMS message or phone from a specific number.
Configuring Exchange Server for Windows Mobile Devices

Using the MS Exchange configuration for MS Outlook wizard in Device Manager, you can configure mobile email settings easily and automatically across your entire fleet of devices. These settings will generate the appropriate registry keys to synchronize with an Exchange server:

- Exchange server name
- Settings for receive emails and attachments
- Calendar settings
- Other settings

On the General tab, elect the appropriate device operating system type because different configuration options are available depending on the operating system release. For instance, tasks synchronization is available for Windows Mobile 6 devices but not for Windows Mobile 2003 or Windows Mobile 5.

If you create an Exchange tunnel, the value you enter in Server address has to be the same as the value you enter in Specify a local alias on the Tunnel tab, if you specified an alias. The server running Device Manager manages and optimizes the data stream and communication between the Exchange server and the mobile device.
The roaming settings in Device Manager for Windows Mobile devices will generate the appropriate registry keys for a better control of the wireless communications costs while traveling abroad and using other mobile operator networks than your default mobile operator (for example, the name of the mobile operator stored on the SIM card of the mobile device). In roaming situations, when the device has a cellular connectivity setup, the device will connect to the server running Device Manager according the following settings:

**Note:** You can select more than one setting.

- Use on demand connection only. The device will only connect to the Device Manager Server if the end-user manually triggers the connection using the Device Manager Agent screen on his device, or if a mobile application requests a forced connection (such as a push mail request if the Exchange server has been set accordingly). Note that this option temporarily disables the default device connection schedule policy as defined in the Scheduling wizard within the Registry tab.

- Block all cellular connections except the ones managed by Device Manager except for the data traffic officially declared in a Device Manager application tunnel or other Device Manager device management tasks, no other data will be sent or received by the device. For example this option will disable all connections to the Internet via device web browser (Pocket IE).

- Block all cellular connections managed by Device Manager. All "application" data transiting through a Device Manager tunnel will be blocked (including the Device Manager Remote Support application). However the data traffic related to pure "device management" (such as the deployment of a new Device Manager package) will not be blocked.

- Block all cellular connections to the Device Manager Server. In this case, until the device is either reconnected via USB, Wi-Fi or via its default mobile operator cellular network, there will be no traffic transiting between the device and the Device Manager Server.

In the Deployment tab, you can also configure rules to avoid deploying a specific package (say “XYZ”) when roaming. In this case, if the Block all cellular connections managed by Device Manager option has been selected, all packages except “XYZ” will still be deployed even in roaming situations.

In the Tunnel tab, a given application tunnel can be forced to block all data traffic when roaming. For example, if the Block all cellular connections except the ones managed by Device Manager option has been selected, the “CRM_App” data traffic will still be blocked although it is managed as a Device Manager tunnel.
To configure a new registry manually

1. Click New configuration.

2. Enter a name. This will create a default, blank registry value set, for which you can create custom registry entries to suit your requirements.
To delete a registry key configuration

1. Select the registry configuration to be deleted.
2. Click Delete.
3. When prompted, click to confirm the deletion.
To use the schedule wizard to configure connections for Windows Mobile devices

1. On the Policies tab, click Registry Configurations, and then on the Wizard menu, click Scheduling.

2. In Scheduling configuration parameters, select the following options:

   - Do not define connection policy. The device will not reconnect unless the user clicks Connection in Device Manager.

   - Keep connection permanently live. If the connection is permanent, Device Manager on the mobile device will attempt to reconnect automatically to the server running Device Manager after a network connection loss and will monitor the connection by transmitting control packets at regular intervals (This configuration is not recommended because it consumes more battery charge and generates more network traffic.)

   - Define a permanent and/or occasional connection schedule within a given time range. Keep the connection live during the following time range:

      - Define a period in which the device will stay connected to the server. Device Manager on the device will attempt to reconnect to the server running Device Manager after a network connection loss and will monitor the connection by transmitting control packets at regular intervals.

      - Force one connection during the time range below. The connection will automatically shut down once updates have taken effect. This option forces a scheduled, one-time connection to the server, in particular to check for availability of new deployments. To avoid a connection peak at the beginning of the selected range, the relevant devices will connect randomly during the defined range. Device Manager on the device will only reconnect after a network connection loss if an operation was in progress. The server running Device Manager will likewise terminate the connection after an inactive period.

   **Note:** Both of the preceding options include an option to see the schedule to the local device clock or to UTC time.
Creating Symbian Configuration Profiles

Symbian devices configuration in Device Manager is done by sending OMA Device Management commands to the devices. The list of supported commands can be found from the Nokia Developer web site. A search for the “OMA Device Management” keywords in the document section of web site will return a number a documents describing Device Description File (DDF) for features that can be controlled using OMA-DM.

OMA DM will allow control of Symbian devices by:

- Defining Wifi or GPRS Access Points.
- Defining Mail for Exchange parameters.
- Encrypting device and/or SD cards.
- Customizing devices.
- Configuring VoIP parameters.

That list is not an exhaustive list of what can be configured on Symbian devices, and features may depends on the device model. For instance, device encryption was supported on S60 3.2 devices only on the E-Series devices.

Device Manager support <Alert>, <Add>, <Replace> and <Exec> OMA-DM commands.

The following example will display the message “Management in progress...” on the user device during 30 seconds.

<Alert>
  <CmdID>_cmdid_</CmdID>
  <Data>1100</Data>
  <Item>
    <Data>MIINDT=30</Data>
  </Item>
  <Item>
    <Data>Management in progress...</Data>
  </Item>
</Alert>

The structure of an OMA-DM command must always contain a command ID. In this case, the command ID is interpreted on the fly. It is then replaced by the placeholder "_cmdid_". The following example will configure the Mail for Exchange client.

<Replace> <CmdID>_cmdid_</CmdID> <Item>
  <Target><LocURI>./MailForExchange/Server</LocURI></Target>
  <Data>webmail.mycompany.com</Data>
</Item>
</Replace>

<Replace> <CmdID>_cmdid_</CmdID> <Item>
  <Target><LocURI>./MailForExchange/UseDefaultPort</LocURI></Target>
</Item>
</Replace>
Creating Symbian Configuration Profiles

Note that this is just an example; a lot more options are available to configure in the Mail for Exchange client. As you can see, several commands can be chained in the same command block. User attributes can be used using $user.attributename macro. Those macro will be replaced on the fly by the actual user data.

Special config can be created to enforce configuration on Symbian devices. Those configurations will have the following format.

<Exec> <CmdID>_cmdid_</CmdID> <Item>
<Target> <LocURI>am_policy</LocURI> </Target>
</Item> </Exec>

The <LocURI> parameter can be set with the following values:

- am_policy / am_policy_del: This will set or remove the application management policy enforcement to control application installation and removal.
- ap_policy / ap_policy_del: This will set or remove the Access Points policy management enforcement.
- custo_policy / custo_policy_del: This will set or remove the device customization policy management enforcement.
- ds_policy / ds_policy_del: This will set or remove the Data Synchronization policy management enforcement.
- email_policy / email_policy_del: This will set or remove the Email policy management enforcement.
- im_policy / im_policy_del: This will set or remove the Instant Messaging policy management enforcement.
- wlan_policy / wlan_policy_del: This will set or remove the WLAN Access Points policy management enforcement.
To create a new Symbian configuration profile, on the Policies tab, click New configuration in the Configurations section of Symbian.
To use the schedule wizard to configure connections for Symbian devices

1. On the Policies tab, click Configurations, and then on the New Configuration menu, click Scheduling.

2. In the Scheduling dialog box in Device Manager, enter a name for the configuration and optionally, a description.

3. In Scheduling configuration parameters, select the following options:

   - Do not define connection policy. The device will not reconnect unless the user clicks Connection in Device Manager.

   - Keep connection permanently live. If the connection is permanent, Device Manager on the mobile device will attempt to reconnect automatically to the server running Device Manager after a network connection loss and will monitor the connection by transmitting control packets at regular intervals (This configuration is not recommended because it consumes more battery charge and generates more network traffic.)

   - Define a permanent and/or occasional connection schedule within a given time range. Keep the connection live during the following time range:

     - Define a period in which the device will stay connected to the server. Device Manager on the device will attempt to reconnect to the server running Device Manager after a network connection loss and will monitor the connection by transmitting control packets at regular intervals.

   - Force one connection during the time range below. The connection will automatically shut down once updates have taken effect. This option forces a scheduled, one-time connection to the server, in particular to check for availability of new deployments. To avoid a connection peak at the beginning of the selected range, the relevant devices will connect randomly during the defined range. Device Manager on the device will only reconnect after a network connection loss if an operation was in progress. The server running Device Manager will likewise terminate the connection after an inactive period.

   **Note:** Both of the preceding options include an option to see the schedule to the local device clock or to UTC time.
Creating Deployment Packages

You can remotely deploy a package of settings to a mobile device from the Deployment tab in the web console. You can use the Package building wizard to build out packages by using preconfigured objects. Connected devices receive the package as soon as scheduling rules are met. Reconnecting devices receive the package as they reconnect subject to other rule criteria.

Packages are compilations of previously created resources, prepared into configurations for the various user groups. Packages include the following:

- A package name
- Groups of users
- Resources, which, depending on the operating device, are a combination of the following:
  - A server group
  - App tunnels
  - Registry configurations
  - XML configurations
  - Software inventory
  - Applications
  - Files
  - Deployment schedule
  - Deployment rules
Device Manager contains pre-configured base deployment packages that automatically deploy to devices as soon as a user enrolls the device in Device Manager. The base packages are important for enabling basic device management.

The base packages in Device Manager contain the following policy configurations, categorized by device platform:

- **iOS.** Software inventory and MyAppStore (Citrix Worx Home web clip) policies, plus the following Citrix apps:
  - Citrix ShareFile
  - Citrix Receiver
  - Citrix Podio
  - Citrix GoToMeeting
  - Citrix Mobile Connect (for preexisting customers upgrading to Citrix Worx Home)

- **Android.** Scheduling policies for connections to XenMobile, remote support tunnel, and software inventory policies, the Worx Home web clip, plus the following Citrix apps:
  - Citrix Receiver
  - Citrix GoToMeeting

- **Windows Phone 8.** Passcode policy.

- **Windows 8 Tablet.** Software inventory policies.

- **Symbian.** Passcode policy.

- **Windows Mobile.** Remote Support tunnel, scheduling, passcode, client config policies.

For more information about configuring policies, see Creating Policies. For more information about deployment packages, see Creating Deployment Packages.
To create a software inventory package

A software inventory policy in Device Manager enables you to check all the applications and software packages installed on a device. A software inventory policy exists inside of a Device Manager Deployment Package. You can deploy the policy to any user group for any device platform.

1. Click the Deployments tab and then click New Package > New <platform> package.

2. On the Package Name page of the Create New Package wizard, enter a name for the app removal policy and then click Next.

3. On the Groups of users page, select the users from whose devices you want to remove the app and then click Next.

4. On the Resources to be deployed page, in Available Resources, select the Software Inventory policy, click the right arrow to move Software Inventory into the Resources to Deploy column and then click Next.

5. On the Deployment schedule page, configure the package to push the app Now or at a specified time in the future and then click Next.

6. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment and then click Next.

7. On the Package summary page, review the app removal package configuration and then click Finish.

To deploy the package and remove the app, select the package and then click Deploy. Connected devices receive the package as soon as scheduling rules are met.
To create and deploy a deployment package

1. In the Device Manager management console, click the Deployment tab.


3. On the Package Name page of the Create New Package wizard, enter a name for the app removal policy, and then click Next.

4. On the Groups of users page, select the users from whose devices you want to remove the app and then click Next.

5. On the Resources to be deployed page, in Available Resources, select the app removal policy you want to use for the package, and then click the right arrow button to add the resource to the package.

6. Click Next.

7. On the Deployment schedule page, configure to push the app no or at a specified time in the future.

8. Click Next.

9. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment. For more detailed information, see Deployment Rules.

10. Click Next.

11. On the Package summary page, review the app removal package configuration and then click Finish.

12. Click Deploy in the toolbar.

All connected devices receive all configured packages as soon as scheduling rules are met. Reconnecting devices receive the package when they connect subject to other rule criteria.
Configuring Deployment Rules

You can set any number of parameters that will affect the deployment outcome of a package.

For example, your package deployment could be based on a specific operating system version, on a particular hardware platform, or some other combination. In this wizard, you will find both a Simple and Advanced rule editor, with the Advanced view being a free-form editor.

Simple Deployment Rules

Simple deployment rules are comprised of pre-defined tests and resulting actions. Where ever possible, the results are pre-built into the example tests. For example, when basing a package deployment on a hardware platform, all existing known platforms are populated into the resultant test, drastically reducing your rule creation time, and limiting possible errors.

Click on New rule to add a rule to the package.

Note: The rule builder includes further information, specific to each test.

To create a new rule, you select a rule template, select the condition type, and then customize the rule. Customizing the rule includes modifying the description. When you finish configuring settings, you add the rule to the package.

You can add as many rules as you want. The package is deployed when all of the rules match.

Advanced Deployment Rules

If you click on the Advanced tab, the Advanced Rule Editor appears.

In this mode, you can specify what relationship is set between the rules. The operators AND, OR, and NOT are available.
Configuring Deployment Schedules

The Deployment schedule allows you to define when to deploy a package.

You can schedule the deployment for:

- A future time (Do not deploy).
- A single deployment (one time).
- A permanent deployment to make sure that the devices always have the package content (On every connection). This is designed to ensure the devices initially and continue to comply with your application policies.

The available options might change depending on the platform type.

You can configure the schedule to make sure that a package is only deployed one time. For example, if users change deployment settings, the package will not deploy again.

The windows differ by device type; for Windows Mobile devices the window appears as follows:

**Important:** Some devices may not observe the schedule. When selecting a precise date for the deployment, the targeted devices receive the information to try to reconnect during that time frame, even if they do not have a connection scheduling in place during the specified time frame. However, if the device does not connect or connects later than the configured time frame, the device does not receive the package.
Configuring Package Hierarchy and Relationship Rules

Hierarchy rules apply to assignments between packages and sub-packages. The following table shows how user group and package structures work.

<table>
<thead>
<tr>
<th>User Group Structure</th>
<th>Package Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC (Group parent)</td>
<td>XYZ</td>
</tr>
<tr>
<td>· Marketing (Group child)</td>
<td>· Marketing</td>
</tr>
<tr>
<td>· R&amp;D (Group child)</td>
<td>· RD</td>
</tr>
<tr>
<td>· Admin (Group child)</td>
<td>· Admin</td>
</tr>
</tbody>
</table>

Scenario 1: If the assignment was made at the parent package level but not at that of the sub-package, the latter inherits its parent package’s assignments. The conditions are:

- The XYZ package is not assigned to a specific group.
- The Marketing, R&D and Admin sub-packages are assigned to the ABC.Marketing, ABC.RD, and ABC.Admin subgroups, respectively.

The result is that the ABC Marketing, ABC RD, and ABC Admin subgroups inherit from the XYZ package solely because this package is not assigned.

Scenario 2: If the assignment was made at the parent package level as well as at that of the sub-package, the latter retrieves its own assignments alone. The conditions are:

- The XYZ package is assigned to the ABC group.
- The Marketing, R&D and Admin sub-packages are assigned to the ABC.Marketing, ABC.RD, and ABC.Admin subgroups, respectively.

The result is that the ABC Marketing, ABC RD and ABC Admin subgroups do not inherit from the XYZ package.

Note: You can restrict the deployment of a package to a subset of devices within the selected user group by defining rules.
With Automated Actions, you can configure Device Manager to perform actions based on user or device properties, events, or the existence of applications on devices.

For example, you can configure the following Automated Actions:

- You can automatically notify users whose iOS or Android devices is jailbroken or rooted that they are in violation of company policy and that the device will be selectively wiped if the device is not brought into compliance.

- You can automatically enforce a geo-fencing policy whereby if a user's device leaves a defined geographical perimeter, the device is blocked from accessing your organization's email, is selectively wiped, or is revoked.

- You can alert users automatically when mobile devices are roaming domestically or internationally and that they may be charged extra for the service.

- You can wipe a user's device automatically when the user leaves the company, and can disable the user's Active Directory account, so that the user can no longer access your organization's data.

- You can place a user's device into an Out Of Compliance state automatically if the user installs a blacklisted app, and you can send the user a notification informing them that they have broken the organization's mobile app policy.

**Note:** Before you can use Automated Actions to send automated notifications, make sure you have configured notification servers for SMTP and SMS so that Device Manager can send the message. For details, see To configure a Notifications SMTP Server and To configure an SMS Notifications Gateway.
How Automated Actions Work

You can configure Automated Actions in Device Manager to trigger an event when a user device is out of compliance. You configure the following settings when you configure Automated Actions:

- **Trigger.** The state that must exist to cause the event.
- **Condition.** The setting that defines the trigger explicitly.
- **Action.** The result that occurs if the trigger conditions are met.
- **Options.** The ability to delay an action to notify users of the policy violation and allow time for users to remedy the condition.

Before you start using Automated Actions, consider the following:

- If devices are shared between two users and you want to re-enroll the device to the second user, make sure that you delete the device entry from the Device Manager Devices tab before enrolling the second user.

- Automated Actions are only triggered when a device connects to Device Manager. For example, a notification is not sent to a device until the device attempts a connection back to the server. Likewise, if any of the managed devices are currently blocked by Secure Mobile Gateway, notifications are not sent to those devices until users initiate an Active Sync activity, such as receiving email or if the device synchronizes with Exchange.

- You can deploy Automated Actions to anonymous devices if you deploy the package to anonymous users. You cannot perform Notify (SMTP/SMS) Automated Actions on anonymous (unauthenticated) users.

- The only Automated Actions you can perform on unmanaged devices - that is, on devices that are revoked, have been selectively wiped, or are not enrolled -are the Notification and Set as Out Of Compliance actions.

- The Out Of Compliance action keeps a device in that state until another action explicitly changes the state of the Out of Compliance property.

- You cannot set the Secure Mobile Gateway block notification cannot on a device that is not enrolled.

- If you are using an Automated Action to detect when users disable their location servers on an iOS device and you want to send a notification, wipe, or revoke the device, you must enable Report if location services are disabled when you configure an iOS geo-tracking policy. For details, see To configure geo-tracking on iOS devices.

- If you want to create an Automated Action based upon a user whose Active Directory account is disabled, you can use the Event Trigger named ‘AD Disabled User’.
How Automated Actions Work

- If you create custom notification templates of the following type - Out of Compliance and AD Disabled User - you cannot select the templates when you configure an Automated Notification.

- There is a default one-hour waiting period for event-based triggers. Recurring notifications may be delayed due to the original event that causes the notification to be sent. For example, if you configure Device Manager to send a recurring notification every hour, but users do not receive the notifications. The reason for the delay is due to the fact that recurring notifications are not sent until the configured trigger occurs again after the Repeated Wait time expires.
Choosing Automated Actions Trigger Types

Triggers are the states, events or properties that cause an automated action to occur. There are four categories of triggers: Device Property, User Property, Applications, Event. Each trigger can contain multiple types.

The following table provides a few examples of triggers and trigger events.

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device Property</td>
<td>Useful device properties you can use as triggers for automated actions:</td>
</tr>
<tr>
<td></td>
<td>· Jailbroken or Rooted. If users jailbreak their device, you can set an action to notify the user and if the user does not undo the jailbreak in a given amount of time, selectively wipe the device.</td>
</tr>
<tr>
<td></td>
<td>· Out Of Compliance. If a device is put into a state of being out of compliance, you can block that device from the SMG (and thus corporate email) and notify the user.</td>
</tr>
<tr>
<td></td>
<td>· Passcode compliant. If this trigger’s condition is false, then you can set the device to Out Of Compliance or selectively wipe the device.</td>
</tr>
<tr>
<td></td>
<td>· Perimeter Breach. If the device leaves the geo-perimeter defined in an iOS geo-fencing policy, this condition can set and used to notify the user, wipe the device, and so on.</td>
</tr>
<tr>
<td></td>
<td>· Many more: Look in the Automated Actions dialog to view all user properties.</td>
</tr>
<tr>
<td>User Property</td>
<td>Useful user properties you can use as triggers for automated actions:</td>
</tr>
<tr>
<td></td>
<td>· Active Directory failed login attempts. If an Active Directory user attempts to log in more times that allowed, you can notify the user that they will have to wait a certain time period before they can try to log in again.</td>
</tr>
<tr>
<td>Applications</td>
<td>This trigger allows you to specify whether or not an app is installed on a device, by name, and then set an appropriate action such as notify or set the device as out of compliance.</td>
</tr>
</tbody>
</table>
The following system events can be used as triggers in automated actions:

- Secure Mobile Gateway block. A user's device has been blocked by the Secure Mobile Gateway and the device lost access to your organization's email.

- Device unmanaged. The device lost its ability to connect to and communicate with (and thus be managed by) the Device Manager server.

- Device jailbroken. A user broke the iOS user agreement and warranty in order to install unauthorized software.

- Device not blacklist or whitelist app compliant. A user's device breaks an app blacklist or whitelist policy, you can choose an action to perform.

- Device revoked. The device has lost ability to connect to the Device Manager server.

- Device international roaming. A user's device is roaming internationally.

- Device domestic roaming. A user's device is roaming domestically.

- Location perimter breach. A user's device has gone outside of a defined perimeter.

- Location services disabled. The location services on a user's device is disabled.

- Active Directory disabled. If you disable a user's Active Directory account, such as when an employee leaves a company.

### Types of Automated Actions

The following list details the types of actions you can configure to occur automatically based on trigger type.

- Selective Wipe. Clears organizational data from the device while retaining personal information and selected settings. The MDM profiles and all packages pushed by Device Manager to the device are removed. The device can, however, be re-enrolled at a future time.

- Revoke. Revoking a device prohibits any further connection from the device. If the device attempts to connect to Device Manager, the MDM profile and all packages deployed to the device are removed. The device is barred from re-enrollment unless it is re-authorized by an administrator.

- Set as Out Of Compliance. The device is given a property named Out of Compliance and the property is set to True. When a device is out of compliance (has this property set to true), then it appears in the Out of Compliance on the Dashboard Alerts widget.
1.

2. In the Location Services - Configuration creation dialog box, select Report if location services are disabled and then click Create.

3. In the New automated actions dialog box, do the following:
   a. In Name, type a name for the automated action.
   b. Under Trigger, in Trigger type, select Event and in Event, select Location services disabled.
4. Click Create.
Automated Actions Example: Notifications for Blacklisted Apps

This topic is an example procedure that illustrates using Device Manager Automated Actions to set up an automatic notification to inform users when they install a forbidden (also known as "blacklisted") app on their device. You can manage user devices to make sure that a work device installs the approved list of apps only, and that the device does not have any forbidden apps installed.

This example shows the following tasks:

- Configure the notification template you want to send.
- Create an Applications Access policy to designate an iOS app named Word with Friends for Free as forbidden (blacklisted).
- Create an Automated Action that sends a notification when a device violates a forbidden Applications Access policy.
- Deploy the Automated Action and Applications Access policy to your device in a deployment package.
- Install the Words with Friends for Free on your iOS device.
- Receive the Notification.

To configure a notification template

When users install a forbidden app on their device, you can send the correct notification by using a template for the message that is sent when the non-compliant blacklist or whitelist trigger is correctly configured.

By default, all notification templates are configured to use the $\{user.mail\}$ macro, which uses the email address of the device owner who receives the notification. If you want notification emails to be sent to an administrative user; for example, to notify an administrator that a device has been jailbroken, you can enter the administrator email address in the To field.

1. In Device Manager, click Options.
2. In the Server Options dialog box, in the left pane expand Notification Templates.
3. In the right pane, under Notification Templates, click Non Compliant Blacklist / Whitelist.
4. In the Edit a Notifications Template dialog box, on the Settings tab, in Channels, select the channels of communication you want to use.
5. Click the SMTP tab and do the following:

   a. In From, enter the name or email address from whom the notification is sent. This is not a mandatory field, however Citrix recommends adding the name or email address.

   b. In To, leave the command ${user.mail}. If you modify the To field, the email might not be sent correctly.

   c. In Message, you can modify the message except for the macros
      ${firstnotnull(device.TEL_NUMBER,device.serialNumber)} and
      ${outofcompliance.reason(whitelist_blacklist_apps_name)}. If you modify or remove the macros, the email might not be sent correctly.

6. Click Update. When you click Update the template is ready for the Automated Action.

Next you will create a blacklist for an app, so you can use the blacklisted app as a trigger for your automated action later. This example uses the Words with Friends Free app.

To create an Applications Access policy for a forbidden app

1. In the Device Manager web console, click the Policies tab.


3. Click New Applications Access Policy.

4. In the Add a new Applications Access Policy dialog box, type Words with Friends for Free.

5. In Access policy, click Forbidden (blacklist).

6. In OS type, select the iOS.

7. Click New app.

8. In the Add a new application dialog box, enter the following:

   a. In App Name, type the name of the app. For example, type Words with Friends Free.

   b. In App bundle ID, type the bundle name of the app. For example, type com.zynga.WordsWithFriendsFree.

9. Click Create. This will create the application in the list. The app appears in the list in the Add a new application dialog box.

10. Click Create again to create the Application Access Policy. Once created, you can add this policy to a deployment package and deploy to the devices you want to manage.

Next, you create an Automated Action that sends a notification email to users when they install a blacklisted app on their device.
To create an Automated Action

1. In Device Manager, click the Policies tab.

2. In the left pane, under Global, click Automated Actions and then in the right pane, click New.

3. In the New automated action dialog box, do the following:
   a. In Name, enter Blacklist Notify.
   b. Under Trigger, in Trigger Type, select Applications and in Name, select Installed.
   c. Under Condition, in Condition, select Is and then in Value, enter WordsWithFriendsFree.
   e. Under Action, in Template, select Non Compliant Blacklist / Whitelist.
   f. Under Options, select Delay and then configure 10 minutes.
   g. Under Options, select Repeat wait and configure one hour. This option allows you to delay sending the notification message in the event that there is a communication failure between the device and Device Manager.

4. Click Create.

In the last task, you will create a deployment package that contains Automated Actions and then push that deployment to user devices.

To deploy automated action and Applications Access policy to devices

Once on your device, you install the blacklisted app to trigger both the notification message that your device is out of compliance, and to trigger the Secure Mobile Gateway block on the server.

Citrix recommends that you create separate deployment packages for your Automated Actions and deploy them separately from other packages. Additionally, make sure you configure Deploy to anonymous users in the Groups of users page of the package, to include those users who may have removed their agents, or who have had their Active Directory account disabled.

You run the Create New Package wizard to deploy packages. During the wizard, you select the following:

- Groups to which the policy is deployed.
- Resources that include the Automated Actions you created and the Software Inventory resource.
1. In Device Manager, click the Deployment tab, and then click New Package > New iOS Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the package and then click Next.

3. In the Groups of users window, select a group you want to deploy this policy to and then click Next.

4. In the Resources to be deployed window, under Available Resources expand the Automated Actions section and select the two Automated Actions you previously created in the last step. Then, click the right arrow to add the resource to the deployment package.

5. Next, in the Available Resources list, under Applications Access Policy, select the Forbidden policy you previously created and click the right arrow button to add it to the package. Click Next.

6. In the Deployment schedule window, select the If not deployed Start Now option. Click Next.

7. In the Deployment rules page, click Next.

8. In the Package summary page, click Finish.

9. When the wizard is complete, in Device Manager, click Deploy to deploy the packages.

When Device Manager finishes the deployment, select the deployment package, and then click the Details button to see information about the success of the package deployment. When the package shows as deployed, then you can move on to the next step and install the blacklisted app on your iOS device.

When the users targeted in the deployment install the blacklisted app on their iOS device, Words with Friends Free, users receive a notification message that the app is not allowed.
Changing Device Compliance with Automated Actions

Automated actions allow you to change the status of a device from a state of compliance to a state of non-compliance based upon specific conditions. For example, you can set an Automated Action to change a device to a state of Out Of Compliance=True if the device has been jailbroken or rooted, if the user disabled location services on the device, or if the user installs a blacklisted application.

In a cases where a user’s device is put into an out of compliance state, and then the user fixes the device so that the device is in compliance, you will need to configure a policy to deploy a package that resets the device into a compliant state.

For example, let’s say you want to define the following three compliance policies in your organization by using Device Manager Automated Actions:

1. Location Services Policy. This policy states that if a user disables location services on their device, then the Automated Action should then set the device property Out of Compliance to True.

2. Blacklisted App Policy. This policy states that if a user installs a blacklisted app on their device, then the Automated Action should then set the device property Out of Compliance to True.

3. Jailbreak Policy. This policy states that if a user jailbreaks their device, then the Automated Action should then set the device property Out of Compliance to True.

Naming and Setting the Order of Deployment Packages

Device Manager deploys packages to target devices. When you create your Automate Action compliance policies, you need to name your policies in a very specific way, so that they are run in the correct order.

Device Manager deploys packages according to their name, deploying those packages in an alphanumeric order. Thus, you want to make sure that you deploy your Compliance Reset Automated Action package first and that it does not reset any of the other Automated Action packages that are designed to track compliance device compliance.

When you name your policies, make sure that the global compliance reset policies deploy first and then deploy your Automated Action compliance deployment packages.

In the example above, you might want to create three packages to track device compliance, the Geo-fence, Blacklist, and Jailbreak policies. Automated Actions tracks the devices and sets the devices to Out of Compliance=False when the user violates the policy. You also want to be able to reset the devices when the user brings the device back into compliance.
For example, you want to reset the device property when the Device Manager detects that
the device is out of compliance. You want this policy to run before any other policy. You
can provide the name *aaa-OOC-Reset* so that it will run before the policies that can set a
device out of compliance.

You can create an Automated Action by setting the device property to out of compliance if
users disables location services on the device. If you want this policy to run after the reset
policy, you can give the policy the name *aab-location-services-disabled*. You can then set
the delay for a specific number of minutes so this policy runs after the compliance reset
Automated Action that runs before this policy.

You can also create an Automated Action that sets the device property to out of compliance
if users install a blacklisted app on their device. You can give the policy the name
*aac-blacklisted-app* and set the delay for four minutes so it runs after the two policies
preceding this one.

You can create an Automated Action to set the device property to out of compliance if
users jailbreak or root their device. You can give the policy the name *aad-jailbreak* and set
the delay to five minutes so it runs after the three policies preceding this policy.

### Setting the Order of Compliance Packages

Your last step to make sure that your compliance policies run in the correct order. To do
so, create Deployment Packages and use the same names you used for the Automate
Actions. You follow the same principles in naming that you use for Automated Actions.
When you use this naming conventions, you can make sure that the packages deploy in the
same order as the Automated Actions.
Showing Automated Actions That Have Run

You can view all of the automated actions that have run from inside of Device Manager at any time.

1. In Device Manager, click the Policies tab and then select Show Executions.
Troubleshooting Automated Actions

To check whether or not automated notification was sent to a user, you can try a few things:

- Check the Deployment of the package that contained your Automated Action to make sure it is actually deployed.
- Check the Device Manager Device Event Log and see what if any of the events specified in your automated actions have run.
- Check the Device Manager Device Sent Notifications Log and see which notifications have been sent, which have failed, who received them, when they were sent, and more.
Configuring Notifications

You can use notifications in Device Manager to do the following:

- Communicate with select groups of users easily from the Device tab, such as all iOS device users, users whose devices are Out Of Compliance, all users with employee-owned devices, or all users with unmanaged devices, and so on.

- Enroll users and their devices into Device Manager.

- Automatically notify your users (through Automated Actions) when certain conditions are met, such as when a user’s device is about to be blocked from corporate access due to compliance policy violations, or when a user’s device has been jailbroken or rooted.

Notifications are used to send messages over three different channels: SMTP, SMS, and Agent Push (currently iOS only).

Before you can send notifications, you must configure a notifications server and a SMS gateway and carrier SMS gateway. Also, you must select a notification channel in the notification template.

**Note:** Port 25 must be opened from the Device Manager server located in your DMZ to point back to the SMTP server on your internal network for notifications to be sent successfully.
Sending Ad-Hoc Notifications

You can send a one time, ad-hoc notifications in Device Manager to single or multiple users directly to their devices using SMTP (email), SMS, or Agent Push.

1. On the Devices tab, select a single or select multiple devices. You can choose to filter the list of devices depending on your purpose. For example, you might want to send a message to all users who have jailbroken devices, or send a message to all users whose devices are listed as Out Of Compliance.

2. Select the devices to which you want to send notifications and then click Notification.

3. In the Notifications dialog box, enter the following information:
   a. From. Enter who you want to be shown as the sender of the notification (optional).
   b. To. The users associated with the devices you selected will automatically be added to the notification recipient list. If you want to add other users beyond the list of devices you selected, you can enter the user’s email address as known by Device Manager (case sensitive) and then click the plus icon to add the user.
   c. Template. You can choose a template to fit the purpose of your notification. For example, if you want to notify users whose devices have been jailbroken and are out of compliance, you can select a custom ad hoc notification template built for this purpose.
   d. Message. You can enter text, or if you choose a notification template, this field is populated with the text from the template.
   e. Channel. Select the communication channel you want to use to send the message, SMTP (email), SMS, or Agent Push (iOS only).

4. Before you send the notification, if you are sending the message via SMS, and you do not have a Nexmo subscription or SMS gateway server configured in Device Manager, click the Detailed Device List button to check if the recipients you have selected can be contacted through the Notification mechanism.

5. In the Detailed Device List dialog box, you can troubleshoot any of the devices that show red lights, which indicate channels of communication that are not currently working to send notification. The red lights indicate the recipients who may not receive the notification unless you add a carrier SMS gateway and address to use for sending the notification. The green lights in the SMTP column indicate that the SMTP server is functioning and will send the notification via email.

6. To manually enter an SMS carrier gateway and address, select the recipient and fill out the appropriate information.

7. When you are finished adding the SMS information, click Close.

8. Click Send to send the notification. Device Manager either delivers the message or queues it for sending. If the message is queued, the Sent Notification Log report indicates the results. Queuing occurs because either the system is busy (sending
To create a custom notification template in Device Manager

1. Click Options.

2. Click Notification Templates and then click New.

3. In the Create a Notification Template dialog box, on the Settings tab, enter the following information for your template:

   - Name. Enter a name for the template that indicates its use and purpose. For example, if this is a warning message regarding banned apps, you could name it Banned App Notification.

   - Description. Enter a brief description of this notification's purpose.

   - Notification Type. Determines the Automated Action event type the template is used with.

   - Channel. Select the channels through which you want to send the notification. Agent push is currently for iOS only.

4. Click the SMTP tab and then enter the following information:

   - From. (Optional) Name used in the email From field. Only enter a value here if you do not want to use the default value from the Notification Server definition.

   - To. An email address, system macro, or list (delimited by semicolons). System macros are used when sending automated action notifications. The system macro ${user.email} is the default To field.

   - Subject. Enter a generic subject line for the message.

   - Message. Enter message text. If you want to use system macros in your custom notification template, open one of the predefined notification templates and borrow one of the commonly used macros, such as the macros used for users or devices.

5. Click the SMS tab and then enter the following information:

   - To. A system macro or mobile number. There are two system macros for use in enrollment templates and non-enrollment templates. For enrollment templates, use ${user.mobile}. For non-enrollment templates, use ${firstnotnull(device.TEL_NUMBER,user.mobile)}

   - Message. Enter a message text that the user will see when the message is received.

6. Click the Agent tab and then enter the following information to be used for agent push notifications (iOS only):

   - To. Enter the following variable - ${device.TOKEN} - for the device's token ID, which is used to identify and communicate with the device via agent push notification.
To create a custom notification template in Device Manager

- **Message.** Enter a message text that the user will see when the message is received.
- **Sound File.** Select a sound file to be played when the user receives the push notification on their device.

7. When you are finished, click Create.
Using Notification Templates

You can use notification templates in Device Manager when you do the following:

- Send enrollment invitations inviting users to enroll their devices.
- Send ad hoc notifications notifying users their devices are jailbroken or letting users know important IT information, without using a template.
- Configure Automated Actions to send notifications, such as an automatic notification when a user’s device has a blacklisted app or has moved beyond an organization-defined geo-fencing policy.

Device Manager comes with a set of predefined templates that reflect the capabilities of the Automated Actions feature. Each template reflects a distinct type of event that Device Manager automatically responds to for each and every device in the system.

You can modify a pre-defined notification template, but you cannot delete one. Citrix recommends that you do not edit or modify the macros (for example, ${user.mail}) used inside of pre-defined templates, or they may not work.

The following table describes the predefined notification templates that come with Device Manager:

<table>
<thead>
<tr>
<th>Template name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android Download Link</td>
<td>Provides a download link Web address for users who are enrolling their Android devices into Device Manager.</td>
</tr>
<tr>
<td>Enrollment</td>
<td>Provides a Web address to the Device Manager server that allows users to enroll their devices.</td>
</tr>
<tr>
<td>Enrollment URL</td>
<td>Provides a special enrollment Web address that allows users to enroll their devices securely, combined with other forms of authentication, depending on the chosen enrollment mode.</td>
</tr>
<tr>
<td>Enrollment PIN</td>
<td>Provides a one-time generated PIN that is used in PIN based enrollment modes</td>
</tr>
<tr>
<td>iOS Download Link</td>
<td>Provides a download link Web address for users who are enrolling their iOS devices into Device Manager.</td>
</tr>
<tr>
<td>Jailbroken Device</td>
<td>Provides a message indicating that a specific device has been jailbroken.</td>
</tr>
<tr>
<td>Location Perimeter Breach</td>
<td>Provides a message informing a user that the device has gone outside of a predefined geo-fencing perimeter and thus could be blocked from corporate access.</td>
</tr>
<tr>
<td>Notification Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Location Services Disabled</td>
<td>Provides a message informing a user that the device has had its location services turned off and thus could be blocked from corporate access.</td>
</tr>
<tr>
<td>Non-Compliant Blacklist/Whitelist</td>
<td>Provides a message informing a user that their device has an app installed that violates a corporate blacklist or whitelist policy.</td>
</tr>
<tr>
<td>Revoked Device</td>
<td>Provides a message informing a user that the device has been revoked and that any further connection from the device to Device Manager is prohibited. The device is barred from reenrollment unless it is reauthorized by an administrator.</td>
</tr>
<tr>
<td>Roaming Domestic</td>
<td>Provides notification when device is roaming domestically across carrier network, indicating both device and user name associated with the device.</td>
</tr>
<tr>
<td>Roaming International</td>
<td>Provides notification when device is roaming internationally across carrier network, indicating both device and user name associated with the device.</td>
</tr>
<tr>
<td>SMG Blocked</td>
<td>Provides a message stating that a specific user's device has been blocked because it has violated a specific compliance policy.</td>
</tr>
<tr>
<td>Unmanaged Device</td>
<td>Provides a message indicating that a specific user's device has become unmanaged (possible due to un-installation of Device Manager agent or certificates) and must be reenrolled by a specific date or the device will no longer have access to corporate email.</td>
</tr>
</tbody>
</table>
System Macros in Device Manager
Notification Templates

Notification templates in Device Manager use the following system macros when you use the Automated Actions feature for automated sending. Citrix recommends that you do not modify macros in templates or else the notifications may not work.

- Notifications are sent to the correct SMTP recipient address. For example, ${user.mail}.

- Enrollment invitation Web addresses use the proper syntax to ensure secure authentication. For example, ${enrollment.url}.

- Enrollment PINs can be generated. For example., ${enrollment.pin}.

- The correct Device Manager server host name is used. For example, http://${zdmserver.hostPath}/enroll.

- The correct user device (ID, name, and so on) is used when sending notifications. For example, ${firstnotnull(device.TEL_NUMBER,device.serialNumber)}.

- The cause of a automated notification is given to the user. For example, ${outofcompliance.reason(smg_block)}. 
Configuring General Device Manager Options

You can use General Options to set general Device Manager device display settings, device access relative to number of users per device, device triangulation enablement, and the Enterprise App Store availability for iOS.

- Inactivity Days Threshold. Defines a time period in days within which a device must communicate back to the Device Manager server before changing the device status to "inactive".

  **Note:** If you are using Cisco ISE (or other NAC appliance) in conjunction with the Device Manager server to filter device access to your network, and if the Inactivity Days Threshold value changes, restart the Device Manager service on the Device Manager server for the changes to take effect.

- Number of Devices per User. Maximum number of devices a user can enroll. If you want to prevent device sharing, you can restrict the number of users per single mobile device, as well as restrict the number of devices that a single user can register and enroll. If you set the value to zero that means a user can own any number of devices. When a device or user limit is exceeded, the users receive an error that indicates that a connection or license limit is reached, which prevents the additional user or device from enrolling.

- Number of Users per Device. Maximum number of users that can share a single device. The default value is zero, which means an unlimited amount of users can share the device.

- Highlight Jailbroken or Rooted column, SMG Status column, Managed column. When enabled, these options provide status "lights" to indicate a device status. When disabled, the status lights (red or green) will not display and text will be used to indicate status.

- Enable Device Triangulation. Enables devices to be located geographically for GPS geo-location and geo-fencing policies.

- Send Android Domain Users to Secure Mobile Gateway. When enabled, this option ensures that Device Manager sends Android device information to Secure Mobile Gateway in the event that Device Manager does not have the Android device user's ActiveSync identifier (ID).
Configuring Device Manager Security Options

The security options dialog box allows to customize the security features of the service. By default, when Secure Device is included in the license, it is automatically activated during installation, with a strong level of security. If you need to change those parameters, use that dialog box.

- Enforce SSL. Forces devices to communicate by using an SSL transport. All HTTP (unsecure) requests from devices will be rejected.

- Strong Authentication. Enables strong authentication by generating a Strong ID for devices that is then used as a second method of authentication during the enrollment process.

- Strong ID Valid Once. Allows Strong ID passcodes to only be used once. When the Strong ID is used once to generate a device certificate, it cannot be reused. The device has to be revoked and re-authorized.


- Always Add Device. Registers devices automatically into Device Manager even when Secure Device is activated.

- Block Rooted Android and iOS Enrollment. Enabling this function blocks rooted or jailbroken devices from enrolling.

- 8 Char Strong ID. Enables a Strong ID character string that is limited to 8 characters.

- Enable SHP Console for Users. Enables or disables the Self-Help Console for user management of devices.

- XDM/SHP console max inactive interval. The time (in minutes) between client requests before the server invalidates a log on session. If you set the value to zero, log on sessions do not timeout. For example, if the console max timeout value is set to 1 (one minute) and a user logs on and does not interact with the UI for over one minute, then the user is logged off. The console might still appear as if the user is logged on until the user attempts to interact with the UI, but then the console will be refreshed and the user will see the log on page.

- iOS agent auto logout (minutes). Length of time before an iOS agent user is logged off due to inactivity.

- Enable client cert authentication for iOS. If enabled, iOS enrollment agent uses certificate authentication. If disabled, iOS enrollment agent uses session-based authentication.
To enable Strong ID

Strong ID is a form of 2 factor authentication used to provide an extra layer of extra security when enrolling a device. Devices cannot enroll until the device’s serial number or IMEI is known. When you enable Strong ID, Citrix recommends enabling the character string to be 8 characters in length.

1. In the Device Manager console, click Options > Security.

2. You can add the devices manually or import the devices from the Devices tab by using the serial number of IMEI, which generates a Strong ID for the device.

When users are ready to enroll their device, users need to call support personnel and give the serial number or IMEI. Support personnel can then proved the Strong ID from the device properties.
Configuring Role-Based Access Control

You can configure the following settings for role-based access control:

- Access Role Based Access Control Settings
- Create a New Access Control Role (Associate Actions with Roles)
- Add Groups to a Role
- Associate Users with Roles

To configure role-based access control

1. In the Device Manager Options console, in the left pane, expand Access Control and then click Role Based Access Control.

2. In the right pane, click New.

3. In the Create a Role dialog box, enter a name for the role, select the features you want to enable for the role and then click Create.

To add groups to a role

When you create a new role, you can also associate a user group with the role as part of the role definition.

1. In the Device Manager Options console, in the left pane, expand Access Control and then click Role Based Access Control.

2. In the right pane, select a role and then click Edit.

3. In the Edit a role dialog box, select the feature access you want to associate with a role, and then select the group you want to have access to the role. Any group, and the group's users, that you select receives access to the selected features.

4. Click Update to save the changes.

To associate users with a role

When you create a new role, you can associate users with the role.

1. Select the Users tab and double-click a user in the user table. Or, click New User.

2. In the New User dialog box, enter the user name and password, and then in Role, select the role you want to associate with the user.
3. Click Create.
Configuring System Settings for iOS

The following system settings apply to your iOS Devices only:

- Store User Password. Provides the following options:
  
  - Enable. If you select Enable, a user's password on the iOS Connect app is securely stored and used for ongoing authentication with the Device Manager. On the user's device Connect app, the logon/logout button will be enabled, and the user will be required to log in again if the user manually logs out.

  - Disable. If you select Disable, Device Manager does not store a user's passwords and uses a certificate for all ongoing authentication with Device Manager. On the user's device Connect app, the logon/logout button will not display, and the user will never be logged out.

Note: Note that when this setting is selected, you can allow users to register and authenticate with a domain password because an enrollment invitation overrides this setting when other enrollment modes are configured.

- User property for VPP country mapping. The mapping used to choose the property pool of the Apple Volume Purchase Plan. This code allows a user to download apps from app stores specific to country based on this mapping. For example if your user property is US, you will not be able to download the apps if the VPP code for the app is distributed in the UK.
Scheduling Option for Hardware and Software Inventory

The Scheduling option enables you to globally enforce hardware inventory and software deployments for those devices that are always connected to Device Manager.

You may want your devices configured to always be connected to Device Manager ('Always On' or 'Permanently Alive'); for example, you may want a device to be always connected to Device Manager in the event you need to remotely wipe the device in case of a data security breach. Using policies, you can configure your devices to always be connected to Device Manager.

Using this option allows you to set the time interval (in minutes) that a hardware inventory and a software deployment runs.

For more information, see Configuring Deployment Schedules.
Configuring Macros

Device Manager provides powerful macros that provide a method to populate user or device property data within the text field of any profile or policy or notification/enrollment templates (for some automated actions), to name a few usages. With macros, an administrator can configure a single policy and deploy it to a large user base and have user-specific values appear for each targeted user. For example, an administrator could pre-populate the mailbox value for a user in an Exchange profile across thousands of users.

This section provides an overview on the use of macros in Device Manager.

This feature is currently only available in the context of configurations and templates for iOS and Android devices.
Defining User Macros

The following user macros are always available:

- loginname (username + domainname)
- username (loginnname minus the domain, if any)
- domainname (domain name, or the default domain)

The following administrator-defined properties may be available:

- c
- cn
- company
- companyname
- department
- description
- displayname
- distinguishedname
- facsimiletelephonenumber
- givenname
- homecity
- homecountry
- homefax
- homephone
- homestate
- homestreetaddress
- homezip
- ipphone
- l
- mail
Defining User Macros

- middleinitial
- mobile
- officestreetaddress
- pager
- physicaldeliveryofficename
- postalcode
- postofficebox
- telephonenumber
- samaccountname
- sn
- st
- streetaddress
- title
- userprincipalname
- domainname (overrides property described above)

Additionally, if the user is authenticated by using an authentication server, such as LDAP, all the properties associated with the user in that store are available.
Macro Syntax

A macro can take the following form:

- ${type.PROPERTYNAME}
- ${type.PROPERTYNAME ['DEFAULT VALUE'] [ | FUNCTION [(ARGUMENT1, ARGUMENT2)]]}

As a general rule, all syntax following the dollar ($) sign must be enclosed in curly brackets ({ }).

- Qualified property names reference either a user property, a device property or a custom property.

- Qualified property names consist of a prefix, followed by the actual property name.

- User properties take the form ${user.[PROPERTYNAME] (prefix="user.")}.

- Device properties take the form ${device.[PROPERTYNAME] (prefix="device.")}.

For example, ${user.username} populates the username value in the text field of a policy. This is useful for configuring Exchange ActiveSync profiles and other profiles used by multiple users.

For custom macros (properties that you define), the prefix is ${custom}.. You can omit the prefix.

Note: Property names are case-sensitive.
Viewing Reports

The Device Manager server repository database keeps a log of connections and data exchanges between each mobile device and the Device Manager server (Logs). Device Manager reporting provides detailed information such as by tunnel or by user. Device Manager reports are available through an integrated set of reports.

The Display a Report collection provides the following reports that assist you understand and manage your mobile device asset base:

- Session report (connection logs)
- Administrator options
- Groups, users and roles summary
- Device Software Report
- Hardware inventory
- Deployment Rule Report per device
- Deployment Rule Report per package
- Jailbroken or rooted devices
- Inactive devices
- Device enrollment
- Distribution of devices
- Blacklist / Whitelist application compliance report
- Device Events
- Terms and Conditions

You can also export reports to a Microsoft Excel CSV file and delete reports from Device Manager by using the Manage Historical Data Collection. When you export the report to a CSV file, Device Manager creates a text file containing all of the activity report data for the specified range of dates.

The Delete option removes data logged before a specified date from Device Manager. Use this option carefully; it cannot be undone.
Navigating Reports

Each report uses a navigation bar to aid in moving through the report and its sections. The navigation bar allows you to export the report data, print the report, hide/reveal subsections of the report, page throughout the report, search for a specific string, and set a zoom level for the rendered data.

Many Device Manager reports present data by using a summary page, followed by one or more subsections that provides additional detailed information. You can use the Group Tree icon in the report navigation bar to view the subsections and open that subsection’s page.

Additionally, you can use the drop-down list in the navigation bar to go to a particular subsection.

Some Device Manager reports require parameters to run; parameters are supplied to a report from pop-up windows.
Report Types

You can view the following report types in Device Manager.

Session Reports (Connection Logs)

This report is a summary of mobile device activity. It includes total usage per user and overall data compression ratio.

The connection logs can contain a large amount of data. The date is created over a period of time by the Device Manager server and it is stored in the Device Manager repository database. Citrix recommends that you limit the use of connection logs to processing of small datasets to avoid impacting the performance of the Device Manager server.

Content reports, which are part of session reports, provide a summary of total data usage for a specified period of time. You create this report by using a custom date range. The report includes:

- List of users connected to the Device Manager server. You can view details of a specific user in the list.
- Real volume passing through this Device Manager server.
- Data traffic optimized by the Device Manager server.
- Percentage of data compression achieved by the Device Manager server or Agent software for all data streams (incoming raw data as opposed to actual data transmitted over-the-air).

Other Reports

In addition to session reports, you can also view the following reports:

- Administrator operations. Summarizes administrator activity, including insertions, updates and deletions of any resources in database.

- Groups, users, and roles summary. Summarizes the list of groups, roles and users defined in the Device Manager server, and reports the modification and creation dates of these elements. This report provides an administrative overview of all users, roles and groups creation and modification data, and is meant to assist in the administrative side of your mobile IT infrastructure.

- Device software report. Provides a summary of the installed applications within the mobile device environment.

- Hardware inventory report. This report summarizes the mobile device asset base by hardware property - such as operating system, operating system version, platform, or device type.
Report Types

- Deployment rule report per device. This report summarizes package deployments for each device. *State* refers to the deployment state; specify All states, Pending, Successful, Failed, or Not applicable.

- Deployment rule report per package. This report summarizes package deployments for each package. *State* refers to the deployment state; specify All states, Pending, Successful, Failed, or Not applicable.

- Jailbroken or rooted devices. Lists jailbroken iOS devices and rooted Windows, Android, and Symbian devices.

- Inactive devices. Lists devices that are inactive.

- Device enrollment. Lists devices enrolled during a specified period of time.

- Device distribution. List of devices owned by employees or by your organization.

- Blacklist and whitelist application compliance report. This report provides three options for device compliance reporting:
  
  - Blacklisted apps shows devices with apps that are not allowed and are installed on the user device.
  
  - Non-whitelisted apps shows devices with apps installed on the device that are not on the whitelisted.
  
  - Missing whitelisted apps shows devices that do not have all the whitelisted apps present.
Adding User-Defined Reports

Reports must be in Crystal Report report format (file with a .rdp extension).

When you configure the reports, the property fields can have the following values:

- *reportFilename*. “My_report.rpt” is the personalized report in the Crystal Report.
- *format name* is the text that appears on the Device Manager Administration Console tab when you click on the link Click here for my report.
- *linkLabel* is the hypertext used to generate the report.
- *description* is the help that appears below the hypertext.

The link of the new user-defined report appears under the User-defined reports section on the Reports tab in Device Manager web console.

To add user-defined reports

1. Stop the Device Manager Server service.

2. Open the WEB-INF/classes/external-reports.xml file with a text editor that can read and write UTF-8 files, such as Notepad.

3. In the file, locate the tags `<list></list>` and add the following parameters:

   ```xml
   <bean class="com.sparus.nps.reports.EWCrystalReport">
   <property name="reportFilename" value="my_report.rpt"/>
   <property name="name" value="my_report"/>
   <property name="linkLabel" value="Click here for my_report"/>
   <property name="description" value="Description of my report"/>
   </bean>
   ```

4. Save the file in UTF-8 format.

5. Restart the Device Manager server service.
Managing Security and Identity

In Device Manager, you use certificates to create secure connections and authenticate users.

To establish a secure connection, a server certificate is required at one end of the connection. A root certificate of the Certificate Authority (CA) that issued the server certificate is required at the other end.

- **Server certificate.** A server certificate certifies the identity of a server. Device Manager requires this type of digital certificate.

- **Root certificate.** A root certificate identifies the CA that signed the server certificate. The root certificate belongs to the CA. The user device requires this type of digital certificate to verify the server certificate.

You can submit certificates for signing to a CA who signs the certificate and returns it to you.

In addition to certificates, you can configure security and identity in Device Manager in the following ways:

- Configure Device Manager by using Microsoft Certificate Services to generate user certificates for certificate-based authentication with WIFI, VPN, and Exchange ActiveSync profiles. You can also configure Device Manager as the CA to generate requests and to issue device identity certificates with Microsoft Certificate Services.

- Configure your own SAML service and identify provider in Device Manager. The SAML-based infrastructure can authenticate users and their mobile devices.

- Include Secure Device in your license that is activated automatically when you install Device Manager. Secure Device provides a strong level of security for user devices.

- Enable Strong ID that is a form of two-factor authentication. This provides extra security when enrolling devices in Device Manager.

- Configure filters in Device Manager that work with Network Access Control. Filters set users devices to be either compliant or not compliant. If a device is not compliant, the device is blocked from accessing the internal network.
The XenMobile Public Key Infrastructure (PKI) Integration feature allows you to manage the distribution and life-cycle of security certificates used on your devices with great flexibility.

The main feature of the system is the PKI Entity. A PKI entity models back-end component for PKI operations. That component may be either local to XenMobile (internal) or a part of your corporate infrastructure (external, such as a Microsoft, RSA, or OpenTrust PKI). The PKI entity handles the back-end certificate issuance and revocation. It is the authoritative source for the certificate’s status. The XenMobile configuration will normally contain exactly one PKI Entity per back-end PKI component.

The second feature is the Credential Provider. A Credential Provider is a particular configuration of certificate issuance and life-cycle. It will control things like the certificate’s format (subject, key, algorithms) and the conditions for its renewal or revocation, if any. The Credential Providers delegate operations to the PKI Entities. In other words, while Credential Providers control when and with what data PKI operations are undertaken, PKI Entities control how those operations are performed. The XenMobile configuration will normally contain many Credential Provider per PKI Entity.

The third feature of the system are Server Certificates. Server Certificates are X.509 certificates used functionally by the PKI Entity or the Credential Provider configurations.
Server Certificates

Server certificates are certificates used functionally by the XenMobile server that are uploaded into the Device Manager web console in the PKI integration section of the Options dialog box. They include CA (Certificate Authority) certificates, RA (Registration Authority) certificates, certificates for client authentication with other components of your infrastructure. In addition, you may use it as a storage for certificates you wish to deploy to devices. This will especially apply to CAs used to establish trust on the device.

XenMobile may or may not possess the private key for a given certificate. For some certain usages, XenMobile will require the private key, whereas for others, it will not. Each certificate you upload will be represented by an entry in the Server Certificates table, summarizing its contents. Later on, when you configure PKI integration components that require a certificate, you will be prompted to choose from a list of those Server Certificates that satisfy the context-dependent criteria.

For example, you might want to configure XenMobile to integrate with your Microsoft CA. The connection to the Microsoft CA should be authenticated using a client certificate. You can upload the CA certificate (without the private key) the CA will use to sign requests, and an SSL client certificate (with the private key) client authentication. When configuring the Microsoft CA entity, you need specify the CA certificate, which you can then select from a drop-down list with all Server Certificates that are CA certificates (context-dependent criterion). Likewise, when configuring client authentication, you can select from a drop-down list with all the Server Certificates for which XenMobile has the private key (context-dependent criterion).
To import a server certificate

XenMobile supports the following input formats for certificates:

- PEM or DER-encoded certificate files
- PEM or DER-encoded certificate files with associated PEM or DER-encoded private key file
- PKCS#12 key stores (P12; also known as PFX on Windows)
- Java Key Store (JKS) and Extended Java Key Store (EJKS)

Key stores, by design, can contain multiple entries, so when you loading from a key store, you will be prompted to specify the entry alias identifying the entry you wish to load. If you do not specify an alias, the first entry from the store will be loaded. Since PKCS12 files usually contain only one entry, you should leave the alias empty for those files.

When importing a certificate, either from a file or a key store entry, XenMobile will attempt to construct a certificate chain from the input, and will import all certificates in that chain (creating a Server Certificate entry for each). This will only work if the certificates in the file or key store entry really do form a chain, such as if each subsequent certificate in the chain is the issuer of the previous one. You can add an optional description for the imported certificate for heuristic purposes. The description will only be attached to the first certificate in the chain (you can update the description of the remainders later on).

1. From the Device Manager web console, click Options.
2. In the XenMobile Server Options dialog box, from the left side select PKI > Server Certificate.
3. Click Upload Certificate to import a certificate.
4. From the Certificate Type list, select either Certificate or Keystore.
5. Next, click Choose File to select a certificate.
6. Next, click Choose File to select a private key file for the certificate.
7. Enter an optional description, and then click Upload.
Updating a Certificate

XenMobile only allows one certificate per public key to exist in the system at any given time. If you attempt to import a certificate for the same key pair as an already imported one, you will be presented with the option to either replace the existing entry or to delete it.

To most effectively update your certificates, simply upload the new one in the Device Manager web console’s Options dialog box, under PKI > Certificates. When a Server Certificate entry is updated, components that were using the previous one will automatically switch to using the new one. Likewise, if you have deployed the Server Certificate on devices, it will automatically be updated on the next deployment.
PKI Entities

A XenMobile Public Key Infrastructure (PKI) Entity configuration represents a component performing actual PKI operations (issuance, key escrow, revocation, status information). These components may either be internal to XenMobile, in which case they’re called discretionary, or external to it, if they are part of your corporate infrastructure.

XenMobile supports the following three types of PKI entities:

- Discretionary CAs
- Generic PKIs (GPKI)
- Microsoft Certificate Services
Common PKI Concepts

Regardless of its type, every PKI Entity is said to have a subset of the following capabilities:

- **sign** Issuing a new certificate, based on a Certificate Signing Request.
- **fetch** Recovering an existing certificate and key pair.
- **revoke** Revoking a client certificate.

Table 1. PKI Capabilities

<table>
<thead>
<tr>
<th>PKI Type</th>
<th>Capability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discretionary</td>
<td>Sign</td>
</tr>
<tr>
<td>PKI</td>
<td>The adapter is capable of taking Certificate Signing Requests,</td>
</tr>
<tr>
<td></td>
<td>transmitting them to the PKI and returning newly signed certificates.</td>
</tr>
<tr>
<td>Microsoft</td>
<td>Sign</td>
</tr>
</tbody>
</table>

About CA Certificates

When configuring a PKI entity, you will have to inform XenMobile which CA certificate is going to be the signer of certificates issued by (or recovered from) that entity. One and the same PKI entity may return (fetched or newly signed) certificates signed by any number of different CAs; the certificate of each of these CAs will have to be provided as part of the PKI entity configuration, by uploading it to the Server Certificates repository and then referencing them in the PKI entity. For discretionary CAs, this will implicitly be the signing CA certificate, but for external entities, you will have to specify it manually.

**Note:** XenMobile will verify that the actual issuer of a certificate newly obtained through a sign or fetch operation matches the purported, configured issuer. An error will be raised if this is not the case.
Discretionary CAs

A Discretionary CA is created by providing XenMobile with a CA certificate and the associated private key. XenMobile will handle certificate issuance, revocation, and status information internally, according to the parameters you specify. However, XenMobile will never store the private keys of issued certificates, and so will not offer escrow services. Status information for certificates issued by a discretionary CA.

When configuring a Discretionary CA, you will have the option to activate OCSP support for that CA. If, and only if, OCSP support is enabled, the CA will add an id-pe-authorityInfoAccess extension to the certificates it issues, pointing to XenMobile’s internal OCSP Responder located at:

https://[server]/[instance]/ocsp

When configuring the OCSP service, you will have to specify an OCSP signing certificate for the Discretionary Entity in question. You can use the CA certificate itself as the signer. If you wish to avoid the unnecessary exposure of your CA’s private key (recommended), you will have to create a delegate OCSP signing certificate, signed by the CA certificate and including an id-kp-OCSPSigning extendedKeyUsage extension.

The XenMobile OCSP Responder service supports Basic OCSP responses and the following hashing algorithms in requests:

- SHA-1
- SHA-224
- SHA-256
- SHA-384
- SHA-512

Responses are signed with SHA-256 and the signing certificate’s key algorithm (DSA, RSA or ECDSA).
The Generic PKI (GPKI) protocol is a proprietary XenMobile protocol running atop a SOAP Web Service layer for purposes of uniform interfacing with various PKI solutions. The GPKI protocol defines three fundamental PKI operations:

- **sign** The adapter is capable of taking Certificate Signing Requests (CSR), transmitting them to the PKI and returning newly signed certificates.

- **fetch** The adapter is capable of retrieving (recovering) existing certificates and key pairs (depending on input parameters) from the PKI.

- **revoke** The adapter is able to cause the PKI to revoke a given certificate.

The receiving end of the GPKI protocol is the GPKI Adapter. The adapter translates the fundamental operations to the specific type of PKI for which it was built (in other words, there is a GPKI Adapter for RSA, another for OpenTrust, and so on).

**Figure 1. GPKI Communication Overview**

The GPKI Adapter, being a SOAP Web Services endpoint, publishes a self-describing WSDL. Creating a GPKI PKI Entity amounts to providing XenMobile with that WSDL, either through a URL or by uploading the file itself.

Support for each of the PKI operations in an adapter is optional. If an adapter supports a given operation, it is said to have the corresponding capability (sign, fetch or revoke). Each of these capabilities may be associated with a set of user parameters.

User parameters are parameters that are defined by the GPKI adapter for a specific operation and for which you need to provide values to XenMobile. Which operations the adapter supports (which capabilities it has) and which parameters it requires for each of them is determined by XenMobile by parsing the WSDL. The connection between XenMobile...
Generic PKI (GPKI)
Microsoft Certificate Services

XenMobile interfaces with Microsoft Certificate Services through its web enrollment interface. It only supports issuing new certificates through that interface (the equivalent of the GPKI sign capability).

To create a Microsoft CA PKI Entity in XenMobile, you must specify the base URL of the Certificate Services web interface. The connection between XenMobile and the Certificate Services web interface may optionally be secured using SSL client certificate authentication.

Note: This integration method is historical and limited. It will be migrated to the GPKI protocol in the future.
Migrating Previous PKI Configurations

Since the new XenMobile PKI integration capabilities have been significantly enhanced, migrating to the new system is not automatic. If you had used PKI configurations in previous versions, you will be able to continue to use these in 8.0 without changes, but if you wish to make use of the new capabilities, you will have to manually upgrade existing PKI entities.

Your pre-8.0.1 PKI entities (Microsoft CA or GPKI) will appear in the list of PKI entities, but will be marked as not ready to be used, indicated by a red icon in the Valid column in the Options dialog box, under PKI > Entities.

To ready the entity for 8..10 usage, edit the entity and provide the missing settings (the system will indicate which settings are missing when you try to save the configuration). This process requires providing the CA certificate(s) for the entity.
Credential Providers

Credential Providers are the actual configurations you will use in the various parts of the XenMobile system. They define the sources, parameters, and life-cycles of your certificates, whether these are part of device configurations or stand-alone, that is pushed as is, to the device.

The certificates’ life-cycle is constrained by the device enrollment. That is, no certificates are issued before enrollment, although some may indeed be issued as part of enrollment, and all certificates issued within the context of one enrollment are revoked when the enrollment ends.

Figure 1. Certificates Lifecycle
One Credential Provider configuration may be used in multiple places, to the effect that configuration may govern any number of certificates at the same time. The unicity, then, is on the deployment resource and the deployment: if the Credential Provider P is “deployed” to device D as part of the configuration C, then P’s issuance settings will determine the certificate that is deployed to D, its renewal settings will apply when C is updated, and its revocation settings will apply when C is deleted or D is revoked.

With the aforementioned in mind, the Credential Provider configuration:

- Determines the source of certificates — that is, which PKI Entity certificates will be obtained from
- Determines the method using which certificates are obtained — signing a new certificate or fetching (recovering) an existing certificate and key pair
- Determines the parameters for the issuance or recovery (for example, CSR parameters such as key size, key algorithm, distinguished name, certificate extensions, and so on)
- Determines the manner in which certificates are delivered to the device
- Determines revocation conditions. While all certificates are revoked when the management relationship is severed, the configuration may specify an earlier revocation, for instance when the associated device configuration is deleted. In addition, under some conditions the revocation of the associated certificate in XenMobile may be sent to the back-end PKI; that is, its revocation in XenMobile may cause its revocation on the PKI
- Determines renewal settings. Certificates obtained through a given Credential Provider may be automatically renewed when they near expiration, or, separately from that, notifications may be issued when that expiration approaches.

To what extent various configuration options are available will mainly depend on which type of PKI Entity and issuance method are selected for a Credential Provider.
Methods of Certificate Issuance

There are two fundamental methods of obtaining a certificate, which in this context shall be called methods of issuance:

- SIGN. With this method, the issuance involves creating a new key pair, creating a Certificate Signing Request (CSR) for the key pair, and submitting it to a CA for signature.

- FETCH. With this method, the issuance (from the point of view of XenMobile) is in actuality a recovery of an existing certificate and key pair.

A Credential Provider uses exactly one of these methods; which method is selected impacts which configuration options are available. Notably, CSR configuration and distributed delivery are only available if the issuing method is sign. If the certificate is fetched, it is always sent as a pkcs#12 to the device (equivalent to centralized delivery mode for the sign method).

Which issuing methods are available for a Credential Provider will depend on the capabilities the PKI Entity it uses supports.
Certificate Delivery

An important notion is the delivery mode of certificates. The delivery is independent of the issuance, although it only applies when the issuing mode is newly issued [sign], not recovered [fetch] from the PKI).

Two modes of certificate delivery are available: centralized and distributed. Distributed mode uses the SCEP protocol and is only available in situations where the client supports the protocol, and is even mandatory in some situations.

For a Credential Provider to support distributed (SCEP-assisted) delivery, a special configuration step is necessary: setting up Registration Authority (RA) certificates. Those are required because when using the SCEP protocol, XenMobile acts like a delegate (a registrar) to the actual CA, and must prove to the client that it has the authority to act as such. That authority is established by providing XenMobile with the aforementioned certificates.

Two distinct certificate roles are required (although one and the same certificate can fulfill both): RA signature and RA encryption. The constraints for these roles are as follows:

- The RA signing certificate must have the X.509 key usage digital signature.
- The RA encryption certificate must have the X.509 key usage key encipherment

To configure the Credential Provider’s RA certificates, you must first upload them to the Server Certificates repository, and then link to them in the Credential Provider.

A Credential Provider is considered to support distributed delivery if, and only if, it has a certificate configured for each of the aforementioned roles. Each Credential Provider can be configured to either prefer centralized mode, to prefer distributed mode, or to require distributed mode. The actual result will depend on the context: if the context does not support distributed mode, but the Credential Provider requires it, deployment will fail. Likewise, if the context mandates distributed mode, but the Credential Provider does not support it, deployment will fail. In all other cases, the preferred setting will be honored.

Table 1. SCEP Distribution Availability

<table>
<thead>
<tr>
<th>Context</th>
<th>SCEP supported</th>
<th>SCEP required</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS Profile Service</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>iOS MDM enrollment</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>iOS configuration profiles</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>SHTP enrollment</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>SHTP configuration</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Windows Phone enrollment</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Windows Phone configuration</td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>
Certificate Revocation

There are three separate aspects to a certificate’s revocation, three types of revocation: internal revocation, externally propagated revocation and externally induced revocation.

- **Internal revocation** Internal revocation affects the certificate’s status as maintained by XenMobile (in its database). This status is taken into account when XenMobile evaluates a certificate presented to it, or when it has to provide OCSP status information for some certificate). The Credential Provider configuration determines how this status is affected under various conditions. For instance, the Credential Provider may specify that certificates obtained through it should be (flagged as) revoked when they have been deleted from the device.

- **Externally propagated revocation** Also known as “Revocation from XenMobile”, this type of revocation applies to certificates obtained from an external PKI, and means that the certificate will be revoked on the PKI when it is internally revoked by XenMobile (under the conditions defined by the Credential Provider configuration). The call to perform the revocation requires a revoke-capable GPKI Entity.

- **Externally induced revocation** Also known as “Revocation from PKI”, this type of revocation also only applies to certificates obtained from an external PKI, and means that whenever XenMobile evaluates a given certificate’s status, it will query the PKI as to that status, and, if the PKI returns that the certificate is revoked, will internally revoke it. This mechanism uses the OCSP protocol.

These three types are not exclusive, but rather apply together: the internal revocation is caused either by an external revocation or by independent findings, and in turn it potentially effects an external revocation.
Certificate Renewal

A certificate renewal is the combination of a revocation (of the existing certificate) and an issuance (of another certificate).

Note that XenMobile will first attempt to obtain the new certificate before revoking the previous one, in order to avoid discontinuation of service if the issuance fails. If distributed (SCEP-supported) delivery is used, the revocation will also only happen once the certificate has been successfully installed on the device; otherwise, the revocation will occur before the new certificate is sent to the device and independently of the success or failure of its installation.

The revocation configuration requires that you specify a certain duration (in days); when the device connects, the server verifies whether the certificate’s NotAfter date is later than the current date minus the specified duration. If it is, a renewal is attempted.
To create a credential provider using discretionary CA entities

Configuring a Credential Provider varies mostly as a factor of which issuing entity and which issuing method are selected for it. You can distinguish between Credential Provider using an internal entity, such discretionary, and those using an external entity, such as Microsoft CA or GPKI.

This task shows you how to create a discretionary entity. The issuing method for a discretionary entity is always sign, meaning that with each issuing operation, Device Manager will sign a new key pair with the CA certificate selected for the entity whether the key pair is generated on the device or on the server will depend on the selected distribution method.

1. In the Device Manager web console, click Options.

2. In the Options dialog box, select PKI > Credential Provider.

3. In the Define a new credential provider dialog box, on the General tab, enter the following information

   a. Credential Provider name. Type a unique name for the new provider configuration. This name will be used subsequently to refer to the configuration in other parts of the administration console.

   b. Description. An optional description for the configuration.

   c. Issuing method. The method that the system should obtain certificates from the configured entity. Select a discretionary entity.

4. On the CSR tab, you configure the parameters for the key pair that will be created during issuance, as well as the parameters of the new certificate. Enter the following information:

   a. Key algorithm. The key algorithm for the new key pair. Available values are RSA, DSA and ECDSA.

   b. Key size. The size, in bits, of the key pair. Note that which values are permissible depends on the type of key (for instance, the maximum size for DSA keys is 1024 bits). To avoid false negatives (which will be dependent on the underlying hardware and software), Device Manager will not enforce key sizes. You should always test Credential Provider configurations in a test environment before activating them in production.

   c. Signature algorithm. The signature algorithm for the new certificate. Values are dependent on the key algorithm; in this case, Device Manager will limit your choices to matching values.

   d. Subject name. The Distinguished Name (DN) of the new certificate’s subject. For example: CN=${user.username}, OU=${user.department}, O=${user.companyname}, C=${user.c}
To create a credential provider using discretionary CA entities

e. Subject alternative names. X.509 subject alternative names. To create a new entry, click on New alternative name and then click on the first column to select the type of alternative name from those available. Last, enter a value in the second column. Note that as for the subject DN, you can use Device Manager macros in the value field.

5. Next, click on the Distribution tab. Because the Credential Provider uses a Discretionary CA Entity, the CA certificate for the Credential Provider will always be the CA certificate configured on the entity itself; it will be presented here for mere consistency with configurations that use external entities.

The second element on this tab is the configuration of certificate delivery. If you have defined RA certificates at the entity level, they will be filled by default here, but you can change them if you desire (but that the constraints on RA certificates still apply).

You can then select the delivery mode for certificates obtained from this entity. If you select the Prefer centralized delivery mode, RA certificates are optional; otherwise, they’re mandatory.

6. Next, click on the Revocation tab. In this tab, you can configure under what conditions Device Manager should (internally) flag certificates issued through this provider configuration as revoked. You can also instruct the system to send a notification when it flags a certificate as revoked. Do this by selecting a template for the event type Certificate revoke. Device Manager will create a default template for that type, but you can edit it or create others). Note that the revocation configured here will be what determines the responses from the Device Manager OCSP Responder for certificates created with this configuration, if OCSP support is enabled for the PKI Entity.

7. Next, click on the Renewal tab. In this tab, you can configure the renewal of certificates obtained through this configuration. Two basic operations can be configured:

a. Renewing the certificate, optionally sending a notification when this is done (notification on renewal), and optionally excluding already expired certificates from the operation. Note that “already expired” in this case means that their NotAfter date is in the past; not that they already have been revoked. Device Manager will not renew certificates once they have been internally revoked.

b. Issuing a notification for certificates that near expiration notification before renewal).

To have notifications sent for either case, simply specify a Notification Template for the appropriate event type. The event type for the former is Certificate is renewed; for the latter, Certificate will expire. Device Manager will create default Notification Templates for both these event types, but you can modify them or create new ones.

Note that renewal takes precedence over notification before renewal. That is, if at a given moment Device Manager determines that a certificate must be renewed, it will not also send a notification before renewal (instead, the notification on renewal, if any configured, will be used). You should configure a greater period for the notification before renewal if you imperatively need both to be sent. Notifications before renewal will only be sent at most once for a given certificate.

8. Click Create.
To create a credential provider using external PKI entities

When you create a Credential Provider using an external (Microsoft or GPKI) entity, the main difference in configuring the two is the issuing method for the provider; which methods are available depends on the capabilities of the selected PKI entity:

- If you opt for using a Microsoft CA entity for your Credential Provider, your choice of issuing method will be limited to sign. The sign method of issuance involves creating a new key pair, creating a Certificate Signing Request (CSR) for the key pair, and submitting it to a CA for signature. (You will be prompted to select which certificate template to use for issuance). You must choose a value from those you have defined during the creation of the PKI entity. This is the template name that will be sent to the Microsoft CA along with the Certificate Signing Request during issuance. A Credential Provider can only use one template. If you want to issue certificates based on different templates, create a Credential Provider configuration for each of them.

- If you opt for using a GPKI entity for your Credential Provider, your choice of issuing method will depend on which capabilities are supported by the adapter. If the GPKI adapter defines user parameters for the selected capability, you will be presented with an interface to specify values for each of those. The parameters are specific to a capability; different capabilities have different sets of user parameters.

Note: For information on configuring Microsoft Certificate Services to work with Device Manager, see Configuring Device Manager with Microsoft Certificate Services

1. In the Device Manager web console, click Options.

2. In the Options dialog box, select PKI > Credential Provider.

3. In the Define a new credential provider dialog box, on the General tab, enter the following information
   a. Credential Provider name. Type a unique name for the new provider configuration. This name will be used subsequently to refer to the configuration in other parts of the administration console.
   b. Description. An optional description for the configuration.
   c. Issuing method. The method that the system should obtain certificates from the configured entity. Select a Microsoft of GPKI entity.

4. On the CSR tab (sign method only), you can configure the parameters for the key pair that will be created during issuance, as well as the parameters of the new certificate:
   a. Key algorithm. The key algorithm for the new key pair. Available values are RSA, DSA and ECDSA. Key size. The size, in bits, of the key pair. Note that which values are permissible depends on the type of key (for instance, the maximum size for DSA keys is 1024 bits). To avoid false negatives (which will be dependent on the underlying hardware or software), Device Manager will not enforce key sizes. You should always test Credential Provider configurations in a test environment before
b. Signature algorithm. The signature algorithm for the new certificate. Values are dependent on the key algorithm; in this case, Device Manager will limit your choices to matching values.

c. Subject name. The Distinguished Name (DN) of the new certificate’s subject. The format the system expects is as described in [5]. Note that you can use Device Manager macros for the DN field values. For example:

\[
CN={user.username},
OU={user.department},
O={user.companyname},
C={user.c}
\]

d. Subject alternative names X.509 subject alternative names. To create a new entry, click on “New alternative name”; then click on the first column to select the type of alternative name from those available; and finally enter a value in the second column. Note that as for the subject DN, you can use Device Manager macros in the value field.

5. Next, click the Distribution tab. Here, you are required to specify the issuer CA of the certificates returned by the PKI entity in the configuration you have selected. You will be offered to choose one of the CA certificates defined on the entity. If the issuing method for this Credential Provider is sign, you will also be able to configure the delivery method, since the fetch method retrieves the key pair from the PKI server and hence there is no key generation involved at all, distributed key generation is not available with that method.

6. Next, select the Revocation XenMobile tab. On this tab, you can configure the conditions and actions for the internal revocation of certificates. You can opt to have certificates revoked under the following conditions:

a. When they are removed from the device, that is, either when the system detects that they have been removed from the device without server interaction, or when the server has removed them subsequent to the scheduling of a removal command.

b. When they are updated on the device, that is, when they are replaced with a newer certificate for the same function.

c. When the enrollment is revoked by the administrator.

d. When the device is deleted.

e. Or any combination of these.

You can further opt to have a notification sent when the revocation action is undertaken; to do so, simply configure a notification template for the appropriate event type (Certificate revoke).

In addition to these conditions, since the certificates obtained through this configuration will have come from an external source, you can opt to propagate the revocation status externally (the common case would be to propagate it to the PKI that issued the certificates, but your choice is not restricted in that matter). The propagation is achieved using a GPKI entity with the revoke capability; the interface will propose you the list of revoke-capable GPKI entities that exist in the system. If the selected entity defines user-parameters for the revoke operation, you will be prompted to enter values for them. You can use Device Manager macros for the values.

7. Next, select the Revocation PKI tab.
In this tab, you can configure the system to perform external certificate status checks for certificates issued through this CredentialProvider configuration. The checks are performed using the OCSP protocol [1] and take place when a deployment is initiated. For the checks to occur, the back-end PKI must insert corresponding OCSP responder address extensions (ASN.1 OID: 1.3.6.1.5.5.7.1.1) in the certificates it issues. If that is not the case, the setting will be silently ignored.

As part of the OCSP protocol, the initiator of the OCSP request (in this case, XenMobile) must be able to validate the OCSP responder’s (likely your PKI server) signing certificate. To that effect, as part of the external revocation check configuration, you must specify the CA certificate of your PKI’s OCSP Responder’s signing certificate. The CA certificate must be uploaded to the Server Certificates repository so that you can select it in the drop down. Its private key is not required for this purpose.

Note that OCSP Responder certificates are usually either the CA certificate itself (that is, the CA that signed the certificate the status of which is queried), or a certificate signed directly by that CA. It that sense, specifying that CA certificate in this section will usually be adequate.

You can further define what actions XenMobile should undertake in the event that the OCSP verification yields a status indicating that the certificate in question was revoked. If that is the case, you can opt to:

- Do nothing.
- Remove the corresponding configuration from the device, that is, the configuration the certificate in question was deployed as part of.
- Revoke the enrollment and wipe the device.

In addition to the action you opt for, you can choose to have a notification sent in that case, by selecting a notification template for the appropriate event type (Certificate revoked by PKI). The external revocation and the internal revocation configured in the tab before are complementary, in the sense that if the external revocation check yields a revoked status and you have opted, for instance, to revoke the entire enrollment in that case, then the settings you have specified in the Revocation XenMobile tab will apply to all other certificates present on the device. The same thing goes for all certificates that were part of the same configuration if you have merely chosen to remove the configuration the certificate was deployed as a part of.

8. Next, select the Renewal tab.

On this tab, you can configure the renewal of certificates obtained through this configuration. Two basic operations can be configured:

- Renewing the certificate, optionally sending a notification when this is done (notification on renewal), and optionally excluding already expired certificates from the operation. Note that ‘already expired’ in this case means that their NotAfter date is in the past; not that they already have been revoked. XenMobile will not renew certificates once they have been internally revoked.

- Issuing a notification for certificates that near expiration (notification before renewal).

To have notifications sent for either case, simply specify a Notification Template for the appropriate event type. The event type for the former is Certificate is renewed; for the latter, Certificate will expire. XenMobile will create default Notification Templates for both these event types, but you can modify them or create new ones.
To create a credential provider using external PKI entities

It is important to note that renewal takes precedence over notification before renewal. That is, if at a given moment XenMobile determines that a certificate must be renewed, it will not also send a notification before renewal (instead, the notification on renewal, if any configured, will be used). You should configure a greater period for the notification before renewal if you imperatively need both to be sent. Notifications before renewal will only be sent at most once for a given certificate.

9. Click Create.
Configuring a SAML Service Provider

Device Manager supports configuration of your own Security Assertion Markup Language (SAML) service and identity provider and SAML-based infrastructure to authenticate users and their mobile devices. With your own SAML configuration, you do not need to pre-provision user account information in Device Manager, such as user names, group association, or other directory attributes. SAML implementations allow network administrators to provide single sign on access to servers, web sites, and apps.

SAML Use Cases

Initial Registration of Mobile Device

The Device Manager agent should be able to register the device with the Device Manager server using the SAML token. No pre-provisioning of the user name, group association, or other directory attributes in the Device Manager server should be required.

Ongoing Authentication and Authorization for Policy Updates and Device Controls

Once the mobile device manager (MDM) agent registers the device and receives the initial policy updates, the mobile device must be able to re-authenticate with the IDP server each time the SAML token expires to receive policy updates and allow for security actions, such as lock, revoke, wipe devices, and so on, including know when the user has changed groups that might impact MDM policies, or proper authorization.

Single Sign-On With Other SAML-Enabled Applications

After the mobile device registers with the SAML token, other SAML-enabled applications should be able to authenticate the user without prompting for the corporate credential to provide a single sign-on user experience. It needs to be determined whether all SAML-enabled applications, including popular ones, such as SF.com, Google Apps, Microsoft365, Box.net, and so on can be supported or only applications that are managed by Device Manager or written to the App SDK.

Decommissioning Devices and Removing Users

When a user is removed from the corporate directory; for example, the user leaves the organization, there must be a mechanism to deactivate users and decommission the devices in the Device Manager server.
SAML Test Requirements

- Establish a "relying party trust" between the iDP server and the Device Manager Service Provider server, including required certificates for the trust relationship.

- Develop claim attribute mapping with User ID, Group Membership, Email Address, and other directory attributes.

- Device Manager agent requests the SAML token from the customer iDP server and redirects back to Device Manager server for mobile device registration.

- SAML token on mobile device is presented to Device Manager server for device registration; Device Manager server validates the SAML token and extracts directory attributes; the device is registered and the user identity is created properly.

- Device configuration appears as expected in the Device Manager console; for example, as the software inventory.

- All reports list devices and inventory properly.

- Lock and revoke device using the Device Manager console security commands.

- Change the users group association from Group A to Group B. Push out different Device Manager policy updates to the devices for Group A and Group B. Verify that device gets the proper (Group B) policy updates.

- Access other SAML-enabled applications using HTML-based mobile apps to determine if user is prompted for corporate directory credentials to issue a separate SAML token.

- Access other SAML-enabled application using native mobile apps to determine if user is prompted for corporate directory credentials to issue a separate SAML token.

- Remove user from directory, ensure device state is changed to inactive, and user is removed automatically.

- User is able to reactivate by re-registering the device using the same SAML-based process for initial registration.
To add a SAML service provider

1. Click Edit.

2. In the Service Provider Configuration dialog box, click the General tab and then enter the following information:
   - Entity ID. Enter the ID of the SAML Service Providers Entity ID (globally unique name given to a SAML entity). An entity ID is typically rooted in the organization’s Primary DNS Domain.
   - Base URL. URL of the SAML Service Provider.
   - Organization name. The name of your company (optional).
   - Organization Description. Description of your company (optional).
   - Organization URL. The URL of your company (optional).

3. On the Main Parameters tab, select the following options:
   - Supported Bindings
     - SAML Redirect. Select if your SAML server has implemented a URL redirect binding.
     - General
       - Sent SAML Request must be signed.
       - Reserved Assertion must be signed.
       - Received assertion must be encrypted.

4. On the Contacts tab, you can enter the email addresses for the technical, support, and administrative contacts in your organization.

5. On the Certificates tab, you can enter upload a certificate for the SAML connection, as well as the Keystore password for SAML server authentication.

6. Click Save.
To configure a SAML identity provider

1. In the Identity Providers tab, click New.

2. In the General dialog box for the SAML Identity Provider, enter the following information:
   - Metadata URL. Web address used to access the SAML service provider metadata.
   - User domain. Domain under which the SAML metadata URL resides.

3. Click Create.
Configuring General Security Options

You can configure security in the Options dialog box to customize the security features of the service. By default, when Secure Device is included in the license, it is automatically activated during installation, with a strong level of security. If you need to change those parameters, use that dialog box.

- Enforce SSLForces devices to communicate using an SSL transport. All HTTP requests from devices will be rejected.
- Strong Authentication. Enabling Strong Authentication generates a Strong ID for devices that is then used as a second factor of authentication during the enrollment process.
- Strong ID Valid Once. Allows Strong ID passcodes to only be used once. When the Strong ID is used once to generate a device certificate, it cannot be reused. Device has to be revoked and re-authorized.
- Certificate Renewal. Sets the renewal time frame for certificates used in Strong Authentication mode. '0' disables the certificate renewal process.
- Always Add Device. Allow to automatically register devices into Device Manager even when Secure Device is activated.
- Block Rooted Android and iOS Enrollment. Enabling this function will block rooted or jailbroken devices from enrolling.
- 8 Char Strong ID. Enables a Strong ID character string that is limited to 8 characters.
- SHP Console for Users. Enables or Disables the Self-Help Console for user management of devices.
- XDM/SHP console max inactive interval. The time (in minutes) between client requests before the server will invalidate a session. 0 means that a session will never timeout.
- iOS agent auto logout (minutes). Length of time before an iOS agent user is logged due to inactivity.
- Enable client cert authentication for iOS. If enabled, iOS enrollment agent will use certificate authentication. If disabled, iOS enrollment agent uses session based authentication.

To enable Strong ID

Strong ID is a form of 2 factor authentication used to provide an extra layer of extra security when enrolling a device.

1. Enable Strong ID from the Options menu on the Security tab in the Device Manager web console. Citrix also recommends that you enable 8 Char Strong ID). At this point, no devices will be able to enroll until the device's serial number or IMEI is known.
2. Add the devices manually (or import) from the Devices tab using the device's serial number or IMEI, which will generate a Strong ID for the device.

3. When a user is ready to enroll, the user needs to call their administrator and give their Serial/IMEI, so the administrator can provide the Strong ID from the device properties.
Configuring Network Access Controls

If you have a Network Access Control (NAC) appliance set up in your network (such as a Cisco ISE), you can enable filters to set devices as compliant or not compliant for NAC-based on rules or properties. If a Device Manager managed device does not meet the specified criteria, and thus is marked Not Compliant, the device will be blocked on your network by the NAC appliance.

To set unmanaged devices as not compliant, enable the associated filter and set to "Not Compliant". The "Implicit Compliant / Not Compliant" filter sets the default value only on devices that are managed by XenMobile. For example, any devices that have a blacklisted app installed and/or are anonymous (not enrolled) are marked as Not-Compliant and will be blocked from your network by the NAC appliance.

The NAC compliance filters are as follows:

- Blacklisted Apps. Device has a blacklisted app installed.
- Rooted Android/Jailbroken iOS Devices
- Revoked Status. Device has been revoked.
- Unmanaged Devices. Device is in an unmanaged state.
- Suggested Apps Only. Device has "suggested" app installed.
- Inactive Devices. Device is in an inactive state.
- Anonymous Devices. Device is anonymous.
- Out of Compliance Devices. Device has property of Out of Compliance set to True.
- Encryption. The device has disk encryption enabled.
- Implicit Compliant/Non-Compliant. Indicates that if none of the above filters match, return device to be compliant or not (according to the option selected).
Configuring Device Manager with Microsoft Certificate Services

You can configure Device Manager with Microsoft Certificate Services to generate user certificates for certificate-based authentication with WIFI, VPN, and Exchange ActiveSync profiles. You can also configure Device Manager as a Registration Authority to generate requests and to issue device identity certificates with Microsoft Certificate Services.

In addition, you can configure Device Manager to use external SSL server certificates and digital signature certificates from other PKI-trusted certificate authorities.

**Caution:** Changing the digital signature certificate or the SSL certificate authority will disable the management of currently enrolled devices and require a re-enrollment across all devices.

Device Manager can make certificate requests to Microsoft Certificate Services through web enrollment to enable certificate-based authentication for WIFI, VPN, and Exchange ActiveSync profiles. Device Manager does this by acting as a client to Microsoft Certificate Services and requesting certificates on behalf of users with enrolled devices. This section describes how to create a Microsoft Certificate Server entity and configure Device Manager to request certificates for users enabling certificate-based authentication.

**Prerequisites**


- Port 443 (default) open from Device Manager to Microsoft Certificate Services server.

- Microsoft KB 980436 patch needs to be installed on Microsoft Certificate Services server.

- Microsoft KB 272175 - Guidelines for configuring client certificate authentication mode for IIS 6.

- Microsoft KB 953461 patch needs to be installed on Microsoft Certificate Services server on Windows 2008 Server Enterprise.

- Web enrollment for Microsoft Certificate Services needs to be enabled.

- SSL enabled on Microsoft Internet Information Services (IIS).

- IIS configured to accept client certificate authentication.

- The client certificate in .p12 format which is used to authenticate against Microsoft Certificate Services should be copied to the Device Manager server and made accessible.
To enable Web enrollment for Microsoft Certificate Services

1. In Administrative Tools, click Server Manager.

2. Under Active Directory Certificate Services, check to see if Certificate Authority Web Enrollment is installed.

3. Select Add Role Services to install Certificate Authority Web Enrollment, if needed.

4. Select Certificate Authority Web Enrollment and then click Next.

5. Click Close or Finish when the installation is complete.
To enable IIS Web services

1. Go to Administrative Tools and click Server Manager.
2. Select Server Roles on the left side.
3. Select the Active Directory Certificate Services role and the Web Server IIS role, and click Install.
4. Close the Server Manager.
To configure Microsoft Internet Information Services for self-signed or external certificates

1. Go to Administrative Tools and click Server Manager.

2. Under Web Server (IIS), under Internet Information Services (IIS), select the host or top of the root and then click Server Certificates.

3. Create a self-signed certificate or import an external certificate.
To configure Microsoft Internet Information Services

1. In Administrative Tools, select Server Manager.

2. Under Web Server (IIS), under Role Services, verify that Client Certificate Mapping Authentication and IIS Client Certificate Mapping Authentication are installed. If not, install these role services.

3. In Administrative Tools, click Internet Information Services (IIS) Manager.

4. In the left-hand pane of the IIS Manager window, select the server running the IIS instance for web enrollment and then click Authentication.


6. Click Sites and then in the right pane, click Bindings.

7. Add an HTTPS binding if one does not exist.

8. Go to Web Server (IIS) > Sites > Default Web Site > CertSrv

9. Click SSL Settings and then click Accept for Client Certificates.
To create a certificate template for XenMobile certificate requests

1. Open the an MMC Console with a domain administrator account and then add a Snap-In for Certificate Templates.

2. Open Certificate Templates.

3. Right-click the User template and then click Duplicate Template.

4. Select Windows 2003 Server for the template type and then click OK.

5. In Template Display Name, enter a certificate. Note the actual Template Name because you will need it later in the configuration.


7. Click the Request Handling tab and then specify Signature and Encryption.

8. Enable or disable Allow private key to be exported.

9. Select Enroll subject without requiring any user input.

10. Select Supply in the request.

11. Click OK on the warning window.

12. Click the Security tab.

13. Grant Enroll permissions to a user account that will be making the certificate requests from Device Manager.

14. Open MMC and add a Snap-In for Certification Authority. Expand the CA server and right-click Certificate Templates.

15. Make sure that User template as shown in the screenshot below exists within Certificate Templates. Make sure that User template exists, otherwise the server will be unable to issue a user certificate.

16. Click New and then click Certificate Template to Issue. Select the certificate template you created in the preceding steps.
To generate the XenMobile client certificate

You can request certificate from any system in the domain; however, make sure to logon using domain service account credentials. The domain account must have local administrator rights to the system requesting a certificate from the Certificate Server.

1. Either Run As a Domain User or initiate a Remote Desktop session to a system using Domain User credentials.

2. Open a web browser and open the web enrollment page for Microsoft Certificate Services. This page is usually https://server.company.com/certsrv (certsrv is case-sensitive).

3. Click Request a Certificate.

4. Click User Certificate and the click Submit.

5. Click Install the Certificate.
To export the client certificate

The client certificate that you request must be exported as a .p12 or PKCS12 certificate and copied to the Device Manager server.

1. Export the certificate as a .p12 or PKCS12 certificate from the web browser used or from the Certificates console on the CA server.

2. Open an MMC Console and add the Certificates Snap-in.

3. Right-click the certificate that you requested and then click All Tasks and Export.

4. In the Certificate Export window, click Next.

5. Click Yes to export the private key.

6. Enter a password for the exported certificate. You will need to remember this password.

7. Enter a file name for the certificate export and then click Next.

   **Note:** The file name cannot contain spaces.

8. Click Finish.

9. Copy the filename.pfx or filename.p12 to the Device Manager server and specify a location.
To configure a Microsoft certificate server entity

1. In the Device Manager web console, click Options.

2. In the Options dialog box, from the left side select PKI > Entities.

3. Click New > New MsCertSrv entity.

4. In the Add a MsCertSrv entity dialog box, on the General tab enter the following information:
   a. Entity name. Type a name for your new entity, which you’ll use later on to refer to that entity. Entity names must be unique.
   b. Service root URL. The base URL of your Microsoft CA’s web enrollment service; for example, https://192.168.2.113/certsrv/ (the URL may use plain HTTP or HTTP-over-SSL).
   c. certnew.cer page name. The name of the certnew.cer page, if you have renamed it for some reason. If not, then you can leave this field empty.
   d. certfnsh.asp page name. The name of the certfnsh.asp page, if you have renamed it for some reason. If not, leave this field empty.
   e. Authentication type. Select No authentication, HTTP-Basic Authentication or SSL client certificate authentication. For the latter, you will have to upload the SSL client certificate to the repository (with its private key) and select it here.

5. Next, select the Templates tab. On this tab, you will need to list the Certificate templates for your Microsoft CA. Note that those must be the internal names, not the display names.

6. Next, select the Custom HTTP parameters tab. On this tab, you can specify custom parameters that XenMobile should inject in the HTTP request to the Microsoft Web Enrollment interface. This will only be useful if you have customized scripts running on the CA.

7. Next, select the CA Certificates tab. On this tab, you will be required to inform XenMobile of the signers of the certificates the system will obtain through this entity. When your CA certificate is renewed, all you need to do is update it in the repository and then the change will be effected to the entity transparently.

8. Click Create.
To configure a Microsoft certificate services policy

Before you can configure a Microsoft certificate services policy, you need to configure a Microsoft CA credential provider in the Device Manager Options dialog box. Once the Microsoft CA credential provider has been configured, then you can create the policy that references the provider. For instructions, see To create a credential provider using external PKI entities.

1. Click the Policies tab in the Device Manager console.

2. On the left-hand pane, under iOS, click Configuration profiles.

3. Click New Configuration > Profiles and Settings > Credentials.

4. In the Credential configuration creation dialog box, on the General tab, enter the following information:

   a. Identifier. Type a name for the profile that identifies it uniquely to the user. This name must be unique and not in use by any other profile, or if this name matches the name of another policy, the first policy will be overwritten.

   b. Display name. Type a name of the profile as it will appear in the Device Manager web console.

   c. Organization. Type your company or organization name.

   d. Description. Type an optional description to describe the policy.

   e. In the Allow Profile Removal section, choose one of the following:

      - Always. Allows the profile to always be removable.

      - Authentication. Allows you to enter a required password that is used when profile is removed. Requires a password.

      - Never. Prevents the profile from ever being removed.

   f. Select the Automatic Removal Date check box if you want to select a specific date on which to remove the profile.

   g. Select the Duration until removal (in days) check box to specify a set a period of time after which the profile will automatically be removed.

5. Next, select the Credential tab, and configure the following settings:

   a. Credential Type. Select Credential Provider.

   b. Credential Provider. Select the Microsoft CA credential provider you previously configured in the Device Manager Options dialog box.

6. Click Create.
To configure a Microsoft certificate services policy

This policy can now be deployed to iOS devices. For information, see Creating Deployment Packages
Configuring an OpenTrust PKI Adapter for Device Manager

XenMobile OpenTrust Adapter was validated with OpenTrust PKI Version 4.7.1 (r131349).

The XenMobile OpenTrust Adapter is a web application running on Tomcat:

- Windows 2008 R2
- Java 1.6.0_29 or above, 32 bits version
- Apache Tomcat 7.0.27

**Note:** You only need Tomcat core features, not the manager or the documentation, unless you need it. After installation, you can also delete the directory `<tomcat_dir>/webapps/ROOT`.

The XenMobile OpenTrust Adapter provides an interface that allows Device Manager to submit certificate requests for a signature to an OpenTrust Certificate Manager server. Device Manager submits a request to the OpenTrust adapter to sign a certificate. The OpenTrust Certificate Manager receives the request, signs the certificate and returns it to Device Manager. Device Manager makes these certificate requests in order to generate device identity for mobile device management mutual authentication, or user credential certificates to be used in conjunction with WiFi, VPN, and Exchange ActiveSync profiles for iOS devices. XenMobile recommends that the OpenTrust Adapter is installed on a separate server from the Device Manager host, using its own instance of Tomcat 7.0.

**To install OpenTrust Adapter**

1. Copy the provided WAR file to the Tomcat webapps directory. You can change the WAR file name to fit the usage of this adapter instance (wifi_certificate, exchange_certificate, and so on).

2. Start Tomcat. It will automatically expand and install the web application in its directory.

3. To check that the adapter is properly running, connect to:
To obtain an authentication certificate from OpenTrust PKI

The authentication between the OpenTrust Adapter is secured by using a client certificate that needs to be generated from OpenTrust PKI server.

1. Log in to the OpenTrust PKI server, browse to Enrollment Entity and then click Request a Certificate.
2. Select Other and then click Next.
3. Select Authentication and then click Next.
4. Enter the required parameters and then click Next.
5. You now need to validate the certificate request. Navigate to Registration Authority > Enrollment > List Certificate Requests.
6. Select your certificate request and then click Process selected requests.
7. Click Approve.
8. You now need to retrieve the certificate. Navigate to Enrollment Entity > Search for a Certificate > Enrollment.
9. Enter your search criteria and then click Search.
10. Find your certificate and then click the name.
11. Click Integrate this certificate into your browser (or smartcard).
12. Open the certificate store of your web browser. For example, with Firefox, navigate to Options, click the Encryption tab and then click View Certificates.
13. In the Certificate Manager, click the Your Certificates tab.
14. Select your certificate and then click Backup.
15. Enter the password and save the resulting p12 file. You will need the file and password when you configure the adapter.
To set up access rights on OpenTrust PKI

You need to provide the required access rights to the generated identity.

1. Navigate to Access Control.

2. Select your User.

3. If you already have a group defined to allow SOAP access to the Registration Authority, you can add this user to this group. Select the group and then click Save.

4. To give individual rights to that user, click the Rights on Modules tab.

5. Select the Execute check box to give access rights to the Registration Authority.

6. Click the Rights on Zones & Profiles tab.

7. For each profile you want the user to be able to control, next to Enrollment, select the Execute check box.

8. Click Save.
To configure the OpenTrust adapter

1. Open the file `opentrust_adapter.properties` in `tomcat/webapps/<adapter_name>/WEB-INF/classes` and edit it accordingly:

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OpenTrust.RA.Url</strong></td>
<td>Web address used to access the SOAP interface of the OpenTrust PKI server</td>
</tr>
<tr>
<td><strong>Enrollment.Profile</strong></td>
<td>OpenTrust Profile name used by this instance</td>
</tr>
<tr>
<td><strong>KeyPair.FileName</strong></td>
<td>Path to the keypair used to authenticate to OpenTrust PKI SOAP interface</td>
</tr>
<tr>
<td><strong>KeyPair.Psw</strong></td>
<td>Password of the above mentioned keypair</td>
</tr>
</tbody>
</table>
To set the connection to the adapter

1. To configure Device Manager with your adapter, on the Options menu, click PKI Entity.

2. Click New and then enter the required information:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity Name</td>
<td>Name your adapter connection.</td>
</tr>
<tr>
<td>URL</td>
<td>Enter the URL of the adapter web services interface: http://&lt;server&gt;:&lt;port&gt;/&lt;adapter_name&gt;/GpkiAdapter?wsdl</td>
</tr>
<tr>
<td>Certificate path</td>
<td>If you are using an authenticated HTTPS connection, select your client cert (p12).</td>
</tr>
<tr>
<td>Certificate password</td>
<td>Enter the password for the above p12.</td>
</tr>
</tbody>
</table>

3. Click Load to initiate the connection with the adapter.

4. Click Ping to check the connectivity.

5. Click Create to save the adapter configuration.
To configure an iOS profile to deliver certificates to iOS devices

To deliver certificates to iOS devices, you need to configure an iOS profile in Device Manager. For more information on configuring PKI integration with Device Manager, see About XenMobile PKI.

1. Click the Policies tab.

2. On the left side, under iOS, click Configurations.

3. Create a new policy for the PKI authority that you installed by clicking New Configuration > Profiles and Settings > Credentials.

4. On the General tab, enter the following information:
   a. Identifier. Enter a unique identifier to distinguish the certificate policy.
   b. Display name. Enter a name that will be used to label the policy on the device.
   c. Organization. Enter your company name here.
   d. Descriptions. Type an optional description.

5. In Allow profile removal operation, click one of the following options:
   - Always: This option allows the profile to always be removable.
   - Authentication: Allows you to enter a required password that is used when profile is removed. Requires a password
   - Never: Prevents the profile from ever being removed.

6. Select the Allows you to select a specific date check box to specify a date you want to remove the profile.

7. Select the Duration until removal (in days) check box to enable you to set a period of time after which the profile will automatically be removed.

8. Next, on the Credential tab, enter the following information:
   a. Credential name. Provide a unique name for the credential.
   b. Description. Optionally, you can type a description for the credential.
   c. Credential Type. Select a credential type according to the PKI configuration you have set up for Device Manager, such as a certificate, a keystore, a server certificate, or a credential provider.
   d. Credential file path, Server certificate, or Credential provider. Select the path or the name of the credential you are adding to the policy. If you are using a Keystore file, then you need to provide the keystore password.
To configure an iOS profile to deliver certificates to iOS devices

9. Click Create.
To configure an OpenTrust adapter to use HTTP by using a self-signed certificate

If you want the adapter to be accessible using HTTPS, you need to configure the Tomcat connector accordingly. You can configure the adapter by using a self-signed certificate. This process uses openssl and java keytool.

1. Create a directory called certs. In that directory, create another directory called ca.

2. Create a root CA. You need to adapt the subject name and passwords to fit your needs. In the certs directory, issue the following commands:

   openssl genrsa -aes256 -passout pass:zenprise -out ca/ca.key 1024

   openssl req -new -x509 -passin pass:zenprise -key ca/ca.key -out ca/ca.pem -days 3650 -subj "/C=US/ST=CA/L=RWC/O=Zenprise/OU=Zenprise/CN=ZenTestCA/emailAddress=none@zenprise.com"

   openssl x509 -inform PEM -in ca/ca.pem -outform DER -out ca.crt

3. Create an HTTPS certificate using that CA. Change at least the CN to fit the XenMobile OpenTrust Adapter server name. For example:

   openssl genrsa -aes256 -passout pass:zenprise -out server-key.pem 1024

   openssl req -new -passin pass:zenprise -subj "/C=US/ST=CA/L=RWC/O=Zenprise/OU=Zenprise/CN="MyServerName.zenprise.com"/emailAddress=none@zenprise.com" -days 3650 -key server-key.pem > server.csr

   openssl x509 -req -passin pass:zenprise -in server.csr -out server-crt.pem -CA ca/ca.pem -CAkey ca/ca.key -CAcreateserial -CAserial ca.srl

4. Create a p12 containing your key and certificate.

   openssl pkcs12 -export -in server-crt.pem -inkey server-key.pem -out MyServerName.p12 -name server

5. Create a java keystore containing that PKCS12 file.

   keytool -importkeystore -deststorepass changeit -destkeypass changeit -destkeystore keystore.jks -srckeystore MyServerName.p12 -srcstoretype PKCS12 -alias server

6. Modify the Tomcat server.xml file to create the HTTPS connector. The file needs to reference the keystore previously created.

   <Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"

   maxThreads="150" scheme="https" secure="true"

   clientAuth="false" sslProtocol="TLS"

   keystoreFile="C:\Zenprise\Apache Software Foundation\Tomcat 7.0\conf\keystore.jks" keystorePass="changeit"/>

7. Import the root cert in the java keystore of DeviceManager so that this server certificate can be trusted. On the Device Manager server, issue the following command:

   keytool -import -trustcacerts -alias root -file ca.crt -keystore cacerts
To configure an OpenTrust adapter to use HTTP by using a self-signed certificate

The keystore file used by Java (cacerts) is usually located in: C:\Program Files\Java\jdk1.6.0_22\jre\lib\security
To configure Device Manager to generate identity certificates from OpenTrust adapter

You will need to generate a certificate from OpenTrust with the following keyUsage:

- keyEncipherment
- digitalSignature

Furthermore, you will need an OpenTrust root certificate and a CA certificate.

Caution: This procedure will invalidate all certificates used previously by Device Manager. All devices using a certificate to authenticate, such as iOS and Android, Symbian, and Windows Mobile using Strong Authentication mode will need to be re-enrolled.

1. Modify pki.xml. This file is located in tomcat/webapps/zdm/WEB-INF/classes. Open it with a text editor, and modify it as follows (the modified parts are in bold text). Keep in mind the following considerations:

   - Path to the certificates.
   - keyUsage of the certs.
   - Name of the OpenTrust connector in the console.
   - The CSR template that has to match your profile definition on the OpenTrust PKI Server.

```xml
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:p="http://www.springframework.org/schema/p"

   <bean id="legacyRoot" class="com.sparus.nps.pki.def.PublicCertFileParams"/>

```
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<bean id="legacyIOsDevicesCa" class="com.sparus.nps.pki.def.KeyStoreParams"
    p:keyStoreType="${ios.mdm.pki.ca-mdm.keystoretype}"
    p:keyStorePath="${ios.mdm.pki.ca-mdm.certificatefile}"
    p:entryAlias=""
    p:keyStorePass="${ios.mdm.pki.ca-mdm.privatekey.password}"
    p:publiclyTrusted="false"
    p:issuerParams-ref="legacyRoot"
/>  

<!-- SHTP is the proprietary protocol ZDM uses to communicate
    with Windows and Android devices -->

<bean id="legacyShtpDevicesCa" class="com.sparus.nps.pki.def.KeyStoreParams"
    p:keyStoreType="${secure.device.keystore.type}"
    p:keyStorePath="${secure.device.certificate.file}"
    p:entryAlias="${secure.device.alias}"
    p:keyStorePass="${secure.device.private.key.password}"
    p:publiclyTrusted="false"
    p:issuerParams-ref="legacyRoot"
/>  

<alias alias="legacyDigitalSigner" name="legacyIOsDevicesCa" />

<bean id="legacySslCert" class="com.sparus.nps.pki.def.KeyStoreParams"
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<p:keystoreType="${ios.mdm.pki.ssl.keystoretype}"
    p:keystorePath="${ios.mdm.pki.ssl.certificatefile}"
    p:entryAlias=""
    p:keystorePass="${ios.mdm.pki.ssl.privatekey.password}"
    p:publiclyTrusted="false"
/>

<bean id="OT_Root_cert" class="com.sparus.nps.pki.def.PublicCertFileParams"
    p:certificateFilePath="C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\otroot.cer"
    p:publiclyTrusted="false"
/>

<bean id="OT_CA_cert" class="com.sparus.nps.pki.def.PublicCertFileParams"
    p:certificateFilePath="C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\otinter.cer"
    p:publiclyTrusted="false"
    p:issuerParams-ref="OT_Root_cert"
/>

<bean id="OT_RA_cert" class="com.sparus.nps.pki.def.KeyStoreParams"
    p:keystoreType="PKCS12"
    p:keystorePath="C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\otadmin.p12"
    p:entryAlias=""
    p:keystorePass="opentrust"
    p:issuerParams-ref="OT_CA_cert"
/>

<bean class="com.sparus.nps.pki.spi.impl.GpkiCa" id="OT_CA">
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

<property name="caCertificate">
    <description>
        This CA's certificate.
    </description>
    WARNING! In order for tomcat to accept clients presenting identities
    issued by this CA, tomcat's truststore has to be modified accordingly
    (e.g. installing in it the certificate referred to here).
</property>

<property name="entityName" value="OTAdapter">
    <description>
        This is the GPKI entity name as defined in the console.
    </description>
</property>

<property name="requestProperties">
    <description>
        If the adapter defines user parameters (i.e., non-injected parameters),
        then they can be defined here. EMC adapter currently does not define
        any parameters.
    </description>
    <bean class="com.sparus.nps.pki.gpki.util.SimpleRequestProperties">
        <constructor-arg index="0" type="java.util.Map">
            <map key-type="java.lang.String" value-type="java.lang.String">
                <!--entry key="[PARAMETER NAME]" value="[PARAMETER VALUE]" /-->...</map>
        </constructor-arg>
    </bean>
</property>
To configure Device Manager to generate identity certificates from OpenTrust adapter

<constructor-arg>
</constructor-arg>

</bean>

</property>

<property name="raEncryptionCert">
<description>
RA encryption cert. MUST be issued by the certificate referred to in property caCertificate, i.e. the CA certificate, i.e. the certificate that will sign device identities.

This cert MUST have keyUsage: keyEncipherment.

RA encryption cert may be the same one as RA signing cert.
</description>

<bean factory-bean="certFactory" factory-method="buildPrivate">
<constructor-arg ref="OT_RA_cert" />
</bean>
</property>

<property name="raSigningCert">
<description>
RA signing cert. MUST be issued by the certificate referred to in property caCertificate, i.e. the CA certificate, i.e. the certificate that will sign device identities.

This cert MUST have keyUsage: digitalSignature.

RA signing cert may be the same one as RA encryption cert.
</description>

<bean factory-bean="certFactory" factory-method="buildPrivate"/>
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<constructor-arg ref="OT_RA_cert" />

</bean>

</property>

<property name="csrTemplate">

<bean class="com.sparus.nps.pki.spi.impl.CsrMacroTemplate">

<description>

Template for the CSR.

WARNING! Macros have to be specified using '%{...}', instead of '${...}', in XML files.

</description>

<property name="dnFields">

<list>

<description>

The following are samples. Remove or add others as you like.

</description>

<bean class="com.sparus.nps.pki.def.DNFieldBean" p:oid="CN" p:value="%{user.loginname}" />

<bean class="com.sparus.nps.pki.def.DNFieldBean" p:oid="OU" p:value="aeotn" />

<bean class="com.sparus.nps.pki.def.DNFieldBean" p:oid="O" p:value="noise" />

<bean class="com.sparus.nps.pki.def.DNFieldBean" p:oid="C" p:value="DE" />

</list>

</property>

<property name="altnames">

<list>

<description>

The following are samples. Remove or add others as you like.

</description>

<bean class="com.sparus.nps.pki.def.AltNameBean" p:sanType="rfc822Name" p:value="%{user.mail}" />

</list>

</property>
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

<bean class="com.sparus.nps.pki.def.AltNameBean" p:sanType="userPrincipalName" p:value="%{user.username}@home.net" />

</list>

</property>

</bean>

</property>

</bean>

<bean id="certFactory" class="com.sparus.nps.pki.def.ZdmCertificateFactory">
<description>
The ZdmCertificateFactory builds public key certificate objects from either PublicCertFileParams, PrivateCertFileParams or KeyStoreParams; and private key certificate objects (public key + private) from PrivateCertFileParams and KeyStoreParams.

Factory method for the former is: buildPublic; for the latter: buildPrivate.

</description>

</bean>
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<bean id="serialNumberGen" class="com.sparus.nps.pki.gen.CertificateSerialNumberSequenceImpl" />

<bean id="com.everywan.security.PkiSpi.internal" class="com.sparus.nps.pki.spi.impl.PluggablePki" lazy-init="true">

  <property name="digitalSignatureRoot">
    <bean factory-bean="certFactory" factory-method="buildPublic">
      <constructor-arg ref="legacyRoot" />
    </bean>
  </property>

  <property name="digitalSigningCertificate">
    <bean factory-bean="certFactory" factory-method="buildPrivate">
      <constructor-arg ref="legacyDigitalSigner" />
    </bean>
  </property>

  <property name="sslRoot"><null /></property> <!-- We don't have the config for this... -->

  <property name="digitalSigningCertificate">
    <bean factory-bean="certFactory" factory-method="buildPrivate">
      <constructor-arg ref="legacyDigitalSigner" />
    </bean>
  </property>

  <property name="sslCertificate">
    <bean factory-bean="certFactory" factory-method="buildPrivate">
      <constructor-arg ref="legacySslCert" />
    </bean>
  </property>

  <property name="shtpCa" ref="OT_CA" />

  <property name="iosMdmCa" ref="OT_CA" />

</bean>

<bean id="com.everywan.security.PkiSpi" factory-bean="com.everywan.security.PkiSpi.factory" factory-method="getBean" />
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<bean id="com.everywan.security.PkiSpi.factory" class="com.sparus.nps.pki.def.PkiSpiFacade">
    <property name="enabled" value="${zdm.pki.enable}" />
    <property name="enabledBeanId"><idref local="com.everywan.security.PkiSpi.internal" /></property>
</bean>
</beans>
```

To add certificates to the Device Manager keystore

You now need to add the intermediate and root ca certificates to the Device Manager keystore.

1. Use the java keytool command (adapt the path to your environment): "C:\Program Files\Java\jdk1.6.0_23\jre\bin\keytool" -importcert -trustcacerts -alias "externalCA" -file "C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\mycert.cer" -keystore "C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\cacerts.pem.jks" -storepass notMeaningful

2. Restart the Device Manager service to activate the new PKI usage.
To activate logging on Device Manager for the adapter

Logs from the adapter can be found in the tomcat/logs directory of the adapter.

1. Add a new logger in the log4j configuration to ensure proper error handling and auditing. In Internet Explorer, navigate to the following URL based on your installation:
   http://<host>/<instance>/log.jsp

2. Navigate to the bottom of the table and in Add New Logger, add an entry for the com.sparus.nps.pki

3. Set the logging level to TRACE.
Configuring the XenMobile RSA Adapter

The XenMobile RSA Adapter provides an interface that allows Device Manager to submit certificate requests for a signature to an RSA Certificate Manager server. Device Manager submits a request to sign a certificate to the RSA adapter. The RSA Certificate Manager receives the request and uses the RSA Xuda Libraries to sign the certificate. The Certificate Manager returns the signed certificate to Device Manager.

Device Manager makes the certificate requests in order to generate device identity for mobile device management (MDM) mutual authentication, or to generate user credential certificates to be used in conjunction with WiFi, VPN, and Exchange ActiveSync profiles for both iOS and Android devices.

Prerequisites

Citrix recommends the following prerequisites:

- Install the RSA Adapter on its own server, separate from the server running Device Manager and that you use a 32-bit instance of Tomcat 6.0.
- Device Manager Versions 7.0, 7.1, or 8.0.1.
- JAVA SDK 1.6 or later.
XenMobile RSA Adapter Certificate Manager Requirements

To install the XenMobile RSA Adapter, the following RSA Certificate Manager configurations are required. For the proper settings, consult your RSA Certificate Manager Installation Guide.

RSA Certificate Manager Installable Elements

- RSA CA Manager version 6.8 build 519 or later
- RSA Certificate Authority Version 6.8 Build 519 or later
- No special OSI-level privileges

RSA Certificate Manager Configurable Elements

- Configuration of CRL publishing: N/A
- Configuration of OCSP responder: N/A
- Configuration of certificate publishing: N/A

Partner Product Installable Elements

- Tomcat 6.0 or later, 32 bit
- Java SDK 1.6 or later

Partner Product Configurable Elements

- CRL checking mechanism: N/A
- OCSP checking mechanism: N/A
- Trust validation: N/A
- Enrollment: N/A
- General modifications to the partner product: N/A
Installing and Configuring the XenMobile RSA Adapter

The XenMobile RSA Adapter provides a mechanism for Device Manager to sign and revoke certification against an RSA Certificate Authority Version 6.8. The RSA Adapter enables device identity for mobile device management (MDM) mutual authentication and user credential certificates for use in conjunction with WiFi, VPN, and Exchange ActiveSync profiles. You perform the following tasks to install the RSA Adapter:

1. Set the Java SDK path on the Windows-based computer where you will install the RSA Adapter.

2. Configure the correct port (80) on your Tomcat server.

3. Copy the RSA Adapter installation and configuration files into a target installation directory.

4. Edit the RSA Adapter properties file with values obtained from the RSA Certificate Authority Manager Console.

5. Copy the RSA Certificate Authority Manager .cert and .key files to the installation computer.

6. Execute the RSA Adapter installation executable to install the software.

7. Verify the installation in a browser.

You perform the following tasks to configure the RSA Adapter:

1. Create and configure a PKI entity profile in Device Manager to be able to connect to the RSA Web Services Description Language (WSDL).

2. Create an iOS profile to enable use of the Certificate Authority.

3. Add a new logger in the log4j configuration to ensure proper error handling and auditing.

4. Configure the new PKI profile so it can be deployed to an iOS device and validated.
To install the RSA Adapter on Windows Server

Make sure you have access to the zenadapter.war file that is included as part of the RSA Adapter product distribution.

1. On the Windows server where you are installing the RSA Adapter, set the path to include the JAVA SDK 1.6+. For example, \Program Files (x86)\Java\jdk1.6.0_29\bin.

2. Next, configure the Tomcat server to run on port 80, instead of the default port of 8080:
   a. Navigate to %TOMCAT_HOME%/conf directory.
   b. Edit the server.xml file as follows:
      
      · Change non-SSL to:
      
      <!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
      <Connector port="80" ... />
      
      · Change SSL to:
      
      <!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
      <Connector port="443" ... />

3. On the installation computer, create a new directory named C:\Xenmobile.

4. Unzip and copy the entire contents of the RSA Adapter zip package to the directory named C:\Xenmobile.

5. Create a passphrase file that stores a passphrase that will be used by the RSA Adapter.
   Note: Before you execute this command, make sure you are logged in as the Service Account user. The Service Account user you log in as must be the same Service Account that the Tomcat server runs as.

6. Open the Windows command prompt and change directories to the location of the C:\Xenmobile folder. For example, cd C:\Zenprise.

7. From this directory, execute the following command: java -jar WinDPHarness <passphrase filename> <passphrase>

   Note: Note the file path name used in this command <passphrase filename>, because you will need it when you edit the prop.txt file in the following step.

8. Open the C:\Xenmobile\prop.txt file in a text editor and set the following attributes in the file, for example:

   · ldapport=636
To install the RSA Adapter on Windows Server

- ldaphost= rsa1.kqe.xenmobile.com
- camd5= a2064dd584c7025f03ceb0443ca0fe9e
- keyfile=C:\xenmobile\admin.key
- certfile=C:\xenmobile\admin.cert
- protectFlag=0
- jurilD=fe109c4d64430faf6d614c08b75312b0b7e31226
- passphrasefile=c:\xenmobile\passcode.txt
- profileflag-1
- profileID=AC1E02D427C3D8
- keepldapopen=1

Note: These properties are available in your RSA Certificate Authority Manager console. Refer to your RSA Certificate Authority Manager guide for instructions on where to access these properties.

9. From the RSA Certificate Authority Manager server, copy the two RSA CA Manager files - .cert and .key - to the C:\xenmobile folder on the computer where you are installing the Zenprise RSA Adapter.

10. Copy the zenadapter.war file to the %TOMCAT_HOME%\webapps folder.

11. From your command prompt, execute the following commands:

   cd %TOMCAT_HOME%\webapps
   run jar xvf zenadapter.war
   Stop Tomcat
   Start Tomcat

12. Verify that the installation was successful. In Internet Explorer (8 or later), navigate to http://HOST WITH_ADAPTER_TOMCATINSTANCE/zenadapter.war. A page with the adapter WDSL and link should appear.
To configure the RSA Adapter in the Device Manager web console

To configure the RSA adapter in the Device Manager web console, you first configure a new PKI entity. Next, you create a new iOS profile to enable use of the Certificate Authority.

To configure a new PKI entity

1. Log on to the Device Manager web console and then click Options.

2. In the XenMobile Server Options dialog box, under PKI, click Entities.

3. In the PKI entities configuration screen, click New and then click New generic PKI entity.

4. Enter a name and then enter a URL for the WSDL that you installed when you finished the RSA Adapter installation. For example: http://zdm.zenprise.com/gpki/sample.

5. If the adapter is available over HTTPS/SSL, upload the SSL client certificate. If you are not using SSL, skip to the next step.

6. Click Load.

7. Test the connection to the adapter. Click the Capabilities tab and then click Ping. A "Ping Successful" message should appear.
To create a new iOS profile

1. Click the Policies tab, click to expand iOS and then click Configurations.

2. Create a new policy for the PKI authority that you installed by clicking New Profile.

3. On the same server running Device Manager, add a new logger in the log4j configuration to ensure proper error handling and auditing. In Internet Explorer, navigate to the following Web address based on your installation: http://<host>/<instance>/log.jsp

4. Navigate to the bottom of the table and add a new logger entry for the com.sparus.nps.pki.

5. Set the logging level to TRACE.

6. Test the deployment profile on a new iOS device by moving the new PKI package into the Resources to Deploy section so you can deploy the package to an iOS device.

7. Register a new device that is targeted with the package and verify that you see the new certificate on the iOS device. If the package does not deploy, check the log file and then contact IT support.
Configuring an MDM Server Trust Certificate on Device Manager

Before you configure an external Certificate Authority (CA) by using SSL, the following files should be on the Device Manager server and accessible by the Device Manager server:

- An external SSL certificate file in .p12 format issued by a trusted CA that includes the root and intermediate. The file name, externalSsl.p12, is used as an example in this procedure.

- A password for the .p12 certificate file should be known by the installing party.

You need to configure two XML files: The pki.xml file located in the \..\tomcat\webapps\zdm\WEB-INF\classes directory and the server.xml file located in the \..\tomcat\conf directory.

1. Locate the pki.xml file in \..\tomcat\webapps\zdm\WEB-INF\classes.

2. To configure the external SSL certificate, add an "externalSslCert" bean to the file as shown in the following example. Modify the bold fields appropriately. The keyStorePath should reference the .p12 certificate file located on the server. The keyStorePass should contain the password for the .p12 file.

   ```xml
   <bean id="externalSslCert" class="com.sparus.nps.pki.def.KeyStoreParams"
      p:keyStoreType="PKCS12"
      p:keyStorePath="C:\ExternalSSL_Cert\qamdm01\externalSsl.p12"
      p:entryAlias=""
      p:keyStorePass="xxxxxxx"
      p:publiclyTrusted="true"
   />
   ``

3. Set externalSslCert as the sslCertificate property. Replace the highlighted line with the proper bean name you specified in the preceding Step.

4. Locate the server.xml file located in the \..\tomcat\conf directory.

5. Locate the Connector port=“443” and modify the following two parameters for this connector to bind the external SSL certificate to this port. The keystoreFile should point to the .p12 certificate file located on the server. The keystorePass parameter should contain the password for the .p12 file.

   ```xml
   <Connector port="443" maxHttpHeaderSize="8192" maxThreads="400" minSpareThreads="5" maxSpareThreads=""..."
   keystoreFile="C:\ExternalSSL_and_Signing_Cert\ExternalSsl.p12"
   keystorePass="xxxxxxxx"
   truststoreFile="C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\conf\cacerts.pem.jks"
   ```
6. Locate the Connector port="8443" as shown in the following example. The keystoreFile should point to the .p12 certificate file located on the server. The keystorePass parameter should contain the password for the .p12 file.

    <Connector port="8443" maxHttpHeaderSize="8192" maxThreads="20" minSpareThreads="5" maxSpareThreads="5" minSEngland="50" maxSEngland="100" scheme="https" secure="true" clientAuth="false" SSLEnabled="true" keystoreFile="C:\ExternalSSL_and_Signing_Cert\ExternalSsl.p12" keystorePass="xxxxxxxx" truststoreFile="C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\conf\cacerts.pem.jks" truststoreType="JKS" truststorePass="not Meaningful" keystoreType="PKCS12"

7. Restart the Device Manager server.
Configuring a Certificate Services Entity by Using XML

1. Find the sample entity in the file. It will be located in a bean similar to the bold text in the following example.

   ```xml
   <list value-type="com.sparus.nps.pki.conf.MsCertSrvEntity">
     <bean class="com.sparus.nps.pki.conf.MsCertSrvEntityBean">
       <constructor-arg index="0" type="java.lang.String" value="QA-CertSrv-On-SCEP-ClientCertAuth" /> ....
     </bean>
   </list>
   ```

2. Rename the entity to a name for your organization (for example, Company-MS-CA). This name appears in Device Manager.

3. Uncomment the bean for the entity by deleting the comment characters before and after the bean. For example, `<-- including the exclamation and the ending comment -->.`

4. Specify the serverBaseUrl value in the file which should be the certificate server URL to make a certificate request (for example, https://cert-server.company.com/certsrv).

   ```xml
   <property name="serverBaseUrl" value="https://serverca.company.com/certsrv/" />
   ```

5. certFinishPageName/certNewPageName: The default values may be used unless the Microsoft Certificate Server is configured to use non-default pages.

6. Specify the Client Certificate Authentication file, which is the certificate export file that you copied when you exported the client certificate. Modify the following values shown in bold.

   ```xml
   <property name="authentication">
     <bean class="com.sparus.nps.pki.conf.ClientCertAuthentication">
       <property name="keyStoreFile" value="C:\client-certificate-name.pfx" />
       <property name="keyStoreType" value="PKCS12" />
       <property name="keyStorePass" value="xxxxxxx" />
     </bean>
   </property>
   ```
7. Specify a template name to be used for making user certificate requests from Device Manager. The name should match the certificate template name you created for certificate requests. For details, see To create a certificate template for XenMobile certificate requests. Do not use the template display names. For example, “iPhone Encryption” is the display name whereas “iPhoneEncryption” is the template name. Use the template name without any spaces. The properties dialog box of the template should include both display name and the actual template name.

    <property name="availableTemplates">
    <list value-type="java.lang.String">
    <value>TemplateName</value>
    </list>
    </property>

8. Save the file and then restart the Device Manager server.
You can use XenMobile Device Manager to manage iOS, Android, Windows 8 and Windows Phone 8, Windows Mobile, and Symbian mobile devices. With the Device Manager web console, you can do the following:

- Import users from your Active Directory user database.
- Enroll users and their devices with multi-factor security.
- Create and deploy policies.
- Define and enforce compliance standards.
- View reports.
- Configure access controls.
- Set application blacklist and whitelists.
- Configure an email server.
- Locate devices.
- Remotely wipe lost or stolen devices.
- Configure advanced Public Key Infrastructure (PKI) certificates or SAML authentication.

In This Section

This section of eDocs introduces Device Manager 8.0.1 and discusses how to configure Device Manager.

<table>
<thead>
<tr>
<th>About This Release</th>
<th>Contains information about this release, including Device Manager features, components, what’s new, and known issues.</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Requirements</td>
<td>Provides system requirements for a Device Manager server and Device Manager web console, and provides device platform support information.</td>
</tr>
<tr>
<td>Plan</td>
<td>Provides information on evaluating and planning your installation of Device Manager by using the Device Manager Pre-Installation Checklist.</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Deploy</td>
<td>Provides deployment information for Device Manager, including Device Manager server placement in your company internal network and DMZ.</td>
</tr>
<tr>
<td>Install and Setup</td>
<td>Provides information about how to install Device Manager server using the installation wizard, including the Device Manager (PostgreSQL) database.</td>
</tr>
<tr>
<td>Licensing</td>
<td>Describes how licensing works on Device Manager.</td>
</tr>
<tr>
<td>Manage</td>
<td>Describes various tasks and procedures for managing devices by using the Device Manager web console to build and deploy policies.</td>
</tr>
<tr>
<td>Connect Users</td>
<td>Describes tasks related to enrolling users and devices into the Device Manager system.</td>
</tr>
<tr>
<td>Maintain</td>
<td>Provides information on maintaining the Device Manager server.</td>
</tr>
<tr>
<td>Monitor</td>
<td>Provides information on monitoring Device Manager, including reporting, using the Dashboard, server logging, and more.</td>
</tr>
</tbody>
</table>
Device Manager 8.0.1

You can use XenMobile Device Manager to manage iOS, Android, Windows 8 and Windows Phone 8, Windows Mobile, and Symbian mobile devices. With the Device Manager web console, you can do the following:

- Import users from your Active Directory user database.
- Enroll users and their devices with multi-factor security.
- Create and deploy policies.
- Define and enforce compliance standards.
- View reports.
- Configure access controls.
- Set application blacklist and whitelists.
- Configure an email server.
- Locate devices.
- Remotely wipe lost or stolen devices.
- Configure advanced Public Key Infrastructure (PKI) certificates or SAML authentication.

In This Section

This section of eDocs introduces Device Manager 8.0.1 and discusses how to configure Device Manager.

<table>
<thead>
<tr>
<th>About This Release</th>
<th>Contains information about this release, including Device Manager features, components, what’s new, and known issues.</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Requirements</td>
<td>Provides system requirements for a Device Manager server and Device Manager web console, and provides device platform support information.</td>
</tr>
<tr>
<td>Plan</td>
<td>Provides information on evaluating and planning your installation of Device Manager by using the Device Manager Pre-Installation Checklist.</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Deploy</td>
<td>Provides deployment information for Device Manager, including Device Manager server placement in your company internal network and DMZ.</td>
</tr>
<tr>
<td>Install and Setup</td>
<td>Provides information about how to install Device Manager server using the installation wizard, including the Device Manager (PostgreSQL) database.</td>
</tr>
<tr>
<td>Licensing</td>
<td>Describes how licensing works on Device Manager.</td>
</tr>
<tr>
<td>Manage</td>
<td>Describes various tasks and procedures for managing devices by using the Device Manager web console to build and deploy policies.</td>
</tr>
<tr>
<td>Connect Users</td>
<td>Describes tasks related to enrolling users and devices into the Device Manager system.</td>
</tr>
<tr>
<td>Maintain</td>
<td>Provides information on maintaining the Device Manager server.</td>
</tr>
<tr>
<td>Monitor</td>
<td>Provides information on monitoring Device Manager, including reporting, using the Dashboard, server logging, and more.</td>
</tr>
</tbody>
</table>
About This Release

The following features are available in this release:

- Configurable dashboard with device filtering and one-click bulk actions.
- Multi-factor, secure individual or mass device enrollment using notifications.
- Policy creation and deployment for iOS, Android, Windows Mobile, and Symbian with management support for Windows 8 and Windows Phone 8.
- Compliance and security enforcement with automated actions based on customizable compliance criteria.
- Secure document and file sharing with SharePoint resource configurations and secure, on-device document container.
- Encrypted email attachments to ensure only users on managed devices can safely read corporate documents.
- Direct integration with LDAP and Microsoft Active Directory.
- Geo-fencing and geo-tracking for advanced perimeter-based device security and compliance and asset tracking.
- Event-based and ad-hoc notifications and automated actions.
- Remote wipe (full and selective) of device assets.
- App Lock to prevent non-compliant or malicious apps from launching on the device.
Key Features

Device Manager contains the following features:

- Download and install the Device Manager Agent on the mobile device over the wireless or cellular network.

- Deploy device management configuration on the mobile device that includes:
  - Tunnels
  - Registry keys
  - XML configuration files
  - Software packages
  - Files
  - Scripts
- Optimize mobile data traffic by using dynamic compression of data packets transferred between the mobile device and the Device Manager server; maintaining synchronization sessions, with recovery in the event of connection loss or switching between wireless or cellular networks; SSL-AES encryption of all wireless communications applied on data exchange as defined in the Device Manager Administration Console.

- User authentication that supports dynamic integration with Active Directory or IBM Lotus Domino Directory.

- Storage within the repository database of the device that includes:
  - User and device management configurations and settings
  - Device hardware and software inventory
  - Connection logs
  - MortScript scripts signing and macro substitutions before sending files to mobile devices.

- High availability and scalability that provides consistent availability by using a multi-server redundant architecture and load balancing supporting very large deployments.

- Device creation (manual or import from file); hardware and software inventory; remote device wipe.

- User and group creation that includes user role definition and a wizard to modify parameter settings for an LDAP directory connection.

- Application tunnels creation that includes allocation of a channel dedicated to each mobile application with automatic redirection of ports.
Key Features

- Mobile device provisioning that includes the definition of parameters related to:
  - Configuring registry key settings.
  - Running wizards for XML OMA-CP configurations, such as settings for connection parameters and certificate management.
  - Uploading and installing software.
  - Uploading files and scripts.
  - Deployment of packages (tunnels, registry keys, XML files, and software programs) for devices, users, and groups with the capability to restrict deployment based on rules and scheduling.
  - Online mobile activity reports, providing detailed information on user / session / traffic per application / tunnel and connection type.

Device Manager Agent

Device Manager Agent is software which is installed on mobile devices and connects to a Device Manager server. The features depend on the platform type and includes the following Windows Mobile agent's roles:

- Management of all data flow between the mobile device and the Device Manager Agent (device provisioning and application tunnels).
- Installation and enforcement of the configuration setting policies on the mobile device as defined in the Administration Console.
- Sending the mobile device inventory information (hardware and software) to the Device Manager server.
- Display status of the connection to the Device Manager server, such as bytes sent and received, IP address, connection type, security settings, and user credentials.
- Automatic Device Manager Agent installation after the mobile device restarts or users perform a hard reset of the device.
- Execution of signed MortScript scripts send by the server.
- Deployment of a specific release of the MortScript client, along with the Device Manager agent, on a Windows Mobile platform, The release is modified so that:
  - It can run in privileged mode. MortScript executable is signed using a XenMobile signature.
  - It can only execute scripts sent from the Device Manager server. The server signs the scripts before sending them to the mobile device.

Secure Device

Secure Device is an advanced security module composed that includes:
Key Features

- Strong device and server authentication
- Mobile device local data encryption software program

Local data encryption is only available on Windows Mobile devices.

The key features of Secure Device are:

- Integrated Public Key Infrastructure allowing mutual strong authentication between all mobile devices and the Device Manager server based on X509 v3 certificates.
- Automatic and secure initial device enrollment over wireless and cellular networks.
- Automatic download, install and execution of the local device data encryption program including:
  - Data in existing local directories or files on a mobile device
  - Data such as email with attachments, tasks, contacts, and agendas
  - Files stored on removable storage media that are encrypted by using an AES algorithm
  - Local user authentication on the mobile device by using a personal identification number (PIN) code to unlock the mobile device after users restart the device or when bringing the device out of hibernation.
What's New

XenMobile Device Manager contains the following new features:

- **Citrix Mobile Enroll for iOS.** Allows you to enroll your iOS devices and users into Device Manager. (This is a separate app designed for iOS users and is installed on their mobile devices.)

- **Citrix Mobile Connect.** Provides access to your organization’s SharePoint server, allowing you to save documents and files locally, such as secure email attachments, intranet sites, and SharePoint documents, as well as the ability to publish local documents to your SharePoint server. You can configure a wide range of SharePoint access policies for company documents.

- **Certificate Management.** Enables Device Manager to renew or revoke certificates that are issued by XenMobile.

- **SAML Authentication Support for iOS and Android.** You can configure Device Manager (and cloud deployment of XenMobile) to connect with your SAML service and identity providers to enable authentication capabilities that are not dependent on Active Directory.

- **XenMobile NAC REST API (Cisco ISE Enablement).** The XenMobile implementation of Cisco ISE capabilities provides a robust set of REST APIs that enable you to control access to your network by unapproved mobile devices. The REST API queries user devices to execute actions on devices, such as wipe and lock, as well as send notification to devices.

- **Network Access Control.** If you have a Network Access Control (NAC) appliance set up in your network (such as a Cisco ISE), you can enable filters to set devices as compliant or not compliant for network access control based on rules or properties. If a XenMobile managed device does not meet the specified criteria and is marked as **Not Compliant**, the device will be blocked on your network by the NAC appliance.

- **Secure Local Docs on Device from SharePoint.** The XenMobile DLP solution now supports download and markup SharePoint documents and files on your device in the Documents folder. You can check files out for local view, and then check them back in at a later time.

- **SharePoint DLP Personal Folder Support.** Allows access of a user’s personal folder (based on user name) on their SharePoint server. If you allow SharePoint allows personal folders, then you can provide can access to those folders on the user device.

- **App Tunnels for SharePoint Connections.** You can create secure App Tunnels and deploy the tunnels to your SharePoint connections. App Tunnels allow you to create a secure connection to a network resource on a per-app basis. App tunnels define **proxy** parameters between the user component of mobile apps and the app server component.

- **Secure Email Attachments.** Secure email attachments enable you to securely view encrypted email attachments securely through SharePoint DLP and email attachment
What’s New

- Control access and reading of selected file types (\*.doc, \*.PDF, \*.txt, audio, or video to name a few) as encrypted email attachments by using the XenMobile Secure App Container. When you open an attachment that is secured by Device Manager, the attachment is encrypted and secure. If you try to open or view the attachment in any other application or web site on any other device, it is decrypted and rendered unreadable, which protects your sensitive data.

- Restrict or allow the file from being saved locally or opened and read in any other local apps.

- Allow specific file types from being encrypted to allow for viewing, saving, forwarding, and uploading files that do not pose a security risk.

- Restrict attachment viewing so the user cannot copy and paste, print, or email attachments.

- Prevent attached documents from being viewed if the user emails the file to other users or uploads the attachment to a file sharing web site such as Box or Dropbox. Files removed from the XenMobile secure app container become encrypted and unreadable by other users.

- Enable remote selective wipe of email attachment data on a device in the event a user leaves the company or the device is lost or stolen.

- Customize the email subject heading and message to indicate secure nature of attachments.

- Deploy secure email attachment document control policies easily as a standard XenMobile SharePoint DLP policy package.

- Role Base Access Controls for Software Inventory and Location Services. Role-based access controls allow you to manage your software inventory for devices and the device location services.

These permissions allow the main features to function, but allow you to block (de-selected) or allow (selected) users viewing the information. For example, you can block a user from viewing software inventory, but you can block the device by using Secure Mobile Gateway if the users installs a blocked app on the device. You can also block users from viewing location service data but the device can still be geo-fenced or geolocated. Device Manager can also generate reports based on this information.
New Features for Android in Device Manager

- Samsung Certificate Integration. This feature enables you to deploy email, virtual private network (VPN), and Wi-Fi policies to Samsung devices by using two-factor authentication (user name/password + certificate). You can notify and renew certificates automatically without disruption, as well as revoke the certificates and policies of a decommissioned or lost device.

The following new features are provided for your Samsung devices managed by Device Manager:

- Encryption
- Remote Support
- VPN for Samsung Android (including certificates)
- Certificates for Touchdown, Wi-Fi
- Enhanced Android Device Enrollment. You can enroll Android devices by using MDM server discovery, email or SMS notifications and multiple modes for user authentication (password and PIN-based). Your Android device users can enroll by receiving a message, clicking a URL, and then launching the agent installation process to become enrolled. You can choose from several enrollment modes, each of which provides varying degrees multi-factor credentialed security.

- Agent Notification on Android. Device Manager supports sending push notifications to your Android devices.

- HTC MDM Integration. This release provides support for the following policies on Android HTC devices:
  - Exchange ActiveSync. Allows you to remotely configure email by using HTC Home APIs.

Windows 8 Tablet Support

Device Manager supports the following features for the Windows 8 Tablet:

- Device enrollment
- Registry configuration
- Device lock
- Selective device wipe

Windows 8 Phone Support

Device Manager support the following features for the Windows 8 phone:

- Device enrollment
- Hardware inventory
What's New

- PIN-code configuration
- Policy-based device configuration for:
  - Storage Card: Require Device Encryption and Storage Card Disable/Enable
  - Password Policy Configuration
  - Exchange ActiveSync: Configure Exchange ActiveSync server, synchronization settings (email, calendar, contact, frequency), logging.
  - Custom XML configuration abilities
  - Remote Wipe
Feature Support by Device Platform

XenMobile Device Manager allows you to manage the following mobile device platforms:

- Apple handheld devices (iPhone, iPad) using iOS 5.0 or higher
- Android handheld devices using 2.2 or higher
- Windows 8 Phone
- Windows 8 Pro or Enterprise devices (Sideloading license required for enrollment)
- Microsoft Windows Mobile and its derivatives:
  - Windows Mobile 5.x or 6.x (PocketPC or Smartphone Edition)
  - Pocket PC 2003
  - Windows CE 4.x, 5.x or 6.x
  - BlackBerry handheld devices using BlackBerry OS versions 5.x, 6.x, and 7.x
  - Symbian S60 FP2 and above, Symbian 3

Due to platform restrictions or security features, not all features are supported on all platforms. The following table summarizes the features available by platform.

<table>
<thead>
<tr>
<th>Feature</th>
<th>iOS</th>
<th>Android</th>
<th>Windows Mobile</th>
<th>Windows 8 Phone</th>
<th>Windows 8 (Pro or Enterprise)</th>
<th>BlackBerry</th>
<th>Symbian</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dashboard</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Enhanced Enrollment Modes (OTP, Multifactor, Invitation-based)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Invitation Client Download</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Email Attachment Encryption</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>App Lock (‘Kiosk Mode’)</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>App Tunnels Mobile SSL VPN</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storage Card Encryption Policy</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>Autodiscovery Logon</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Feature Support by Device Platform</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Automated Actions</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Notifications</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Agent Notification</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Enterprise App Store</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Locate Device</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>GeoTracking, Geo-Fencing</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Secure Sharepoint DLP</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Remote client installation (OTA)</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Provisioning of devices &amp; users</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Hardware Inventory</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Software Inventory</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Security - Jailbreak detection</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Full Wipe &amp; Lock</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Selective Wipe</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Software download &amp; install</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>File transfer</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Device Remote Control</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Roaming Management</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reports (activity &amp; devices inventory)</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Local device data encryption (option)</strong></td>
<td>![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark] ![Checkmark]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Known Issues

The following is a list of known issues for the XenMobile 8.0.1 release:

- **Self-Signed Certification Limitation.** If you install a self-signed certificate on Device Manager, device enrollment might not succeed on some Android devices. Citrix recommends installing a signed secure certificate on Device Manager. You can also replace the self-signed certificate in the device trusted certificate list. (EWB-10238)

- **Special Character Limitation for Encrypted Email Attachments on Android.** Encrypted email attachments files sent from Android devices cannot contain specific special characters, such as a square bracket, comma, plus, or semicolon, or the files will fail to be encrypted. (EWB-11568)

- **Password Update for non-Samsung Safe Android Devices.** If a certificate is pushed to an Android device, except Samsung SAFE certified devices using the Citrix MDM Connect SAFE agent, the user will receive a prompt to enter a password. Users only need to tap Enter without typing in a password. (EWB-11575)

- **Device Manager Secure Documents Folder Requires Control Policy or Sharepoint Policy.** For example, if your company does not use SharePoint, but does plan to deploy an email attachment policy, it is a pre-requisite that a control policy be deployed. Without it, the Documents folder (where encrypted emails are stored and opened in) will not appear on the device. (EWB-11604)

- **Manage Historical Data Report Export Limitation.** On the Device Manager server, the Manager Historical Data reports exported as a .csv file does not show any data. The exported report is blank. Workaround: You can edit the ‘ew-config.properties’ to enable this report, but be aware that this can increase the reporting database size. This file is located at C:\Program Files (x86)\XenMobile\XenMobile Device Manager\tomcat\webapps\zdm\WEB-INF\classes\ew-config.properties. (EWB-11680)

- **Upgrading to XenMobile Device Manager 8.0.1 - Java SE Perm Gen Increase.** If you are upgrading to XenMobile Device Manager from previous release to 8.0.1, you need to manually increase the Java SE perm Gen size from 156 to 256 (minimum). (EWB-11931)

- **Manually Update Active Directory Password in TouchDown.** If users Active Directory password changes, users need to manually update the password in the TouchDown email application. This does not apply to iOS devices. (EWB-11939)

- **Blacklist/Whitelist Name Changes.** In this release of Device Manager, the terms Blacklist and Whitelist for policies have been replaced. (EWB-11939)

  - **Forbidden (‘blacklisted’).** A ‘forbidden app is an app that should not be installed on a device. If even ONE app on device matches an app in the “Forbidden” list, the device will be considered in violation of the policy.

  - **Suggested (‘whitelisted’).** This is a list of apps that are suggested to users, and they can have one or more of the apps from the list installed and still be in compliance with the policy. However, if the user installed any apps not listed in the policy, they the user’s device is in violation of the policy.
Known Issues

- New - Required. This defines a list of apps that all must be installed on the device to be in compliance with the policy. If any one of the policies is not installed, then the device is in violation of the policy.

- iOS Platform Support. Device Manager support iOS devices running Version 5 and above. Devices running Version 4 are not supported. (EWB-12279)

- Blocking or Allowing Apps Require Updating. Blocking or allowing applications in previous XenMobile (Zenprise) versions was a global setting and applied to all managed devices. With this release, you can configure Forbidden or Suggested application settings by using policy settings in Application Access Policies. If you upgrade to Device Manager 8.0 and previously configured these settings globally, you need to manually deploy the application lists to your managed devices by targeting user groups. (EWB-12280)

- Deploy Web App Store Using Policies. You can deploy the XenMobile web-based app catalog by using the Device Manager Deployment package instead of the XenMobile Agent app. When you deploy the web-based app store, it appears on the device home screen. (12281)

- Deploy Citrix Receiver App to Devices. Citrix recommends that in the first policies that you deploy to you devices, you deploy Citrix Receiver to your devices in order to enable the Enterprise App Store. (EWB-12282)

- Unapproved App Installation Status. If users install an approved app from an external vendor, such as the Apple App Store or synchronizing with iTunes, the XenMobile App Catalog might show the app as uninstalled even though it is installed on the user device. (EWB-12284)

- MTC Tenant Names. When creating new tenants or re-deploying an existing tenant, the Name and Instance Name must be exactly the same with no spaces and case sensitive. If they are not the same, edit Name to match the Instance Name before deploying/re-deploying. (EWB-12391)
Device Manager System Requirements

You can use the following system requirements for installing Device Manager.

**Windows Server**

- Microsoft Windows 2003 Server 64-bit Standard or Enterprise Edition
- Microsoft Windows 2008 Server 64-bit Standard or Enterprise Edition
- Microsoft Windows 2008 Server R2 Standard or Enterprise Edition

**Note:** If you plan to use device certificate templates with Microsoft Certificate Services, the Windows server running the Active Directory Certificate Services must be running Microsoft Windows 2008 Server R2 SP1, Standard or Enterprise Edition.

If you plan to use the SharePoint access management feature, make sure your deployment meets the following Windows 2008 requirements:

- Microsoft Windows 2008 Server R2 Standard or Enterprise Edition with Service Pack 1 or with fix KB976217 installed on the server.
- Microsoft Windows 2008 Server R2 with fix KB976217 installed on the server.

**Java Requirements**

- Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7.

**Note:** If you are upgrading to Device Manager 8.0.1 from a previous version, you need to perform an upgrade to Java 7 from the previous version. See Upgrading Device Manager for more information.

The Java Cryptography Extension (JCE) is an officially released Standard Extension to the Java Platform. JCE provides a framework and implementation for encryption, key generation and key agreement, and Message Authentication Code (MAC) algorithms. For more information, see Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7 on the Oracle web site.

**Note:** Oracle Java components all must be downloaded separately from the download web site. The JCE components must be installed in the JDK's Java Runtime Environment in order to properly support enrollment of iOS devices. Please follow the installation Read Me instructions that accompany the Java JCE download package Java SE Downloads on the Oracle web site.

After you download and extract the JCE package, copy the files local_policy.jar and US_export_policy.jar to the $<java-home>\jre\lib\security folder and overwrite the existing files.
Hardware Requirements

- Physical or virtual Server host environment
- Intel Xeon 3Ghz or AMD Opteron-1.8Ghz server class
- 1 Gb RAM minimum (4 Gb RAM minimum recommended for 64-bit OS)
- 500 Mb free disk space minimum

Example of a recommended configuration for 1,500 mobile users:

- Dual-processor computer with Intel server architecture.
- A minimum of 2 GB RAM available for the application. 1 GB free disk space (without database).

Example of a recommended configuration for 10,000 mobile users:

- Dual quad-core processor computer with Intel server architecture
- 8 GB RAM available for the application. 1 GB free disk space, 5 GB for database

Web Browser Requirements for the Device Manager Web Console

- Internet Explorer Versions 8 and 9
  
  **Note:** The Device Manager 7.0 Dashboard feature is not supported on Internet Explorer 8.0, but is supported on Internet Explorer 9.

- Firefox versions 10 and 11

Device Manager Repository Database Requirements

The Device Manager Server repository requires a relational Database Management System (R-DBMS). The supported databases are:

- PostgreSQL v8.2 is the default RDBMS that is packaged and installed with Device Manager Server
- Microsoft SQL Server 2005, 2008 or 2008 R2

**Important:** The server on which the database is installed requires NTFS partitions.

User Account Needed: For the database server, you will need a service account that has admin rights to SQL server, including the following access rights: Creator, Owner, and Read/Write permissions.
Windows Service Account Requirements

The Windows service accounts for the Device Manager Server and the database must be a Local Administrator of the computer on which the Device Manager Server is installed.

SharePoint Data Leak Prevention Requirements

XenMobile supports SharePoint Data Leak Prevention (DLP). To deploy SharePoint DLP in your environment, use the following guidelines:

- SharePoint 2010 or Office 365.
- Windows Server 2008 R2 - SharePoint 2010 with Service Pack 1 is required or install KB976217.
- Windows Server 2008 - Rest API calls will fail unless KB976217 is also installed.

Installation Requirements

When you install XenMobile, use the following guidelines:

**Note:** Domain membership is not required for the Device Manager server.

- Do not install a new version of IIS, and uninstall IIS if it exists on this server.
- Create an external DNS record for the Device Manager server, such as mobile.yourcompany.com.
- Obtain an Apple APNS certificate which is needed during the installation of Device Manager. You can obtain an APNS certificate by using the XenMobile APNS Certificate Setup Guide.
- Install Java SE 7 Update 11 (jdk-7u4-windows-x64.exe) on the Device Manager server.
- Install Java Cryptography Extension (JCE) USJP 7 on the Device Manager server.

**To install the Java Cryptography Extension**

1. Install Java SE 7u11.
2. Open the JCE zip file and copy local_policy.jar and US_export_policy.jar to your computer desktop.
3. Navigate to the folder /Java/jdk1.7.0_x/jre/lib/security and copy the files from Step 2 to this folder.
Preparing for Device Manager Installation

There are four prerequisite requirements that must be prepared prior to installation of Device Manager server. Each prerequisite has a subset of requirements that belong to the providing service and the infrastructure groups responsible for implementation and change control. A successful installation requires all prerequisites are met.

Obtaining an APNS certificate from Apple for iOS Devices

Management of Apple iOS devices by using the native MDM capabilities of the mobile device hardware and operating system requires an APNS certificate to communicate via Apple Push Network Services. In order to obtain a certificate from Apple, follow the steps outlined in the APNS Certificate Request Guide.

Designating a DMZ IP Address and DNS Host Name

The Device Manager server is designed to be an edge gateway server that resides in the network DMZ. Device Manager requires a static IP address that can be reached from the internet, as well as a registered and published DNS host name so that devices can reach the server during enrollment and communicate with regularly. It is strongly recommended to use a separate A-record or CNAME record for any host living in a DMZ for anonymity of the true server host name.

Opening Ports in the Firewall

There are many inbound and outbound ports that must be configured on the network between the Internet and the DMZ, and from the DMZ to your secure network.

The following table is designed to provide a guide for the TCP/IP port requirements for the Device Manager server and mobile device agent connections.

<table>
<thead>
<tr>
<th>Port</th>
<th>Description</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>By default, the Device Manager SMTP configuration of the Notification Service uses port 25. However, if your SMTP server uses a different port, make sure that your firewall does not block that port.</td>
<td>Device Manager Server</td>
<td>SMTP Server</td>
</tr>
<tr>
<td>Port</td>
<td>Feature Description</td>
<td>Environment</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-------------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>Over-the-Air (OTA) Enrollment and Agent Setup (Android and Windows Mobile)</td>
<td>Internet</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Over-the-Air (OTA) Enrollment and Agent Setup (Android and Windows Mobile), Device Manager management console, Device Manager Remote Support Client</td>
<td>Secure network and WiFi</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device Manager server enterprise connection to Apple iTunes App Store (ax.itunes.apple.com). Used for publishing recommended iTunes App Store apps from the available iOS applications within the Device Manager management console and the iOS Agent.</td>
<td>Device Manager Server</td>
<td></td>
</tr>
<tr>
<td>80 or 44</td>
<td>Device Manager Nexmo SMS Notification Relay outbound connection.</td>
<td>Device Manager Server</td>
<td>Nexmo SMS Relay server</td>
</tr>
<tr>
<td>38 or 63</td>
<td>LDAP/LDAPS connection from Device Manager server to Directory Service Host (Active Directory Global Catalog server or equivalent LDAP directory service host)</td>
<td>Device Manager Server</td>
<td>LDAP or Active Directory Services</td>
</tr>
<tr>
<td>44 or 3</td>
<td>SSL OTA Enrollment or Agent Setup (Android and Windows Mobile), All device-related traffic and data connections (iOS, Android, and Windows Mobile).</td>
<td>Internet</td>
<td>Device Manager Server</td>
</tr>
<tr>
<td></td>
<td>SSL OTA Enrollment or Agent Setup (Android and Windows Mobile), All device-related traffic and data connections (iOS, Android and Windows Mobile), Device Manager management console.</td>
<td>Secure network and WiFi</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device access to 'discovery.mdm.zenprise.com' on port 443 for autodiscovery enrollment.</td>
<td>Autodiscovery</td>
<td></td>
</tr>
<tr>
<td>14 or 33</td>
<td>Remote database server connection to separate SQL Server (Optional).</td>
<td>Device Manager Server</td>
<td>SQL Server</td>
</tr>
<tr>
<td>21 or 95</td>
<td>Apple APNS (Push Notification Service) outbound connection to gateway.push.apple.com, used for iOS device notifications and device policy push.</td>
<td>Device Manager Server</td>
<td>Internet (Apple APNS Service Hosts on public IP network 17.0.0.0/8)</td>
</tr>
<tr>
<td>21 or 96</td>
<td>Apple APNS (Push Notification Service) outbound connection to feedback.push.apple.com, used for iOS device notifications and device policy push</td>
<td>Device Manager Server</td>
<td></td>
</tr>
<tr>
<td>52 or 23</td>
<td>Apple APNS (Push Notification Service) outbound connection from iOS devices connected via Wi-Fi network to *.push.apple.com</td>
<td>iOS device on WiFi network service</td>
<td></td>
</tr>
</tbody>
</table>
### Preparing for Device Manager Installation

<table>
<thead>
<tr>
<th>Port</th>
<th>Description</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>84</td>
<td>Over-the-Air (OTA) Enrollment for iOS Devices only</td>
<td>Internet, secure network, or WiFi</td>
<td>Device Manager Server</td>
</tr>
<tr>
<td>43</td>
<td>Mobile App Tunnel Ports (Android and Windows Mobile) to the destination internal Application Server through Device Manager. All ports are individually defined for each mobile app tunnel used by a device through a Device Manager Device Configuration Policy.</td>
<td>Internet</td>
<td>Application Server through Device Manager Server</td>
</tr>
</tbody>
</table>

When using Remote Support or the Mobile App tunnel (Android and Windows Mobile), the following traffic needs to be open at the firewall:

<table>
<thead>
<tr>
<th>Port</th>
<th>Description</th>
<th>Source</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>Remote Support Console access to Device Manager to retrieve device list.</td>
<td>Remote Support Console</td>
<td>Device Manager Server</td>
</tr>
<tr>
<td>443</td>
<td>Mobile Application Tunnel access to Application Server (port configured in the tunnel definition)</td>
<td>Device Manager Server</td>
<td>Internal Application Server</td>
</tr>
</tbody>
</table>
Device Manager Remote Support Requirements

Hardware Requirements

- Intel Xeon/Pentium 4 -1 GHz minimum workstation class
- 512 Mb RAM minimum
- 100 Mb free disk space minimum

Operating System Requirements

- Microsoft Windows 2003 Server Standard Edition or Enterprise Edition with a minimum of Service Pack 1
- Microsoft Windows 2000 Professional with Service Pack 4
- Microsoft Windows XP with a minimum of Service Pack 2
- Microsoft Windows Vista with a minimum of Service Pack 1
- Microsoft Windows 7
Planning Your Device Manager Deployment

Before you install Device Manager, you should collect and record configuration information in order to complete a successful installation.

Citrix recommends deploying Device Manager in the demilitarized zone (DMZ).

This section includes a checklist that helps you define the information you need about the following:

- TCP/IP LAN connection of 100 Mbps or faster
- Static IP address
- Domain Name Server (DNS) name published on the Internet and your local area network
- Ports
- Virtual private network (VPN) connection settings
- Authentication, including Active Directory settings
Device Manager Pre-Installation Checklist

Before you install Device Manager, you should collect and record configuration information in order to complete a successful installation. This section includes a checklist that helps you define the information you need about the following:

- TCP/IP LAN connection of 100 Mbps or faster
- Static IP address
- Domain Name Server (DNS) name published on the Internet and your local area network
- Ports
- Virtual private network (VPN) connection settings
- Authentication, including Active Directory settings

This checklist lists the tasks you should complete and the configuration values you should note before you install Device Manager 8.0.

For instructions about installing and configuring Device Manager, see Installing Device Manager Server.

Device Manager Basic Network Connectivity

The Device Manager server is designed to be an edge gateway server that lives in the network DMZ. It will need to have a static IP address that is reachable from the public Internet, as well as a registered and published DNS host name so that devices can reach the server during enrollment and communicate with regularly. It is strongly recommended to use a separate A-record or CNAME record for any host living in a DMZ for anonymity of the true server host name.

<table>
<thead>
<tr>
<th>Public IP address</th>
<th>DNS host name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Note the default gateway IP address.</td>
</tr>
</tbody>
</table>

Device Manager Service Account

The Device Manager service account must be a local administrator of the server. When you configure the service account, it must have the credentials to create databases in Microsoft SQL during Device Manager installation.

<table>
<thead>
<tr>
<th>Service account user name</th>
<th></th>
</tr>
</thead>
</table>
Database Installation

Device Manager creates a database during installation to an existing SQL server. Citrix recommends that the database reside on a separate server than Device Manager.

| SQL server fully qualified domain name or IP address |  

---
Deploying XenMobile MDM - Device Manager

XenMobile Device Manager is comprised of the following separate server components:

- Device Manager is one server or virtual machine that is typically located in the DMZ. This component provides advanced mobile device management and security for iOS, Android, Symbian, Windows Phone 8, Windows 8, Windows CE, and Windows Mobile devices.

- Secure Mobile Gateway is the automated enforcement component that can prevent unmanaged or out-of-compliant devices from accessing your secure network mail environment. You can install the Secure Mobile Gateway on an Exchange Client Access server, Exchange Front End, Microsoft ISA Server, or Microsoft Threat Management Gateway and does not have any prerequisites to complete before installation. For more information, see Secure Mobile Gateway 8.0.1. You can install the Secure Mobile Gateway you install the Device Manager server.

You can install Device Manager to support a variety of existing network topologies and a large number of users. Some of the installation considerations are related to how the users connect to their Information System (Wi-Fi, Cellular, and Ethernet), existing security rules (DMZ, firewalls), user authentication (Directories), and more. Device Manager should be installed on a standalone physical server or dedicated virtual machine.

In order to be compliant with existing IT infrastructures Device Manager can be deployed in various scenarios:

- Simple installation: Device Manager is installed behind the firewall.

- Multi DMZ installation: Device Manager is installed in the private DMZ behind a proxy located in the public DMZ.

The following figure shows a typical Device Manager deployment.

Figure 1. Device Manager Deployment
Citrix recommends deploying the Device Manager server in the DMZ as a perimeter security server for mobile device management. In a conventional single DMZ architecture, prior to installing the Device Manager Server, make sure that the firewalls authorize the network streams.

You can also install Device Manager server in the secure network, however it will require all network firewall ports described in “Device Manager Server” in Product Requirements from the Internet to be allowed inbound to your secure network where the server is placed to fully operate correctly. See Preparing for Device Manager Installation for information on required ports for a Device Manager deployment.

For Device Manager Server connectivity, make sure that it has:

- A TCP/IP LAN connection of 100 Mbps or more
- A static IP address, and dedicated Domain Server Name (DNS) published both to the Internet and your network
- Availability of all required network firewall ports to allow device traffic inbound from the Internet to the Device Manager server, as well as from the Device Manager server to and from your internal network

The computers on which you install Device Manager Remote Support connects to the Device Manager server IP address on port 80 (by default) to retrieve the list of connected devices and on a port selected (port 82 for example) during installation for remote control of devices through the Device Manager server.
XenMobile provides an installation setup wizard which automatically installs the necessary components to use Device Manager. During installation, you install the following components:

- Device Manager
- PostgreSQL database
- License files

Before you install make sure that you:

- Disable TCP/IPv6 on the network adapter and in Windows Registry Editor.  
  
  **Caution:** Using Registry Editor incorrectly can cause serious problems that may require you to reinstall your operating system. Citrix cannot guarantee that problems resulting from the incorrect use of Registry Editor can be solved. Use Registry Editor at your own risk.
- Disable the User Account Control Setting in Control Panel.

The installation topics in this section cover the default installation of the Device Manager server on a Windows 2008 Server.
Choosing Device Manager Components to Install

If you are installing Device Manager on your computer for the first time, select Full install, which installs:

- The Device Manager server
- The Device Manager repository database (PostgreSQL) and automatic creation of the database and requisite tables
- The integrated web application server hosting the Device Manager server

**Note:** If you install an Application Server prior to installing Device Manager, remove Application Server before installing Device Manager.

Installing Databases

Device Manager includes the PostgreSQL database server installation. If you installed a SQL database server on your computer or another server, clear the PostgreSQL check box in the list of components during the installation wizard. The install type switches automatically to Custom. When using a Microsoft SQL server please refer to the installation instructions provided by Microsoft for the SQL server installation. If you do not clear the check box, the PostgreSQL installation wizard appears with configuration instructions.

If you install PostgreSQL, an installation wizard appears. The installation program automatically selects all the default PostgreSQL options required to install a Device Manager server. However, you can check any additional options you want to install. You can also change the installation location with the Browse button.

During installation of PostgreSQL, define the service account that runs the PostgreSQL server. The Service name, Account name, and Account domain fields are already completed. You need to enter a password for the service account.

If the user account does not exist, you receive a prompt to confirm creation of the account. In addition, if the password you chose is not a strong password, then you are prompted to replace the password with a random strong password. Click No in the message dialog box to keep the password you originally entered.

Installing License Files

After you configure the PostgreSQL database, you can then install licenses. If you are using a different SQL database and did not install PostgreSQL, after choosing the initial components and installation location, you install the licenses.
To install Device Manager

Before you install Device Manager, make sure you do the following:

- Disable TCP/IP6 on the NIC and in the registry.
- Disable the User Account Control setting in Control Panel.

The setup wizard includes several discrete tasks. You need to complete all of the tasks in this topic in consecutive order to complete the entire wizard.

To start the installation wizard


2. On the Installer Language page, select your preferred language, click OK and then on the Welcome page, click Next to start the installation.

3. On the License Agreement page, read the terms and then click I Agree to accept the terms and conditions.

4. On the Choose Components page, do one of the following and then click Next.
   - If you are installing Device Manager for the first time, select Full.

     Full installation includes the following:

     - Installation of the Device Manager Server
     - Installation of the Device Manager repository database (PostgreSQL) and automatic creation of the database and requisite tables
     - Installation of the integrated web application server hosting the Device Manager server

     **Note:** If the Application Server has already been installed, Citrix recommends that you remove it prior to a fresh installation. Installing the Application Server component only and reusing an existing Device Manager database is supported.

     - If the SQL database server is already installed on your computer or on another server, clear the check box. The install type changes to Customized. For example, clear the PostgreSQL component if you are using a local or separately installed instance of Microsoft SQL Server.

     The following steps assume that you are installing Device Manager for the first time and that you chose Full in Step 4.

5. On the Choose Install Location page, leave the default install location or click Browse to select a folder on your computer and then click Install. The PostgreSQL component installs.
To complete the PostgreSQL installation

1. On the PostgreSQL Installation notes page, click Next.

2. On the Installation options page, keep the default options, select additional options, or click Browse to change the installation location and then click Next.

3. On the Service configuration page, define the server account that runs the PostgreSQL server by doing the following and then click Next.
   a. Leave the default settings for service name, account name, and account domain. Also, leave the Install as a service option selected.
   b. In Account password and Verify password, enter a password that meets the password policy of your organization.

   **Important:** If the password you enter does not meet the password requirements, after Step 9, a Password notification appears prompting you to enter another password. If the password does not comply with the organization policy, the installation may fail without warning.

   An Account error message appears stating that the user is not found and prompts you to create the user.

4. Click Yes.

   **Important:** Although the message appears as an error, the message means that the designated account doesn't yet exist and that you must create the account.

5. If a Password notification warning appears, click No.

6. On the Initialize database cluster page, do the following:
   a. In Locale, click English.
   b. In Superuser name, define an administrator account for the database.
   c. In Password and Password (again), type a password and then click Next.

7. On the Enable procedural languages page, leave the default PL/pgsql check box selected and then click Next.

8. On the Enable contrib modules page, select any modules you want to enable, click Next and then click Next again to start the database installation.

9. When the database installation is complete, click Finish.
To configure the connection to the RDBMS

1. On the Device Manager License page, click Browse to specify the .crt license file on your computer that contains the valid license keyword provided by Citrix and then click Next. Next, you configure the Device Manager Repository connection parameters for the selected relationship database management system (RDBMS).

2. On the Configure database connection page, in Database driver, click an RDBMS, such as PostgreSQL.

   - If you click PostgreSQL or a Microsoft SQL database, you need to configure a user name, as well as the following:
     - In Password, enter the password you defined when you installed the PostgreSQL database.
     - In Database name, enter a database name or leave the default value.
   
   - If you click a database other than PostgreSQL, such as SQL Server, you need to configure the SQL Server home name or IP address, as well as the communication TCP port of the database server. The default TCP port is 1433.

   Note: Be sure to use the correct local or domain account user name, password, and desired database name that you configured during installation. The account used for Microsoft SQL should also have db_owner, db_creator, db_writer, and processadmin rights.

3. Click Check the connection and then click Next.

4. In the Confirmation message, click Create and then when a message appears stating that the connection to the database is successful, click OK.

To configure and register Crystal Reports

With Crystal Reports, you can process the mobile device connection and session logs to generate activity reports online by using the Device Manager web console, or offline from the Device Manager repository database. The reports include watermark with registration information. To remove the watermark, you need a Crystal Reports Developer Edition license and a keycode for the product. If you did not enter a license serial number during installation, you can define it later by following these steps:

1. Edit the crconfig.xml configuration file located at in the Device Manager setup folder, which is typically \Program Files\Xenmobile on a Windows server:

   tomcat\webapps\Device Manager\WEBINF\classes\crconfig.xml

2. Add your serial number by editing the <keycode></keycode> element. For example, if your serial number is XXXX-YYYY-ZZZZ, modify the line as follows:

   <keycode>XXXX-YYYY-ZZZZ</keycode>

3. On the Crystal Report Java Reporting Components configuration page, to leave a watermark on the reports, leave the keycode blank. Or, to remove the watermark, enter your keycode for the product and then click Next.
To configure the server connectors

On this page, you configure the connection between the Device Manager agent and the Device Manager server for the initial download of the Device Manager agent and subsequent updates and for establishing connections between the Device Manager agent and the Device Manager server in a common operation.

1. On the Configure the modes of connection page, configure the following:

   a. Enable iOS. Select this check box if you manage iOS devices.

      **Important:** You can only configure this option during installation. If you do not select this option and you want to enable the mode in the future, you must reinstall the application server.

   b. Both HTTP and HTTPS access (recommended). Click this option to complete the standard configuration, which enables HTTP downloading and common HTTPS connections.

   c. Only https. Click only if the agent setup wizard file (setup.cab or setup.apk) is not downloaded through the web browser of the Windows Mobile or Android device.

   d. Only http. Click only if a VPN is already installed between the Device Manager server and the mobile devices.

   e. Authentication code for applications/tunnels. Enter a prefix that Device Manager uses to create authentication keys used by the software. Use a simple alphanumeric word or passphrase. Use mixed cases, numbers, and letters only. Then, record this value for use later when you configure the system.

2. On the Define an HTTP connector page, leave the default port of 80 or enter a different port number.

3. In Maximum concurrent connections, leave the default value of 20 or enter a different number. In standard operating mode, HTTP connections are used only to upload the Device Manager agent and to connect the Device Manager Remote Support application. Each connection represents how many device (client) connections you want to allow at any one given time, simultaneously.

   If you want to allow inbound connections to the Device Manager server through the HTTP port, you can also give your server an external IP address.

   **Note:** If you change the Device Manager server IP address, the change is transparent to users if the external address in the Device Manager Server SSL certificate has not been changed and you chose the All local Addresses option during configuration. It's also recommended that you update the security rules.

4. On the Define an HTTPS connector page, leave the default port of 443 or enter a different port number.

5. In Maximum concurrent connections, leave the default value of 400 or enter a different number and then click Next.

6. On the Define an HTTPS connector for iOS enrollment page, you can leave the default port of 8443 or enter a different number. iOS standards typically rely on a connection through port 8443.
7. In Maximum concurrent connections, leave the default value of 20 or enter a different number and then click Next.

To integrate the PKI

The Device Manager server has an integrated Public Key Infrastructure (PKI) which incorporates several Certification Authorities (CAs) to manage the key pairs and certificates required to authenticate the server and mobile devices. The certificates are in X509 v3 format. The Device Manager server is always authenticated, although the device authentication is optional and is only activated and applied if the license includes the Device Manager Secure Device option.

**Note:** If the Device Manager Secure Device option is not included in your Device Manager license, Device Manager does not use the CA for mobile devices.

1. On the Define the root certification authority page, do the following and then click Next.
   a. In Keystore file path, do not change the default path. The server configuration provided the file path.
   b. In Keystore password, enter the private password used for each component of the local CA. Although you can use the same password for each CA keystore component, it is recommended to use separate passwords for the Root CA, Server CA, Device CA, and Web Service CA certificates. Be sure to write down all keystore passwords uses and save in a safe location. Matching green color password fields confirm that you entered the same password in the Password and Confirm password fields. Passwords must have at least eight characters, and can consist of alphanumeric and ASCII symbol values. Passwords are case sensitive.
   c. In Common name, leave the default name to associate with the creation of the CA component and certificate. If you change this field, your devices may not receive the proper chain of certificates and will not be able to enroll.
   d. In Organizational unit, enter a value typically given to the entity or group that has management authority over the CA.
   e. In Organization, enter a value typically given to the entity or company that is parent to owning the CA and its rights.

   **Note:** The Root CA certificate is used to issue and sign certificates for intermediate server and client-device CAs. It is also used to regenerate intermediate certificates in the event of compromise. It may be installed in the operating system as a trusted CA root certificate. To avoid alert messages by Internet Explorer 7 as to the validity of certificates issued by this CA, install the root certificate in the operating system.

2. On the Define the server certification authority page, do the following and then click Next. The intermediate mobile device CA is used to issue and sign mobile device certificates. It is also used to regenerate mobile certificates in the event of compromise.
   a. In Keystore file path, do not change the default path. The value is required by the server configuration.
b. In Keystore password, enter the private password used for each component of the local CA. Although you can use the same password for each CA keystore component, it is recommended to use separate passwords for the Root CA, Server CA, Device CA, and Web Service CA certificates. Be sure to write down all keystore passwords uses and save in a safe location. Matching green color password fields confirm that you entered the same password in the Password and Confirm password fields. Passwords must have at least eight characters, and can consist of alphanumeric and ASCII symbol values. Passwords are case sensitive.

c. In Common name, leave the default name to associate with the creation of the CA component and certificate.

d. In Organizational unit, enter a value typically given to the entity or group that has management authority over the CA.

e. In Organization, enter a value typically given to the entity or company that is parent to owning the CA and its rights.

3. On the Define the certificate for HTTPS page, do the following and then click Next. The server shows the HTTPS certificate (SSL server connection) to the mobile devices in order to prove the server identity. The certificate prevents man-in-the-middle attacks. A man in the middle attack is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is controlled by the attacker. The attacker must be able to intercept all messages going between the two victims and inject new ones, which is straightforward in many circumstances (for example, an attacker within reception range of an unencrypted Wi-Fi wireless access point, can insert himself as a man-in-the-middle)

a. In Keystore file path, do not change the default path. The value is required by the server configuration.

b. In Keystore password, enter the private password used for each component of the local CA. Although you can use the same password for each CA keystore component, it is recommended to use separate passwords for the Root CA, Server CA, Device CA, and Web Service CA certificates. Be sure to write down all keystore passwords uses and save in a safe location. Matching green color password fields confirm that you entered the same password in the Password and Confirm password fields. Passwords must have at least eight characters, and can consist of alphanumeric and ASCII symbol values. Passwords are case sensitive.

c. In IP address or FQDN, enter the IP address or fully qualified domain name (FQDN) of the server.

d. In Organizational unit, enter a value typically given to the entity or group that has management authority over the CA.

e. In Organization, enter a value typically given to the entity or company that is parent to owning the CA and its rights.

**Important:** Retain and safely store all keystore passwords used for the four separate CA installation steps. These are the PKCS#12 certificate files containing the PKI key pairs (*.p12 files) and can be found in the default keystore file path listed during each installation step. Also, keep backup copies of the four *.p12 extension files.
4. On the Define the APNs certificate file for iOS page, in Private key password, enter the associated private key password used to generate the original Certificate Signing Request (CSR).

5. In Certificate file path, specify the file system location of a pre-authenticated Apple Push Notification Service (APNS) certificate file downloaded and converted to PKCS#12 format from the Apple iOS Developer for Enterprise portal.

   **Note:** APNS certificates are provisioned by Apple, Inc. To obtain an APNS certificate, sign in to the following site with your Apple ID: [https://identity.apple.com/pushcert](https://identity.apple.com/pushcert). Inspect the values that appear to the page's text area. If the certificate and password match, the Next button is enabled. Click Next.

6. On the Configure tunnel port(s) used by remote support page, define the port range used by remote support for Android and Windows Mobile devices and then click Next. When you connect to the Device Manager server from the Device Manager web console and then click the Security Report icon in your Internet browser, the certificate displays.

---

**To designate the Device Manager administrator**

To connect to the Device Manager web console, you need to configure an account with the administrator role.

1. On the Extended management of the users page, in User name, enter the administrator’s name.

2. In Password and Confirm password, enter your password. The password must have at least eight characters.

3. Click Check the user name and then click Next.

---

**To complete the server configuration**

1. On the Configure all page, click Finish. When the installation is complete, a page appears stating that the setup was completed successfully.

2. Click Next and then click Finish.

Next, you install the Remote Support and the device provisioning modules.
To log on to the Device Manager web console

1. Start a Device Manager administrative session by entering one of the following Web addresses in a Web browser:
   
   - http://<device_manager_server_name_or_IP_address>/zdm
   
   - https://<device_manager_server_name_or_IP_address>:<port>/zdm

2. Enter the logon credentials of the administrative user account created during Device Manager installation.

   **Note:** If you configured LDAP authentication for Device Manager, be sure to use the account credentials of a user who is a member of the administrative group.

   After Device Manager validates the account, the main Device Manager window appears.
Creating an LDAP Connection to a User Directory

From the Options dialog box in Device Manager, you can perform the following actions for LDAP connections:

- Create a new LDAP connection.
- Edit an existing connection.
- Set the default LDAP connection.
- Activate or deactivate an LDAP connection.

1. To create a new LDAP connection, click New.
2. Select which type of directory (LDAP or LDAPS).
3. If you chose an LDAPS connection, enter the required parameters and then click Import.
4. After the SSL Certificate is successfully imported, click Next.
5. Define the connection parameters.
   Make sure that the Search user Service Account has the following rights granted to it:
   - READALLUSERINFORMATION
   - READALLNETWORKPERSON

   **Note:** In the lockout limit field, the default is set to zero. However, Citrix recommends using a higher value, as well as a value that is slightly lower than the lockout limit set on your LDAP server. For example, if your LDAP server is configured to a limit of five attempts before lockout, Citrix suggests that you enter a 4 or a 3 in this field.

6. Click Check to test the connection with the LDAP or LDAPS directory. If the connection check with the directory is successful, the following message appears: LDAP directory binding successful.

7. Click OK and then click Next to map the directory attributes to the Device Manager Repository database. You can leave that step as it is and Device Manager will automatically bind the default fields.

8. Click Next to define the mapping between the LDAP groups and Device Manager roles. To add a new group, press Add a group. Select a group and define the role you want to give to that LDAP group.

   **Note:** Unlike the process for creating groups within the web console in a standalone manner in which roles are given to users, here roles are given to an LDAP group.
9. Specify which LDAP or LDAPS directory groups are imported in the Device Manager Repository database and then click Next. A window appears summarizing the directory connection configuration.

10. Click Finish to save the parameters in the Device Manager database.
Upgrading Device Manager

Upgrading the Device Manager server is a simple, in-place upgrade process. The automated Setup Wizard updates your existing Device Manager installation and database in one step. As a best practice, it is advised to backup the database and Device Manager core application directories and save them to a location as a roll-back plan.

Before You Upgrade

You can upgrade from Zenprise versions 7.0.1 and 7.1 to XenMobile 8.0. But before you do so, you need to perform the following pre-upgrade steps:

1. Make sure that you are running Oracle Java SE 7 JDK (JDK Download Edition) update 11 and above and Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7. The JCE components must be installed in the JDK's Java Runtime Environment in order to properly support enrollment of iOS devices. Please follow the installation Read Me instructions that accompany the Java JCE download package Java SE Downloads on the Oracle web site.

2. After you download and extract the JCE package, copy the files local_policy.jar and US_export_policy.jar to the <java-home>\jre\lib\security folder and overwrite the existing files.

3. Next, make sure that your Java SE application is pointing to the correct path for the Java 7 installation.

4. From the Start menu on the Device Manager server, click Edit Service.

5. In the Zenprise Device Manager dialog box for Java, click on the Java tab.

6. In the Java Virtual Machine field, enter the correct path to the Java 7 Update 11 version of Java on your device manager server. For example, C:\Program Files\Java\jdk1.7.0_11\jre\bin\server\jvm.dll.

7. Next, in the Java Options section, increase the Java SE's Permanent Generation ("permgen") space size to a minimum of 256. For example, -XX:MaxPermSize=268m.

8. Click OK.

9. Restart the Java Virtual Machine by clicking Stop Zenprise Device Manager from the Start menu, and then clicking Start Zenprise Device Manager.

To upgrade Device Manager

1. Double-click the Device Manager executable installation file.

2. Follow the directions in the Setup Wizard.
If you are upgrading from previous versions of Device Manager to 8.0.1, and you had deployed Blacklist policies, then you will need to reconstruct those policies and redeploy them to devices. In Device Manager, Blacklist policies are part of Applications Access Policies and are categorized as "Forbidden" apps. For more information, see Application Access Policies.
Upgrading iOS, Android, and Samsung Client Device Manager Apps

To upgrade the client Device Manager apps to the current Citrix Mobile Connect app, you need to perform the following tasks depending on your device type:

- **iOS.** The ZP MDM app has been renamed to Citrix Mobile Connect. To upgrade the app to the new version, go to the Apple iTunes store and check the available updates. When you update the app, it will install the new Citrix Mobile Connect app. You will not need to re-enroll. If for some reason you do need to re-enroll (if you deleted the app or are experiencing issues), you can go to the iTunes app store and download the Citrix Mobile Enroll app to re-enroll.

- **Android and Samsung.** The Zenprise for Employees and Zenprise for Samsung apps have been renamed to Citrix Mobile Connect (and for Samsung). To upgrade from the previous version of the Zenprise for Employees or Zenprise for Samsung client apps, go to the Google Play store on your device, tap the My Apps icon at the top of the store and then look for the Citrix Mobile Connect app. Tap the app, and then tap the Update button.
Licensing

If you would like to obtain a new or renewed license for your XenMobile products, contact your account representative. Citrix XenMobile requires you to purchase one of the following licensing options:

**XenMobile MDM Edition**

Citrix XenMobile MDM edition is licensed under a per-user or per-device model. User licensing is based on how many unique users have registered devices on the administrative device management server. Device licensing is based on how many unique devices are registered on the administrative device management server. XenMobile MDM edition can be sold under three consumption options:

- Perpetual licensing entitles the user to consume the product in perpetuity (forever).

- Annual licensing entitles the user to consume the product on an annual basis only. Product consumed under the annual model is valid for 12 months from point of purchase after which the license will expire and the administrative management server will no longer be able to control the registered devices. Customers may purchase a new XenMobile annual license prior to the expiration date to assure access to their XenMobile environment will continue to function uninterrupted. Co-termination is not supported for XenMobile annual licenses - meaning Citrix will not support synchronization of the expiration dates for XenMobile annual licenses, either individually or in combination with XenMobile perpetual licenses. Annual licenses are good for 12 months from the date of purchase. This cannot be modified to provide 12 months from the date of the expiration associated with any existing annual licenses.

- Hosted Cloud-based Service entitles the purchased number of users or devices to consume the hosted service for 12 months from point of purchase after which the service will expire and administrators will no longer be entitled to access the administrative device management server.

**XenMobile Enterprise Edition**

XenMobile Enterprise edition is licensed under a per-user model only. XenMobile Enterprise edition can be sold under two consumption options:

- Perpetual licensing entitles the user to consume the product in perpetuity (forever).

- Annual licensing entitles the user to consume the product on an annual basis only. Product consumed under the annual model is valid for 12 months from point of purchase after which the license will expire and the administrative management server will no longer be able to control the registered devices. Additionally, the licenses will cease to support valid connections to the XenMobile environment. Customers may purchase a new XenMobile annual license prior to the expiration date to assure access to their XenMobile environment will continue to function uninterrupted. Co-termination is not supported for XenMobile annual licenses - meaning Citrix will not support
Renewing Device Manager Licenses

You can view the current status of your Device Manager license and update your license on the About tab.

If your current license has already expired and you cannot log in to the Device Manager web console, when you receive your updated license, you can manually add the license file to the server running Device Manager to reinitiate the software.

To update an expired license

1. Log on to the server running Device Manager.

2. Browse to the following location on the server: C:\program files(x86)\Citrix\XenMobile Device Manager\tomcat\webapps\zdm\WEB-INF and save the valid file in the WEB-INF folder with a name, such as license.crt.

3. Restart the XenMobile service.

   The license file is updated and you can now log on to the server running Device Manager.
Managing Device Manager

After you install and set up Device Manager, you manage applications, users, and policies. This section covers the following topics:

- Adding users and groups in Device Manager that includes adding users from LDAP, manually, or from a provisioning file.
- Creating policies that include:
  - Application tunnels to improve network issues, security, and reliability
  - Creating software inventory packages that allows you to check all apps and software packages installed on user devices.
  - Configuring iOS and Android policies
  - Configuring Windows Mobile policies
  - Configuring SharePoint policies
  - Configuring Symbian policies
  - Configuring connections with the Scheduling Wizard
- Configuring security in Device Manager that includes:
  - Configuring certificates
  - Configuring a SAML service provider
  - Configuring general security options
  - Configuring network access controls
  - Configuring Microsoft Certificate Services
  - Configuring location services for devices
- Adding files and applications to Device Manager
- Configuring application access policies
- Creating deployment packages
- Configuring general options in Device Manager
- Configuring automated actions
- Configuring notifications
- Configuring high availability
Defining Users and Groups

User account objects represent the users of the mobile devices managed by Device Manager. User accounts are associated to devices by Device Manager as part of the authentication process. Maintaining an accurate roster of users improves mobile device and service management. Groups are logical collections of users that serve as targets for management tasks, such as applying settings, implementing policies, and deploying software.

**Note:** Device Manager manages group of users, not individual user accounts.

**User Account Information**

Device Manager supports the following sources of user account information:

- LDAP directory. You can configure Device Manager to read an LDAP-compliant directory, such as Active Directory to import groups, user accounts, and related properties.
  
  **Note:** Device Manager retains the source of user accounts. As a result, certain operations are not permitted on user accounts that you source from LDAP directories.

- Manual entry. You can use group maintenance forms in Device Manager to quickly create user accounts.

- Importing a provisioning file. You can develop a file outside of Device Manager containing user accounts and properties and then import the file. Device Manager automatically creates objects and sets properties values.

User accounts appear in the user table within the main display area of the Users tab. The table depicts each user account associated with the group that you select in the Group pane. The User toolbar provides available tasks to perform on user accounts. You can manipulate the table appearance.

The groups in which a user account is a member appear in the Groups column. Note that multiple groups appear as a multi-line entry. User accounts also appear in the Devices table. The user associated with a particular device appears in the User column. The user account shown in the User column represents the user that enrolled on that device.

**Group Information**

The group structure in Device Manager is flexible. Users may belong to multiple groups, groups may be nested inside of other groups, and the number of groups is not limited. You can create permanent or ad-hoc groups to suit any purpose. Device Manager supports the following sources of group information:

- LDAP directory. You can configure Device Manager to read an LDAP-compliant directory, such as Active Directory to import groups, user accounts, and related properties.
Manual entry. You can use group maintenance forms in Device Manager to quickly create groups.

Groups appear in the Group pane, the area to the left on the Users tab. The pane depicts groups in a hierarchical arrangement with the number of members in each group given as a number in parenthesis after each group name. A default group is automatically created during Device Manager installation to serve as the top-level node for the group hierarchy; all other groups appear as children of this node. Groups imported from LDAP-compliant directories also appear in the group hierarchy, with the LDAP directory name as the primary node. The individual groups of the LDAP directory appear as children of the primary node.

Groups may be nested in the hierarchy without limit. Fully-qualified group names use periods as delimiters. For example, a group of name Corporate.Sales.SalesSupport.Admin implies a nesting model based on organizational structure.

**Note:** User accounts may exist at any level. Thus, on a parent node, the count of group members represents the user accounts associated with that discrete node, and not the sum of the accounts associated with the nodes children.

Groups also appear in the User table. The groups a user belongs to appear in the Groups column.
To add, edit, or delete user accounts

You manage user accounts in Device Manager User table toolbar or the context menu.

To add a user account

1. In the group pane, select a group of which the user account will be a member.

2. Click New user from the toolbar or context menu. The Create a new user window appears.

3. Type a unique name for the user and a password.

4. Select an entry from the Role drop-down list. For more information about roles, see User Accounts and Roles.

5. Optionally, on the Properties tab, set user account attributes.

To edit a user account

1. In the group pane, select the group of which the user account is a member.

2. Click the user account to edit and the click Update. The Update a user window appears.

3. Revise the user account data, then click Update to save the changes.

   Note: If you edit the properties of accounts that you source from an LDAP directory, you do not change data in the directory.

To delete a user account

1. In the group pane, select the group of which the user account is a member.

2. Click the user account to delete and click Delete on the toolbar and then click Yes to confirm the deletion.

   Important: You cannot undo this operation.

   Note: If you delete an account that you sourced from an LDAP directory, you only remove the account from the Device Manage database; you do not change the account information in the directory.
To add or delete groups

You manage groups from the Group pane toolbar or context menu. Device Manager does not have a group edit command, because the only accessible property of a group object is its name.

To add a group

1. Select the parent node of the group.
2. Click New group. The Create a new group window appears.
3. Type a name for the group and then click Create. The group name must be unique relative to its peers in the group hierarchy. In addition, groups may not be added to group nodes that you import from LDAP-compliant directories.

To delete a group

Deleting a group has no affect on user accounts. You can only remove user accounts by using the Delete User command.
1. Select the group to delete.
2. Click Delete.
3. Click Yes to confirm the operation and remove the group.

Important: You cannot undo this operation.
Managing Group Membership

To manage group membership for a user account in Device Manager, you can use the Manage icon on the toolbar (or from the context menu) to do the following:

- Assign or revoke membership to one or more groups simultaneously.
- Remove user account membership in a selected group.

Using the Manage tool is optional. User accounts that you source from an LDAP directory or provisioning file may already be assigned to groups as defined in the directory or file. Device Manager does not support group assignment maintenance in groups that are directory based. Similarly, manually created user accounts may be assigned to groups as part of their definition.

To assign or revoke membership in multiple groups simultaneously

1. Click a user account.
2. In the toolbar or context menu, click Manage.
3. Click Membership. The Manage users to groups window appears.
4. Select or clear the check boxes to assign or revoke membership.
5. Click Update to save the group assignments.

**Note:** The Manage users to groups window does not include groups that are represented in an LDAP-compliant directory.

To remove a user account from a group

You use the Remove from group tool to terminate an account membership in a currently selected group.

1. In the Group pane, select the group containing the user account whose membership will be terminated.
2. In the User pane, click a user account.
3. In the toolbar or context menu, click Manage.
4. Click Remove from group and then click Yes to confirm the removal.
User Accounts and Roles

You manage user accounts in Device Manager by using the following commands from the User table toolbar or context menu:

- New user. Add a user account to Device Manager.
- Update. Edit a user account.
- Manage. Maintain a user account’s membership in Device Manager groups, subject to certain limitations.
- Delete. Remove a user account from Device Manager.
- Import. Read a provisioning file containing user accounts or properties to automatically create user account objects and update their attributes.

To search for a user account, on the Users tab, you use the Search tool. Type a search string into the Search field and then click the search icon.

**Note:** Searches are not case-sensitive; search results display matching user accounts in a separate table that does not include a “currently selected group” in the Group pane. (That is, no groups are selected.)

User Roles in Device Manager

Device Manager implements four default user roles to logically separate access to system functions, as shown in the following table. The columns represent the roles and the rows represent the system functions.

Citrix recommends that you assign the Support role to Help desk staff who require the ability to implement remote control sessions on mobile devices.

<table>
<thead>
<tr>
<th>System function</th>
<th>Administrator</th>
<th>Support</th>
<th>Provisioning</th>
<th>User</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log into administration console</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use remote support application</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use device provisioning application</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Use a mobile device</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

You can use role-based access control (RBAC) to create new user roles with permissions to access specific system functions beyond the functions defined by the default roles as shown...
User Accounts and Roles

- To prevent some administrators from viewing or wiping the devices of specific users.
- To allow specific users to only run reports.
- To enable super users to have access to everything, including the ability to create and limit other user roles.

You can view details about users and groups, such as the dates you created and modified a user or group on the Reporting tab.
Role Based Access Controls (RBAC) Permissions

You can use role-based access control (RBAC) to create custom roles in Device Manager, beyond the default roles. Custom roles grant permissions to user accounts to target specific functionality within Device Manager.

For example, you can create roles to allow the following capabilities:

- To give limited access to devices for administrators whom you want to only perform basic device operations and run reports. After the administrator logs on to Device Manager, only the Devices and Reports tabs appear. When a user only has Report rights, then the Device tab will not appear for that user, but the About tab will display. The About tab also will by default display for users who have no other rights at all.

- To allow an administrator to view, add, locate, edit, and lock a device.

You can associate both user and groups with roles. For example, if you import Active Directory groups into Device Manager, you can apply fine-grained access control to the Active Directory groups.

The following table describes the list of features and accessibility you can associate with a role:

<table>
<thead>
<tr>
<th>Role</th>
<th>Functionality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super Admin</td>
<td>Access to all functionality within Device Manager (all functionality listed in this table).</td>
</tr>
</tbody>
</table>
| Authorised Access| Access to the Admin console and/or the Self Help Portal, as well as device access for remote support and remote support access:  
  - Admin Console Access  
  - Self Help Portal Access  
  - Device Access (when Remote Support is enabled)  
  - Remote Support |
| Dashboard        | Access to view all of the Device Manager Dashboard and the ability to customize the Dashboard. In order to perform actions in the Dashboard, however, such as send notification, wipe/selective wipe, revoke, locate, and so on, a user must be granted those specific permissions. Also, if a user is restricted from viewing specific groups, the devices that belong to users in those blocked groups will not appear in the Dashboard. |
### Devices

Access to the Devices tab and the ability to perform general device management tasks, such as connecting to iOS devices, importing devices, editing device properties, locating, locking/unlocking, revoking, wiping, and selectively wiping a device. Specific permissions include:

- Full wipe device
- Selective wipe device
- View locations - when selected, users can see location and locate/track device. Includes:
  - Locate device
  - Track device
  - Lock device
  - Unlock device
- Deploy to a Device - allows you to push a deployment package to a device.
- Edit device properties
- Notification to a device - gives you the ability to select a notification template, send ad-hoc notifications to a device or group of devices from the devices tab using email, SMS, or agent push notifications.
  - Add/Delete device
  - Devices import
  - Revoke device
  - View Software Inventory - when selected, user is allowed to view a device software inventory.

### Users

Ability create users and groups. Includes the following permissions:

- Add/delete groups
- Add/delete users
- Edit a user's property
- Can manage admin users
- Users import - ability to import list of users from a file
## Role Based Access Controls (RBAC) Permissions

| Enrollment                          | Access to the Options dialog all functionality related to enrollment, including setting default enrollment modes, configuring enrollment notification servers (SMTP/SMS Gateway), modifying and creating enrollment templates, and sending enrollment notifications. Includes the following permissions:  
|                                   |   · Edit enrollment  
|                                   |   · Notify user  |
| Policies                          | Access to the Policies tab and all features related to defining and implementing policies, such as security and password policies, Exchange ActiveSync policies, app tunneling (Windows and Android), server groups, registry configurations (Windows), configurations, applications access (blacklist/whitelist), Sharepoint policies, and more. Includes the following permissions:  
|                                   |   · Add/delete policy  
|                                   |   · Edit policy  
|                                   |   · Download policies  
|                                   |   · Apply policies (deploy polices in a deployment package)  |
| Files                             | Access to the Files tab and adding, deleting, and downloading files. Includes the following permissions:  
|                                   |   · Add/delete files  
|                                   |   · Edit files  
|                                   |   · Download files  |
| Applications                      | Allows access to the Applications tab, where you can upload and define applications and create application categories to organize the apps you want to deploy to users' devices. Includes the following permissions:  
|                                   |   · Add/delete applications  
|                                   |   · Edit applications  
|                                   |   · Application download  
|                                   |   · Manage category (create custom app categories for organization)  |
| **Deployment** | Access to the Deployment tab and all functionality related to device deployment, such as the ability to create, edit, deploy, and delete packages. Includes the following permissions:  
  · Add/delete package  
  · Edit package  
  · Deploy packages |
| **Reporting** | Access to the Reporting tab and the ability to run and view Device Manager reports. |
| **About** | Access to the About tab features:  
  · Edit and upload an APNS certificate  
  · Edit XenMobile MDM license  
  · Connections information - provides visibility into server related information, such as security parameters, JVM information, and system health. |
| **Options** | The Options feature provides a user access to the Options dialog box and the following features in the Options dialog box:  
  · Role-Based Access Control  
  · LDAP  
  · Mobile Service Provider  
  · ActiveSync Gateway  
  · Network Access Control  
  · AppC WebServices API  
  · GoToAssist  
  · PKI Entity  
  · Scheduling  
  · Security  
  · General service parameters |

**Note:** If you want this role to have access to the Remote-Based Access Control feature, you need to specifically select the Remote-Based Access Control option in the dialog box.
<table>
<thead>
<tr>
<th>Role Based Access Controls (RBAC) Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Restrict Group Access</strong></td>
</tr>
<tr>
<td>Allows you to associate groups with the current role. When a group is associated with a role, users in that group can only see devices associated with that group. If a user belongs to more than one group, and some of those groups provide a range or permissions, all permissions related to all groups are merged into the role.</td>
</tr>
</tbody>
</table>
Configuring Custom Roles with RBAC

You can use the Role-Based Access Control feature in Device Manager to do the following:

- Create a new access control role (associate actions with roles)
- Add groups to a role
- Associate users with roles

To access the feature, in Device Manager, click Options in the upper-right corner, and then click Role Based Access Control.

To create a new access control role

You need to create an access control role in order to enable role-based access control in Device Manager.

1. In the Role Based Access Control panel, click New.
2. In the Create an admin role dialog box, enter a name for the role.
3. Select the features you want to enable for the role and then click Create.

To add groups to a role

When you create a new role, you can also associate a user group with the role as part of the role definition.

1. In the Role Based Access Control panel, select a role and then click Edit.
2. In the Role dialog box, in the Permissions list, select the feature access you want to associate with a role.
3. Under Restrict Group Access, select the group you want to have access to the role, and then click Save. The group you select and the users in the group users receive access to the features you choose.

To associate users with a role

After you create a new role, you can associate users with the role.

1. In Device Manager, click the Users tab and then in the User table, double-click a user or click New User.
2. In the New User dialog box, enter the user name and password.

3. In the Role list, click the role you want to associate with a user and then click Create.
## Provisioning File Formats

A provisioning file that you create manually and use to import user accounts and properties to Device Manager needs to have the following format:

For a user provisioning file of a .csv file type, the field separator is the ‘;’. The fields are the following:

```
user;password;role;group1;group2
```

**Note:** Because ';' is used as the separator character, it needs to be escaped if present in string values -> ';if';

An example of a user provisioning file content is as follows:

```
user01;pwd\;01;USER;myGroup.users01;myGroup.users02;myGroup.users.users01
```

in which:

- **User:** user01
- **Password:** pwd;01
- **Role:** USER

  **Note:** Role can only be one of the following: USER, ADMIN, SUPPORT, or DEVICE_PROVISIONING .

- **Groups:**
  - myGroup.users02
  - myGroup.users02
  - myGroup.users.users01

  **Note:** The '.' character is used as a separator to create group hierarchy, and so this character is forbidden in the groups name.

An example of the file format to provision user attributes is as follows:

```
user;propertyName1;properyValue1;propertyName2;properyValue2
```

**Note:** Because ';' is used as the separator character, it needs to be escaped if present in string values -> '\;';

An example of a user attributes provisioning file is as follows:

```
user01;propertyN;propertyV\;test\;1\;2;prop 2;prop2 value
```

in which:
Provisioning File Formats

- User: user01
- Property 1:
  - name: propertyN
  - value: propertyV;test;1;2

  **Note:** Property attributes must be lower case. The database is case-sensitive
- Property 2:
  - name: prop 2
  - value: prop2 value
To import user accounts and properties from a file

You can import user accounts and properties from a specially developed file called a provisioning file, which you can create manually.

**Note:** If you are importing users from an LDAP directory, use the domain name along with the user name in the import file. For example, specify username@domain.com. This syntax prevents additional lookups that will slow the import speed. If importing users to the Device Manager internal user directory, disable the default domain in order to speed up the import process. You can reenable the default domain after the import of internal users completes.

After a provisioning file is prepared, use the Import icon on the toolbar to read the file by following this procedure:

1. From the Users tab toolbar, click Import. The Import a provisioning file window appears.

2. In Provisioning file type, click Users or User Properties. If you click User Properties, you do not create an account.

3. In Provisioning file location, browse to the location of the file and then click Import.
Creating Policies

You create and configure Device Management policies on the Policies tab in the web console that you can push or made available to devices. You need to put the policies in a package for further deployment.

The way you configure the devices depends on the device operating system. To create a policy, on the left-hand menu, select the desired configuration option for a given platform. Then, click New to create the new policies or configurations.
Application Tunnels

Device Manager Application Tunnels (App Tunnels) are designed to increase service continuity and data transfer reliability for your mobile apps. App Tunnels are used to define proxy parameters between the client component of any mobile device app and the application server component.

Device Manager tunneling acts as a stream buffer to overcome inherent network issues, such as irregular latency or network hopping. Tunneling also provides checkpoint restart capabilities, which is critical when bouncing between cellular data points. Furthermore, Device Manager automatically applies on the fly data compression and AES encryption to all data traffic transiting within each tunnel.

You can assign a tunnel channel dedicated to each mobile app and monitor the apps. For each App Tunnel you define, Device Manager transmits and monitors the data streams in a separate tunnel.

App Tunnels provide the following benefits:

- Security through encryption of data traffic.
- Efficiency through compression of data traffic (can help reduce strain on your device data plan as well as battery usage).
- Reliability through buffering of data traffic. For example, if a device loses connectivity or switches from WiFi to 3G, App Tunnels make sure data traffic is buffered until the connection is reestablished.

**Note:** Each application requires its own tunnel.

This section includes procedures for creating App Tunnels in Device Manager and for creating a remote support App Tunnel specifically for the Remote Support help desk application.
To add an Application Tunnel

1. In the Device Manager web console, click the Policies tab and then under MDM Policies, click the device type for which you want to add an Application Tunnel (App Tunnel).

2. Click Tunnels and then click New tunnel.

3. In the Create a tunnel dialog box, in Name, enter the tunnel name. Citrix recommends the format Application_Name.

4. Select the Remote Support check box if the tunnel will be used for the Remote Support application. If you select this option, some of the options in the dialog box become unavailable. To complete the remote support tunnel configuration, see To create a remote support App Tunnel.

5. Under Connection configuration, in Connection initiated by, click Device if the connection is client-initiated or click Server if the connection is server-initiated. With the exception of Remote Support, App Tunnels are typically client-initiated.

6. In Protocol, click Generic TCP or Active FTP as the tunnel protocol.

7. In Max. connections per device, set the maximum connections, per device, per tunnel. (1 is recommended.)

8. Optionally, set the connection timeout, in seconds. This option allows for App Tunnels to be closed cleanly, even if the app fails.

9. Optionally, choose to use SSL encryption connection between the server running Device Manager and the desktop running the Remote Support application.

10. Optionally, in Secure Connection, select the Use SSL connection check box to block the traffic through that tunnel when the devices are in a roaming situation.

11. Under Application device parameters, click one of the following options to define the mobile application traffic redirection:

   - Through application settings. If you choose this option, you must set 127.0.0.1 in the application server field on the mobile device.

   - Using a local alias. The application on the mobile device will connect to the alias you enter; the alias will be resolved to localhost and intercepted by Device Manager Client Agent. An alias can be any name; for example: my_crm application, exchange server, and so on.

   - An IP address range. Specify a range of IP address targets for which the mobile application will try to connect to in order to make Device Manager to tunnel the connection. For example:

     - From: 0.0.0.0 to 255.255.255.255. In that case, all the traffic from the mobile device is redirected through Device Manager.
To add an Application Tunnel

- From: 88.10.10.10 to 88.10.10.10. In that case, only the traffic toward 88.10.10.10 is redirected through Device Manager.

12. In Client port, enter the port used by the application on the mobile device. This option is required.

13. In Application server parameters, enter the application IP address or server name, and the server port number. These options are required. In most cases, this is the same value as for Client port.

14. Click Create.

**Note:** To properly use an App Tunnel, you need to configure the device-based apps to connect to the Device Manager server rather than to their own server. Usually, 127.0.0.1 (localhost) is specified as the server address. However, some apps may not allow this type of configuration, or it may be preferable not to change the configuration of applications already deployed. In such cases, check the Specify a local alias box and enter the server's name. This name will be redirected automatically to 127.0.0.1 on the mobile devices.
To update or delete an App Tunnel

You can change the configuration settings of an existing tunnel in Device Manager but you cannot change the name of the tunnel.

1. In the Device Manager web console, click the Policies tab and then under MDM Policies, click the device type for which you want to update or delete the app tunnel.

2. In the list of tunnels in the center pane, select the check box for the tunnel you want to edit or delete.

3. Click Edit to change the settings or click Delete to remove the App Tunnel.

4. In the Edit a tunnel dialog box, change the settings and then click Update.
To create a remote support App Tunnel

You need to create a remote support Application Tunnel (App Tunnel) to support the Remote Support help desk application, which allows for the remote control of mobile devices over-the-air through Device Manager.

1. In the Device Manager web console, click the Policies tab and then under MDM Policies, click to expand the device type for which you want to configure a remote support App Tunnel.

2. Click Tunnels and then click New tunnel.

3. In the Create a tunnel dialog box, in Name, enter a name for the remote support app tunnel.


5. Optionally, under Connection configuration, in Connection time-out, select the Define check box and then enter a value in seconds to indicate the interval in which the connection to the Remote Support application should time out.

6. In Secure Connection, select the Use SSL connection check box if you want to configure a secure connection between the Device Manager server and the Remote Support application.

7. In While roaming, select the Block cellular connections passing by check box if you want to block the tunnel while roaming.

   **Note:** WiFi and USB connections are not blocked.

8. Click Create.
To create a software inventory package

A software inventory policy in Device Manager enables you to check all the applications and software packages installed on a device. A software inventory policy exists inside of a Device Manager Deployment Package. You can deploy the policy to any user group for any device platform.

1. Click the Deployments tab and then click New Package > New <platform> package.

2. On the Package Name page of the Create New Package wizard, enter a name for the app removal policy and then click Next.

3. On the Groups of users page, select the users from whose devices you want to remove the app and then click Next.

4. On the Resources to be deployed page, in Available Resources, select the Software Inventory policy, click the right arrow to move Software Inventory into the Resources to Deploy column and then click Next.

5. On the Deployment schedule page, configure the package to push the app Now or at a specified time in the future and then click Next.

6. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment and then click Next.

7. On the Package summary page, review the app removal package configuration and then click Finish.

To deploy the package and remove the app, select the package and then click Deploy. Connected devices receive the package as soon as scheduling rules are met.
Configuring Applications Access Policies

When you deploy a software inventory package to a device, Device Manager maintains the list of apps. You can work from those lists to configure Applications Access Policies, also known as application blacklists and whitelists to manage your users’ access to applications on their devices.

You can also use the Applications Access Policies in the following ways:

- As triggers for Automated Actions. For example, if Device Manager detects that a device has an unapproved app installed, you can configure an Automated Action that remotely wipes a device, or sends a notification to the user that the user’s device is out of compliance with the organization’s policy.

- To serve as device status flags for the Secure Mobile Gateway rules. For example, if Device Manager detects that a device has an unapproved app installed, you can configure the Secure Mobile Gateway rules to block the device from receiving email from the organization. For more information, see Secure Mobile Gateway Policies and Rules.

Applications Access Policies Types

You can create the following types of Applications Access Policies.

- **Forbidden** (blacklist). A list of apps that users cannot install on their devices. If even one app on device matches an app in the Forbidden list in Device Manager, the device is considered to be in violation of the policy.

- **Suggested** (whitelist). A list of apps that you suggest to users. Users can have one or more of the apps from the list installed and still be in compliance with the policy. However, if users install an app that is not listed in the policy, the user’s device is in violation of the policy.

- **Required** (whitelist). A list of apps that must be installed on the device to be in compliance with the policy. Users must install all of the apps on the list. If users do not install any of the apps in the list, the device is in violation of the policy.

App Definitions

You have the option in Device Manager of using the App bundle ID and App package name when you define iOS and Android apps in your policies. Device Manager can identify apps more reliably, however, when you use these values.

In iOS, an App bundle ID is traditionally a reverse-domain-name style string used when a developer creates a new app. For example, for Angry Birds (www.rovio.com/), the App bundle ID on iOS is ‘com.rovio.angrybirds’. On Android, an App package naming convention is similar to iOS, in which the developer identifies the app with a reverse-domain-name style string. The last part of the name is the name of the App package, often with the file
To configure an Applications Access Policy

1. In the Device Manager web console, click the Policies tab.


3. Click New Applications Access Policy.

4. In the Add a new Applications Access Policy dialog box, enter a name for the policy, such as, Forbidden iOS Apps and then optionally enter a description.

5. In Access policy, click one of the following options:
   · Required (whitelist). Defines a list of apps that users are required to install on their device to be in compliance with the policy. If any of the apps is not installed, the device is in violation of the policy.
   · Suggested (whitelist). Defines a list of apps that are suggested to users. Users can have one or more of the apps from the list installed and still be in compliance with the policy. However, if the user installs any apps that are not listed in the policy, the device is in violation of the policy.
   · Forbidden (blacklist). Defines a list of apps that users should not install on their devices. If any apps on device match an app in the this list, the device is in violation of the policy.

6. In OS type, select the device platform you want to associate with the policy.

7. Click New app.

8. In the Add a new application dialog box, enter the name of an app that you would like to add to the Applications Access Policy list. When you add an app, you can optionally enter the app bundle ID and app package name for iOS and Android. If you configure these fields, Device Manager uses the values to identify the app.

9. Click Create. This will create the application in the list. The app appears in the list in the Add a new application dialog box.

10. Click Create again to create the Application Access Policy. Once created, you can add this policy to a deployment package and deploy to the devices you want to manage.
Creating Encrypted Email Attachment Policies

You can configure secure email attachment policies that both define the types of file attachments should be encrypted when they pass through your corporate Exchange server, and that control the types of actions users can perform on the attachments when they open the files on their devices.

- Email Attachment Control Policies define the operations a user can perform that on an encrypted email attachment that is opened on a managed device, such as whether or not a user can print the document, copy and paste from it, forward the document in an email, open it in another app, and more.

- An Email Attachment Encryption Policy allows you to define the file types you want to encrypt that pass though your corporate Exchange server, to customize a subject header for each encrypted email attachment, and to add any file type exceptions. You can only create one email attachment encryption policy.
To configure a secure email attachment document control policy

When you configure a secure email attachment document control policy, you define the types of operations your users can perform on the email attachments sent through your XenMobile secured email. When an email attachment is encrypted, the attachment can only unencrypted and read recipient of the email on the device and within in the secure document container. If a user forwards the email with the attachment, the attachment is encrypted so that the file cannot be viewed by anyone else. Once the document is opened on the device, the control polices you have configured determine the types of actions a user can perform on the file.

1. In the Device Manager web console, select the Policies tab.
2. On the left side of the console under App Policies, select Email Attachments > Control Policy.
3. From the menu bar, click New Control Policy.
4. In the Create a Control Policy dialog box, enter a name for the policy.
5. Under Control Policy Setting, select the options you want to enable for the policy. If you select an option, that means a user will be able to perform the action on email attachments viewed on their device. Any options left unchecked means the user will not be able to perform those actions on email attachments view on their device. Options include:
   a. Select Wipe on Jailbreak if you want the document to be deleted if the user Jailbreaks or roots the device.
   b. Select Copy/Paste of content. if you want users to be able to copy/paste document content.
   c. Select Copy/Paste of content. if you want users to be able to copy/paste document content.
   d. Select Email document if you want to allow users to email the document.
   e. Select Print Document if you want to allow printing of the document.
   f. Select Open in Another App if you want to allow the document to be opened in another app on the device.
   g. Select Upload to Shared Docs if you want to allow the document to be uploaded to the Shared Docs folder in the Connect app.
   h. Select Annotate Local Docs is you want to allow the user to annotate the doc.
6. Click Create.
To create an email attachment encryption policy

An email attachment encryption policy allows you to define the types of files you want to be encrypted that pass through your corporate Exchange server, add a custom header message to the emails, and add any exceptions for file types you do not want to encrypt.

1. In the Device Manager web console, select the Policies tab.

2. On the left side of the console under App Policies, select Email Attachments > Encryption Settings.

3. Click Create encryption settings. (If the policy already exists and you want to modify it, select it and click Edit.)

4. In the Create Email Attachment Encryption Settings dialog box, on the General tab enter a name and description.

5. Select the Files Types tab and select the types of file attachments that you want to be encrypted. If you select an option, then that file type will be encrypted if sent as an attachment through your Exchange server.

6. On the Subject Append tab, select if you want to add a custom message for the subject line to be added to emails that contain encrypted attachments. Also, enter the message that you want to appear.

7. On the Exceptions tab, enter the file extensions for the types of files that you do not want to be encrypted. For example, to exclude Microsoft Word files from being encrypted, enter .doc on a single line. The entries may contain wildcard expression using the asterisk (*) character. For example, Filenames matching these expressions will NOT be encrypted.

8. Click Create.
Managing iOS Configurations

You can create a variety of policy types and configurations for your iOS devices to help manage user and company data security, including passcode policies, general iOS restrictions policies, App Tunnel configuration policies so your users can securely access your company intranet, email policies so your users can seamlessly connect to corporate email accounts, app distribution policies so you can make useful apps available to your users, app removal policies to revoke unauthorized or out of date apps, and much more.
To configure iOS security policies

The following procedures describe how to configure three basic security policies in Device Manager for iOS users: passcode policies, general restrictions policies, and an App Tunnel policy to enable users to securely access their organization's intranet.
To create an iOS passcode policy

1. On the Policies tab, under iOS, click Configurations.

2. In the New Configuration menu, click Profiles and Settings > Passcode.

3. In Passcode, on the General tab, enter a name for the policy and then configure the policy removal settings.

4. On the Policy tab, configure your iOS passcode policy according to the standards of your IT department. The passcode policy options are as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Require a code on the device</td>
<td>Enables passcode protection on the device. If cleared, the device does not require a passcode on the device (unless the device user sets it manually).</td>
</tr>
<tr>
<td>Allow simple values</td>
<td>Allows the use of a simple passcode, which is defined as a passcode containing repeated characters, or increasing (bottom up) or decreasing (top down) characters (such as 123 or CBA).</td>
</tr>
<tr>
<td>Require alphanumeric values</td>
<td>Requires that at least one character of the passcode is a letter.</td>
</tr>
<tr>
<td>Minimum length codes</td>
<td>Allows you to set the minimum overall length (in characters) required for the passcode.</td>
</tr>
<tr>
<td>Allowed minimum non-alphanumeric characters</td>
<td>Allows you to set the minimum amount of numerical characters required of the passcode.</td>
</tr>
<tr>
<td>Maximum passcode age (1-730 days, or none)</td>
<td>Allows you to specify the number of days for which the passcode can remain unchanged. After the set number of days, the user is forced to change the passcode before the device is unlocked.</td>
</tr>
<tr>
<td>Auto lock (1-5, 10 or 15 minutes or none)</td>
<td>Allows you to specify the number of minutes for which the device can be idle (without being unlocked by the user) before it gets locked by the system. When this limit is reached, the device is locked and the user must enter the passcode.</td>
</tr>
<tr>
<td>Codes History (1 to 50 codes or none)</td>
<td>Allows you to specify that when the user changes the passcode, it has to be unique within the last N number of entries in the history.</td>
</tr>
</tbody>
</table>
To configure iOS security policies

<table>
<thead>
<tr>
<th>Grace period before device lock</th>
<th>Allows you to set the maximum grace period, in minutes, to unlock the phone without entering a passcode. Default is 0, (no grace period), which requires a passcode immediately.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum failed attempts</td>
<td>Allows you to specify the number of allowed failed attempts to enter the passcode at the device's lock screen. When this number is exceeded, the device is locked and must be connected to its designated iTunes in order to be unlocked.</td>
</tr>
</tbody>
</table>

5. Click Create. The new policy appears in the Policies list.

To create an iOS restrictions policy

iOS restrictions enable you to control a wide variety of options and actions on an iOS device, such as restricting voice dialing and camera usage, limiting Siri usage and You Tube, document syncing to iCloud, and more. Where indicated, some features are only supported on iOS 6 and require that the device to be placed into Supervised mode with the Apple Configurator.

1. On the Policies tab, under iOS, click Configurations.

2. In the New Configuration menu, click Profiles and Settings > Restrictions

3. In Restrictions Configuration Creation, on the General tab, enter a name for the policy.

4. On the Restrictions tab, enter the following information:

   Tip: Any option for which you select Allow means that the user can perform the operation or use the feature. For example:

   · Allow installing apps. If selected, the App Store is enabled and its icon is available from the Home screen. Users can install or update their applications. If cleared, the App Store is disabled and its icon is removed from the Home screen. Users cannot install or update their applications.

   · Allow use of camera. If selected, the user will be able to use the camera on their iOS device. If cleared, the user cannot use the camera on their iOS device.

   Note: The following restrictions work only on iOS 6 devices and require that the device is placed in Supervised mode with the Apple Configurator.

   · Allow Game Center. If selected, the Game Center will be available on the iOS 6 device. If cleared, the Game Center is inaccessible and the icon removed from the device desktop.

   · Allow UI configuration profile installation. If selected, users can install configuration profiles and certificates on their devices.

   · Allow iMessage. If selected, users can use iMessage.

5. On the Applications tab, enter restrictions you would like to set for default iOS apps.
To configure iOS security policies

6. On the iCloud tab, enter restrictions you want to configure for Apple iCloud.

7. On the Security tab, enter any security restrictions.

8. On the Ratings tab, enter settings for the type of content you want to allow on iOS devices.

9. Click Create. The new policy appears in the Policies list.

To create an iOS app tunnel

When you create an app tunnel, you ensure that all data flows securely for your applications through a server you specify, such as a server in the internal network. App tunnels are used to define proxy parameters between the client component of any mobile device applications and the application server component.

1. On the Policies tab, under iOS, click Configurations.

2. Under iOS, click the tunnels icon and then click New Tunnel.

3. In the New Tunnel dialog box, enter the following information:
   · Name. Enter a name for the tunnel.
   · Maximum connections, per device.
   · Connection timeout. Allows for app tunnels to be closed cleanly, even if the application fails to do so.
   · Client port. Enter the port used by the application on the mobile device.
   · Application server parameters. Enter an IP address or application server host name.
   · Server port. Enter a server port number. In most cases, this is the same value as for the Client port.

4. Click Create.
To configure an Exchange policy on iOS

**Important:** In this release of Device Manager, Citrix strongly recommends that you do not create and deploy more than one email attachment encryption policy, or conflicts will occur for users. If you deploy more than one policy and the policies are not identical, the rules of the policy will conflict with users to whom you have deployed either one of the policies.
To configure an Exchange policy on iOS

To configure an Exchange ActiveSync policy

1. On the Policies tab, under iOS, click Configurations.

2. On the New Configuration menu, click Profiles and Settings > Exchange ActiveSync.

3. On the General tab, enter a name and description for the policy.

4. On the Exchange ActiveSync tab, enter the Exchange ActiveSync configuration details for your organization as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Exchange ActiveSync account name</strong></td>
<td>Name of the Exchange account.</td>
</tr>
<tr>
<td><strong>Exchange ActiveSync host</strong></td>
<td>The Exchange server host name (or IP address).</td>
</tr>
<tr>
<td></td>
<td>Use SSL. Optional. Specifies whether the Exchange server uses Secure Sockets Layer (SSL) for authentication.</td>
</tr>
<tr>
<td><strong>Domain</strong></td>
<td>Domain under which the Exchange server resides.</td>
</tr>
<tr>
<td><strong>User</strong></td>
<td>The user name for the Exchange account.</td>
</tr>
<tr>
<td><strong>Email address</strong></td>
<td>Specifies the full email address for the account.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> For the preceding user name and email address, Citrix used Device Manager system macros, ${user.username} and ${user.mail}, which will automatically look up specific users and their email accounts based on the format listed.</td>
</tr>
<tr>
<td><strong>Password</strong></td>
<td>Optional. The password of the account. Use only with encrypted profiles.</td>
</tr>
<tr>
<td><strong>Mail last days to synchronize</strong></td>
<td>The time the range in the past from which to synchronize the email account.</td>
</tr>
<tr>
<td><strong>Identity credential (Keystore or MsCertSrv)</strong></td>
<td>Optional. An identity credential type.</td>
</tr>
</tbody>
</table>

5. On the Exchange ActiveSync Policy - Policy tab, enter the following information:

   - Authorize email move between accounts. Select if you want to authorize moving emails between two or more accounts. (Available only in iOS 5 and iOS 6.)

   - Sending email only from Mail application. Select if you want to enforce that emails from the account can only be sent from the mail application (and no other mail clients). (Available only in iOS 5 and iOS 6.)

   - Disable mail Recents syncing. Select if you want to prevent recently used email addresses from being synced with other devices through iCloud. (Available only in
To configure an Exchange policy on iOS

- Enable S/MIME. Select if you want your email server to use S/MIME public key encryption.

6. Click Create.
To create an iOS App Tunnel

Application Tunnels (App Tunnels) ensure that all data flows securely from applications to a server you specify, such as an intranet server within the organization. The App Tunnels only create connections to the specific resources used by the configured application. You use App Tunnels to define proxy parameters between the client component of a mobile device application and the application server component.

1. In the Device Manager web console, select the Policies tab.

2. Click to expand iOS,, click Tunnels and then click New tunnel.

3. In the Create a tunnel dialog box, enter the following information:

   a. Name. Enter a name for the tunnel.

   b. Max. connections per device. Enter the number of connections you want the tunnel to allow. (1 is recommended.)

   c. Connection time-out. Select the Define check box to allow for app tunnels to be closed cleanly, even if the application fails to do so.

   d. IP address or server name. Enter the IP address or application server host name.

   e. Server port. Enter the server port number.

4. Click Create.
To configure network policies for iOS

You can configure WiFi and VPN policies in Device Manager for iOS devices so your users can securely access the internal network in your organization and use resources wirelessly.

To create an iOS WiFi policy

1. On the Policies tab, under iOS, click Configurations.
2. On the New Configuration menu click Profiles and Settings > WiFi.
3. In the WiFi Configuration dialog box, on the General tab, enter a name for the policy.
4. On the WiFi tab, enter the WiFi SSID and encryption type.
5. On the Protocols tab, enter the protocol type used by your the WiFi in your organization.
6. On the Authentication tab, enter the user name and password to be used to log on to the wireless network in your organization.
7. On the Trust tab, if you are using custom trusted certificates for authentication, upload the certificates.
8. On the Proxy tab, if you are using a proxy server to route traffic through the WiFi, configure the proxy settings.
9. Click Create.

To create an iOS VPN policy

1. On the Policies tab, under iOS, click Configurations.
2. On the New Configuration menu click Profiles and Settings > VPN.
3. In the VPN Configuration dialog box, on the General tab, enter a name for the policy.
4. On the VPN tab, enter the VPN configuration for your company VPN network.
5. On the Proxy tab, if your VPN network uses a proxy server, configure the proxy server.
6. Click Create.
To push applications to iOS devices

You can push both external Apple iOS apps or apps that you developed internally within your organization to devices from Device Manager. You can select apps you want to push from the Files tab and then deploy the apps in a package.

Device Manager provides two security policy settings that you can apply to applications before you push them:

- Prevent App Backup. To ensure that certain external apps are only installed on devices that you manage, you can choose to remove a pushed app on an iOS device if the user removes their Device Manager profile.

- Remove App When MDM Profile is Removed. Before you push an iOS 5 app to an iOS device, you can select the Prevent Backup of App Data setting, which will prevent the user from backing up a specified app either on the user's computer (via iTunes) or through iCloud.

Note: Before you can push an iOS app to a device, the app file must already have been imported into Device Manager.

1. On the Files tab, select either an internal or external app and then click Edit or double-click the app.

2. In the app properties dialog box, select:
   - If you want the app to be removed if the user deletes the device's Device Manager profile.
   - If you want to prevent the app from being backed up by the user.

3. Click Update.


5. On the Package Name page of the Create New Package wizard, enter a name for the iOS app and then click Next.

6. On the Groups of users page, select the users whom you want to receive the app on their devices when you push the app and then click Next.

7. On the Resources to be deployed page, in Available Resources, select the app you want to push, click the right arrow to add the app to the package and then click Next.

8. On the Deployment schedule page, configure to push the app Now or at a specified time in the future and then click Next.

9. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment and then click Next.

10. On the Package summary page, review the package deployment configuration and then click Finish.
11. To deploy the package, select the package and then click Deploy. Connected devices receive the package as soon as scheduling rules are met. Reconnecting devices receive the package as they reconnect, subject to other rule criteria. When you push external apps to a store, the user is prompted to enter the user’s Apple ID credentials in order to install the app.
Citrix Worx Store for iOS Apps (MDM-only)

If you are using XenMobile MDM edition (not the Enterprise solution), you can deploy apps to your users' iOS devices using the Citrix Worx Store for iOS, preconfigured in XenMobile Device Manager as an iOS web clip, and included as part of the iOS base package that gets deployed when a user enrolls into Device Manager. For more information about the iOS base package, see iOS base package.

When you install Device Manager for the first time, you need to add iOS apps to Device Manager, add apps to the Worx Store Deployment Package and then deploy the package to users.
To create an app store for iOS devices

1. In the Device Manager web console, click the Policies tab, click to expand iOS and then click Configurations.

2. Click New Configuration, point to Profiles and Settings and then click Web Clips.

3. In the Creating a web clip configuration dialog box, enter the following information:
   - In Identifier, type MyAppStore
   - In Display name, type MDM App Store
   - In Organization, type Citrix
   - In Description, type a description.
   - In Allow profile removal operation, click an option to determine whether or not you want the user to be able to remove the profile from the device. The Always options means that the user can always remove the profile, without have to provide user credentials. The Authentication options means that in order to remove the profile, the user has to provide credentials. The Never option means the user can remove the profile at any time, without having to provide credentials.
   - Select the Automatic removal date check box if you want to specify an exact date on which you want Device Manager to remove the profile.
   - Select the Duration until removal (in days) check box and then enter a period of time after which Device Manager removes the profile.

4. Next, select the Web clip tab and enter the following information:
   - Label. Type MDM App Store
   - URL. Type `${webeas-url}`

5. Click Create.

After you create the app store, you need to add it to a deployment package, add apps to the deployment package, and then deploy the package to iOS devices.
To create a deployment package with the Self-Serve app store for iOS devices

If you are upgrading to XenMobile 8.0.1 and you had deployed an Enterprise App Store to iOS users, as part of the upgrade, you need to create a deployment package with the new Self-Serve app store. The resource name for the Self-serve app store in Device Manager is MyAppStore. Next, you add apps to the store and then you deploy the package to iOS devices.

In order to add iOS apps to the Self-Serve app store, you must first add the apps to Device Manager on the Applications tab.

1. In the Device Manager web console, click the Deployment tab, and then click New package > New iOS package.

2. In the Create new package wizard, in the Package name window, enter a name for the package and then click Next.

3. In the Groups of users window, select one or more groups to whom you want to deploy the package and then click Next.

4. In the Resources to be deployed window, do the following:
   a. Under Available resources, click to expand Configurations, click MyAppStore, and then click the right arrow to add the resource to the Resources to deploy column.
   b. Under Available resources, click to expand Enterprise Application Store, click the apps you want to add to the Self-Serve app store and then click the right arrow to add the apps one by one to the Resources to deploy column.

5. Click Next

6. In the Deployment schedule window, select the If not deployed check box, click Now and then click Next.

7. In the Deployment rules page, click Next.

8. Click Finish.

9. To deploy the Self-Serve app store, on the Deployment tab, click Deploy.

To verify the deployment, check the device of an iOS user to whom you deployed the package and look for the Self-Serve app store on the device Home screen.
To add apps to Worx Store for iOS

To add apps to the Citrix Worx Store app store for iOS, you need to add iOS apps to Device Manager, and then add those apps to a deployment package and deploy that package to iOS users.

1. In the Device Manager web console, click the Deployment tab, and then click New Package > New iOS Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the Self-service package and then click Next.

3. In the Groups of users window, select the group you created earlier and then click Next.

4. Under Enterprise Application Store, select the apps you want to add to the app store and then click the right arrow to add them.

5. Click Next.

6. In the Deployment schedule window, select the If not deployed Start Now option and then click Next.

7. In the Deployment rules page, click Next.

8. Click Finish.

9. To deploy the Citrix Worx Store app store for iOS, from the Deployment tab, click Deploy. To verify, check the device os an iOS user you deployed the package to and look for the Self-serve app on the device Home screen.
Customizing the Citrix Worx App Store for iOS

You can customize your Citrix Worx app store for iOS by modifying its style sheet and a Javascript file on the server where you have installed Device Manager. Device Manager displays the web app store through the Tomcat web server, located in the following directory on the server running Device Manager:

`ZDM_HOME\tomcat\webapps\zdm\webeas\ios\assets`

You can make changes to specific files in the directory to modify the look and feel of your web enterprise app store. Changes to the app store will occur immediately and can be seen when the user refreshes or restarts the Safari browser on the user’s device. You can modify the following for your enterprise web app store:

- Background of the list and app details page. If you wanted to change the background image, you could easily change the images in the images sub-directory under iOS located here:

  `ZDM_HOME \tomcat\webapps\zdm\webeas\ios\assets\css`

  For example, in the CSS, you can change the page background by replacing the image referenced by “background.png” in the following example.

  ```
  body { background: rgb(197, 204, 211)
  url('../images/background.png') repeat; font-family: Helvetica, Arial, sans-serif;
  ```

  You can also add your own custom images to the image directory on the web server and reference the images in the CSS.

- Button label modifications. You can modify values in the file name webEAS-en.js, which is located under the js sub-directory here:

  `ZDM_HOME\tomcat\webapps\zdm\webeas\ios\assets\js`

  For example, open webEAS-en.js in a text editor. Then, change the header text to match your corporate identity, such as Acme Applications as shown in the following example.

  ```
  HeaderText: "Acme Applications",

  BackButtonText: "Back",

  HelpButtonText: "Help",

  HelpDialogTitle: "App Store Help",
  ```

- Page theme. You can change or add a new page theme by changing the name of the theme, shown as “default” in the following example.
To change the theme to aqua, you would modify the value as follows:

Theme: "aqua", //Available themes: default, aqua, redblack
To update new versions of custom iOS apps

A custom app is an app that is not available on iTunes. When a new version of a custom app is available, you can update the app by adding the new .ipa file to the Files tab in Device Manager. The next time the device connects to Device Manager, the app is updated to the new version.

1. On the Files tab, click the iOS app you want to update and then click Edit.

2. To upload an iOS application with a .ipa extension, click Choose File and then browse for the app.

3. Click Update. The new version of the app is pushed to the device the next time the device connects with Device Manager.
To remove an app from an iOS device

You can easily remove applications from iOS devices by creating an app removal policy and then deploying that policy to a device. An app removal policy is based upon specifying an application's app bundle ID in the policy. In order to determine an application's bundle ID, you must push a software inventory policy on to a device. Or, you can use the Apple's Xcode IDE application tool called Bundle identifier to find an app's bundle ID.

Note: Removing an application will only work for applications distributed by Device Manager.

1. On the Policies tab, click iOS > Configurations.


3. In the App Removal dialog box, select an App bundle ID for the app you want to remove.

4. Click Create. You created the new app removal policy.

5. To deploy the policy as a package, click the Deployment tab.

6. On the Package Name page of the Create New Package wizard, enter a name for the app removal policy and then click Next.

7. On the Groups of users page, select the users from whose devices you want to remove the app and then click Next.

8. On the Resources to be deployed page, in Available Resources, select the app removal policy you want to use for the package, click the right arrow button to add the resource to the package and then click Next.

9. On the Deployment schedule page, configure to push the app Now or at a specified time in the future and then click Next.

10. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment and then click Next.

11. On the Package summary page, review the app removal package configuration and then click Finish.

12. To deploy the package and remove the app, click the package and then click Deploy. Connected devices receive the package as soon as scheduling rules are met.
To create a deployment policy for iOS apps

1. Select the Deployment tab, click New Package and then click New iOS Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the package, such as iOS App Store, and then click Next.

3. On the Groups of users window, select the group you created earlier and then click Next.

4. On the Resources to be deployed window, under Available Resources, scroll to the Enterprise Application Store, select the apps you want to add, click the right arrow button and then click Next.

5. On the Deployment schedule window, select the If not deployed Start Now option and then click Next.


8. In the packages list, click Deploy.
To create an app lock policy for iOS 6 devices

App lock, or Kiosk Mode, enables you to set an iOS device to only run a single application. When pushed to a device, the policy runs only the app that is specified, disables the home button, returns the device to the specified application automatically upon wake or reboot.

**Note:** This feature works only on iOS 6 devices and requires that the device to be placed into Supervised mode with the Apple Configurator.

1. On the General tab, enter the Identifier of the policy, the display name, and a description.

2. In Allow profile removal operation, click on of the following options:
   - Always: This option allows the profile to always be removable.
   - Authentication: Allows you to enter a required password that is used when profile is removed. Requires a password
   - Never: Prevents the profile from ever being removed.

3. Select the Allows you to select a specific date check box to specify a date you want to remove the profile.

4. Select the Duration until removal (in days) check box to enable you to set a period of time after which the profile will automatically be removed.

5. On the App Lock - Configuration tab, enter the app bundle identifier of the app you want to use and then click Create.
To configure automatic profile removal for iOS 6 devices

For iOS 6 devices, you can configure automatic profile removal in Device Manager. You can configure profiles to be removed automatically at a specified date, to be removed manually by the user with password authentication, or never to be removed.

In the iOS 6 profile dialog (e.g., APN configuration creation), at the bottom of the General tab, you can configure the automatic profile removal settings:

1. Click New Configuration and then click Profiles and Settings.

2. Select a profile type; for example, APN.

3. In the iOS 6 profile dialog box, on the General tab, configure the automatic profile removal settings as follows:
   a. In Allow profile removal operation, select one of the following options:
      - Always. Allows the profile to always be removable.
      - Authentication. Allows you to enter a required password that is used when profile is removed. Requires a password.
      - Never. Prevents the profile from ever being removed.
   b. Select the Automatic Removal Date check box if you want to select a specific date on which to remove the profile.
   c. Select the Duration until removal (in days) check box to specify a set a period of time after which the profile will automatically be removed.

4. Click Create.
To configure geo-tracking on iOS devices

If the iOS devices you manage in Device Manager have a location services policy applied and you configure geo-tracking, you can view the locations of the device over the time you configured in the location services policy. Geo-tracking enables you to track an iOS device over periods of up to six hours at a time. You can view the geographical location of a device and its movement and you can view the device location on Google Maps. If you want to specify individual parameters for your GPS tracking (as opposed to just activating it), you need to deploy a geo-tracking policy. If you choose to keep the default values, you can enable tracking immediately.

1. Click the Policies tab and then click iOS > Configurations.

2. On the New Configuration menu, click Profiles and Settings and then click Location Services.

3. In the Locations Services - Configuration creation dialog box, enter the following information:
   - Name. Enter a name for the location services policy.
   - Description. Provide an optional description for the policy.
   - Location fix timeout. Enter the time Device Manager waits before timing out if the device location cannot be fixed. If nothing is set, Device Manager attempts to locate the device according the Poll interval you specify.
   - Tracking duration. Enter the period of time that the device will be tracked after an Enable Tracking command is sent to the device. The maximum setting is six hours.
   - Poll interval. Enter a value for how often Device Manager will attempt to fix a location on the device. If the device cannot be located, the attempt to locate the device will time out according to the Location fix timeout setting.
   - Accuracy. Set the accuracy of the location point from the device.
   - Report if location services are disabled. Select to enable the device to report that GPS is disabled to Device Manager and Device Manager will display the status of the device.

   **Note:** You must enable the setting if you are using Automated Actions to trigger an action based upon a location-based trigger, such as Location Perimeter Breach or Location Services Disabled. Also, the device must be contacted for its location to trigger the action.

4. Click Create.

5. To enable tracking of the device according to this configuration, you need to deploy the package to the devices you want to track. Next, deploy the package to the iOS device users to implement the policy.

6. To track the device, do one of the following:
To configure geo-tracking on iOS devices

- On the Devices tab, select the device, right click, and select Security -&gt; Enable Tracking.

- Select the device and from the Security button click Enable Tracking.

To view a device's geo-tracking

1. On the Devices tab, select the device you want to view and then click Edit.

2. Click the Geo-tracking tab. Each point on the map indicates when Device Manager fixed the location of the device. A green point indicates the first location point when tracking started. A red point indicates the last device location point captured before tracking ended. You can mouse over each point to see more detailed geographical information. To see a longer range of tracking points, for example, if the device was tracked several times, you change the Display Points From date range and then click Filter.
To create a geo-fencing policy for iOS devices

Geo-fencing in Device Manager allows you to define a geographic perimeter for an iOS device. You can then perform a selective or full wipe upon the breach of the perimeter you set. The policy also notifies Device Manager and the device user when the device has moved beyond the defined radius of the policy. You have the option of setting a delay before the device is wiped, which can give the user time to return to the allowed GPS location perimeter.

1. On the Policies tab, under iOS, click Configurations.

2. In New Configuration, click Profiles and Settings, and then click Location Services.

3. In the Location Services - Configuration Creation dialog box, on the General tab, enter a name for the new profile configuration and configure the following settings:
   - Name. Enter a name for the location services policy.
   - Description. Provide an optional description for the policy.
   - Location fix timeout. Enter the time Device Manager will wait before timing out if device location cannot be fixed. If nothing is set, Device Manager attempts to locate the device according the defined Poll interval.
   - Tracking duration. The period of time that the device will be tracked once an Enable Tracking command has been sent to the device. Maximum is six hours.
   - Poll interval. Enter a value for how often Device Manager will attempt to fix a location on the device. If the device cannot be located, the attempt to locate will time out according to the Location fix timeout setting above.
   - Accuracy. Enter the accuracy of the location point from the device.
   - Report if location services are disabled. Select if you want the device to report that GPS is disabled to Device Manager and the server will show the status of the device.

4. Click the Geo-fencing tab and then set the following parameters:
   - Radius. Select this option to define the radius of the geo-fence. The default value represents the smallest allowable radius for this feature, which is approximately 164 feet, or 50 meters. Enter a small value; for example, 150 feet.
   - Center Point Latitude. For example, 37.787454.
   - Center Point Longitude. For example, 122.402952.
   - Device Notification on perimeter breach. Select this option so that the device user is notified when the device has breached (has gone outside of) the defined perimeter radius.
To create a geo-fencing policy for iOS devices

- Delay on Wipe. Enter 2 minutes as the time allowed before the device becomes wiped of its corporate data and apps.

5. Click Create.

After you create the policy, you need to deploy it to your iOS devices. In the Devices tab, when you click Deploy, the following actions take place:

- All deployment packages targeting the device are deployed.
- Device inventory, properties, and usage data is refreshed.
To create a profile removal policy for iOS

If you want to remove an application profile for iOS from a user’s device, you need to create a profile removal policy in Device Manager and then deploy the policy to the device.

1. On the Policies tab, click iOS > Configurations.


3. In Add a profile removal operation, enter the Profile ID of the app profile. This is found in the profile’s General tab. You can find the profile ID on the profile General tab.

4. Click Create.

Deploy the policy to a device.
Distributing iOS Volume Purchase Program Apps

The Apple Volume Purchase Program allows you to purchase iOS apps and books in volume and distribute them to your employees, either as free apps or apps for purchase by using Device Manager.

Distributing apps purchased through the Volume Purchase Program requires following general steps:

- Purchase apps by using your Volume Purchase Program account on the Apple Volume Purchase Program website.
- Download the Volume Purchase Program app purchase license spreadsheet containing the app license redemption codes.
- Add the app information and import the license spreadsheet into the Device Manager file repository.
- Build an app package and deploy it to your users’ devices.

Before you begin, make sure you have the following information about the Volume Purchase Program apps you want to distribute with Device Manager:

- App Store Web address for each purchased Volume Purchase Program app you want to distribute.
- Licensing spreadsheet with redemption codes for the Volume Purchase Program apps that will import into Device Manager. You can import license spreadsheets one time, or multiple times if you purchase new licenses at a later date.

After you import Volume Purchase Program license codes, the codes are initially considered to be "Unused." Each code is reserved and will switch to "Pending" status during deployment as they are sent to devices. The device can determine the following:

- The code is not necessary (for example, the app was already purchased by the specified iTunes account), in which case the code status will be switched back to "Unused."
- The device can determine that the code was invalid (for example, the code was already used for a purchase), in which case the status switches to "Invalid."

**Note:** If an app installation fails because the code was invalid, the code is not sent to the device until the next deployment.

- The code is applied successfully to the purchase of the application, in which case the status goes to "Used."
To distribute Volume Purchase Program apps

1. Click the Files tab and then on the New menu, click New external iOS app.

2. In the new app dialog box, enter the Web address of the Volume Purchase Program iOS app. This is the Web address you used to purchase the app at the Apple app store.

3. Click Go Device Manager locates the Web address for the app and then populates the dialog box with the app details.

4. Click the VPP Licenses tab and then click Import a license spreadsheet.

   **Note:** You can also update Volume Purchase Program license redemption codes by adding a new license spreadsheets with new licenses at a later date. For example, if you first buy 100 licenses for an app and then eventually have more target devices to deploy to, you can purchase more licenses at the Apple Volume Purchase Program site and then add them to the existing Volume Purchase Program app definition in Device Manager.

5. Browse to the location on your system where you have saved the license spreadsheet and then click OK.

6. Click Add to complete the file configuration.

7. Next, you will create a deployment package so you can push the Volume Purchase Program apps to your users’ devices. Select the Deployment tab.


9. On the Package Name page of the Create New Package wizard, enter a name for the iOS Volume Purchase Program app and then click Next.

10. On the Groups of users page, select the users on whose devices you want to receive the Volume Purchase Program app when you push the app and then click Next.

11. On the Resources to be deployed page, in Available Resources, select the Volume Purchase Program app from the External iOS list, click the right arrow button to add the app to the package and then click Next.

   **Note:** With Volume Purchase Program app deployments, you can only choose to push these apps as External iOS apps, and not as Enterpise App Store apps.

12. On the Deployment schedule page, configure to push the app Now or at a specified time in the future and then click Next.

13. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment and then click Next.
To distribute Volume Purchase Program apps

14. On the Package summary page, review the iOS Volume Purchase Program app packaged deployment and then click Finish.

15. To deploy the iOS Volume Purchase Program app package, select the package and then click Deploy. Connected devices receive the package as soon as scheduling rules are met. Reconnecting devices receive the package as they reconnect subject to other rule criteria. When you push external apps to a store, the device user is prompted to enter their Apple ID credentials in order to install the app. You will see a message after installation that the user account on the device will not be charged for the app.

After you deploy the app, the purchased app becomes the property of the iTunes account entered on the device. The owner of the account used to install the app on the device can later install the application on the device of their choosing without having to pay for it.
To run a code inventory report on Volume Purchase Program apps

The Apple Volume Purchase Program code inventory report in Device Manager provides a detailed list of all of your Volume Purchase Program app purchases, the number of purchased licenses for each app, whether or not the license is being used by a device, the associated device ID, and more.

- Click the Reporting tab and then click AVPP code inventory to run the report.

  The code summary report results appear in a new tab.
To store iOS user password

If you want to ensure that iOS users have their passwords stored for ongoing authentication, even if they log out of the Connect agent, you can configure that setting in the Options dialog box in the Device Manager console.

When enabled, Device Manager securely stores a users' password that may be used for ongoing authentication with the Device Manager server, such as if the user logs out of the agent.

When disabled, Device Manager will not store users' passwords and will use a certificate for all ongoing authentication with Device Manager. Note that when this setting is enabled, you may still allow users to register and authenticate with a domain password since an enrollment invitation will override this setting when other enrollment modes are configured.

1. Click on the Options dialog box in the Device Manager console.
2. In the left pane, click iOS.
3. Click Store User password settings.
To place a device in supervised mode by using the Apple Configurator

In order to use the Apple Configurator, you will need an Apple computer running OS X 10.7.2 or later.

Some iOS 6 features required that you place your iOS 6 device into supervised mode by using the Apple Configurator.

**Important:** Placing a device into supervised mode will install the selected version of iOS on the device, completely wiping the device of any previously stored user data or apps.

1. Install the Apple Configurator from iTunes.

2. Connect the iOS device to your Apple computer.

3. Start the Apple Configurator. The Configurator shows that you have a device to prepare for supervision.

4. To prepare the device for supervision:
   a. Switch the Supervision control to On. Citrix recommends that you choose this setting if you intend to maintain control of the device on an ongoing basis by reapplying a configuration regularly.
   b. Optionally, provide a name for the device.
   c. In iOS, click Latest, for the latest version of iOS you want to install.

5. When you are ready to prepare the device for supervision, click Prepare.

After you prepare the device, you can now enroll the device into Device Manager and start deploying policies to manage the iOS device.
Managing Android Configurations

You can create a variety of policy types and configurations for your Android devices to help manage user access company data security, including App Tunnel configuration policies so your users can securely access your company intranet, TouchDown Exchange email configurations so users can seamlessly connect to corporate email accounts, app monitoring policies to block unauthorized apps that violate company policy, and a selection of Samsung SAFE device configurations.
You can configure various policies for Android devices in Device Manager so you can more easily manage and ensure consistency across Android device deployments. You can configure the following settings:

- Basic options
- Agent uninstallation
- Password policies (including Encryption for Android 3.0)
- WiFi configurations
- GPRS access point network configurations
- TouchDown email policies
- Security Certificates
- Configurations specific to Samsung SAFE devices
- HTC Exchange ActiveSync configurations

To manage the configuration settings for an Android device, click Configurations in the Policies > Android section in the Device Manager web console and then click New configuration to open the wizards menu. On the wizards menu, you can choose the setting you want to configure.
To configure basic options for Android devices

You can configure some of the agent parameters for Android devices in Device Manager.

1. Click XenMobile Options from the New Configurations menu.

2. In the XenMobile Options dialog box, enter a name to the configuration and optionally enter a comment.

3. Select the Hide traybar icon check box, if you want the tray icon to be visible in the traybar.

4. In Connection time-out, set the connection time-out for the device's connection to the Device Manager server, in seconds. If the device does not connect, cancel the connection attempt.

5. In Keep-alive interval, set the frequency that the device will ping the server, in order to keep the connection alive.

6. Specify the degree to which the device user will be notified of support actions initiated remotely.

7. Click Save.
To uninstall Device Manager on an Android device

1. On the New Configurations menu, click Uninstall XenMobile.
2. Enter a name for the configuration and optionally enter a comment.
3. Select the Uninstall XenMobile from devices check box and then click Save.
To create an Android credential policy

You can create an Android credential to enable integrated authentication with your PKI configuration in Device Manager, such as a PKI entity, a keystore, an a credential provider, or a server certificate. For more information on configuring PKI integration with Device Manager, see About XenMobile PKI.

You can configure an security certificate policy to use for WiFi configurations, TouchDown email configurations, Samsung Exchange ActiveSync configurations, Samsung VPN configurations, and Android SharePoint configurations.

1. In the Device Manager web console, click the Policies tab, click to expand Android and then click Configurations.

2. In the Add a credential dialog box, enter the following information:
   
   a. Credential name. Provide a unique name for the credential.

   b. Description. Optionally, you can type a description for the credential.

   c. Credential Type. Select a credential type according to the PKI configuration you have set up for Device Manager, such as a certificate, a keystore, a server certificate, or a credential provider.

   d. Credential file path, Server certificate, or Credential provider. Select the path or the name of the credential you are adding to the policy. If you are using a Keystore file, then you need to provide the keystore password.

3. Click Add. Now you can access this credential in Android WiFi configurations, TouchDown email configurations, Samsung Exchange ActiveSync configurations, Samsung VPN configurations, and Android SharePoint configurations.
To configure a policy to schedule connections for Android devices

1. In Device Manager, click the Policies tab and then under MDM Policies, click Android.

2. Click Configurations, click New Configuration and then click Scheduling.

3. In the Scheduling dialog box, enter a name for the configuration and optionally, a comment.

4. Under Scheduling configuration parameters, click one of the following options:
   
   - Do not define a connection schedule. The device does not reconnect unless the user clicks Connection in Device Manager.

   - Keep the connection permanently live. If the connection is permanent, Device Manager on the mobile device attempts to reconnect automatically to the server running Device Manager after a network connection loss. In addition, Device Manager monitors the connection by transmitting control packets at regular intervals.

     Note: This configuration consumes more battery charge and generates more network traffic. However, the setting ensures that all commands are executed in real time and completed immediately when they are sent to the device. For more information, see How Scheduling Policies Affect Android Battery Usage.

   - Force a connection every. When you click this option, you also enter a value in minutes. Device Manager on the device forces a connection to the server running Device Manager at every interval you configure.

   - Define a permanent and/or occasional connection schedule within a given time range. Device Manager keeps the connection live during the time range that you configure by selecting one or both of the following check boxes:

     - Keep connection alive during this time period. Device Manager on the device attempts to reconnect to the server running Device Manager after a network connection loss and monitors the connection by transmitting control packets at regular intervals. In the time line, click to select either specific times or time ranges for each day of the week when you want the Device Manager server to connect to the device. Each time segment is in 30-minute increments according to a 24-hour clock. For example, if you want Device Manager to connect between 3 A.M. and 4 A.M. every night of the week, you click the two squares between 3 and 4 for each day of the week, Monday through Sunday.

     - Force one connection during the time range below. The connection automatically shuts down after updates have taken effect. This option forces a scheduled, one-time connection to the server, in particular to check for the availability of new deployments. To avoid a connection peak at the beginning of the selected range, the relevant devices connect randomly during the defined range. Device Manager on the device only reconnects after a network
To configure a policy to schedule connections for Android devices

**Note:** Each of the preceding options includes an option to synchronize the schedule to the local device clock time rather than to Coordinated Universal Time (UTC).

5. Click Create.
How Scheduling Policies Affect Android Battery Usage

When you create a scheduling policy in Device Manager for Android devices, the way you create the scheduling policy can affect battery usage. For example, compared to a device that does not have an XenMobile client agent running on the device, the following may occur:

- If you create a scheduling policy that is set to permanently alive, with app monitoring enabled and basic MDM policy, tests reveal an additional 4 percent battery drain per hour.

- If you create a scheduling policy that is set to permanently always alive, without app monitoring, but with basic MDM policies, tests reveal a 2.5 percent battery drain per hour.

<table>
<thead>
<tr>
<th>Policy and connection conditions</th>
<th>Effect on Android device battery over time</th>
</tr>
</thead>
<tbody>
<tr>
<td>You do not configure an app control policy, but configure the following policies: password policy, TouchDown Email policy, and scheduling policy</td>
<td>8-9 percent over 5 hours equaling approximately 1.6 percent battery drain per hour</td>
</tr>
<tr>
<td>You configure an app control policy, but do not configure a scheduling policy</td>
<td>10-12 percent over 5 hours equaling approximately 2.4 percent battery drain per hour</td>
</tr>
</tbody>
</table>
| You do not configure an app control policy, but you configure the following policies:  
  - Password policy  
  - Software inventory policy  
  - WiFi configuration policy  
  - Credentials policy  
  - SharePoint configuration  
  - TouchDown Email policy  
  - Scheduling policy of permanently alive with a default connection timer set | 10-13 percent over 5 hours equaling approximately 2.6 percent battery drain per hour |
You configure the following policies:

- App control policy (a type of blacklist policy)
- Password policy
- Software inventory policy
- WiFi configuration policy
- Credentials policy
- SharePoint configuration
- TouchDown Email policy
- Scheduling policy of permanently alive with a default connection timer

For more information, see Configuring App Monitoring for Android Apps and To configure a policy to schedule connections for Android devices.
To define password requirements and enable encryption on Android device

You define the requirements for Android device passwords and enable encryption on Android 3.0 devices on the Password Policy window in Device Manager.

1. On the New Configurations menu, click Password Policy.

2. In the Password policy configuration creation dialog box, enter name for the policy and optionally, a description.

3. To establish a password policy, click the Password policy tab.

4. Select Require a code on the device and then complete the configuration parameters.

5. To enable an encryption policy for Android 3.0 devices, click the Encryption tab.

6. Select Enable device storage encryption.

   **Note:** This option is available for Android 3.0 and later. The Android 3.0 encryption operation will prompt the user to accept the action. It also requires the device to be plugged in and the device will not be usable for up to an hour while the encryption operation takes place. This is a function of the Android 3.0 encryption capability.

7. For Samsung SAFE Devices, you have the option of setting a single password for multiples users on a device. Select the Use same password across all users check box to enable this option.

8. Click Create.
To configure WiFi settings for Android devices

You can use the WiFi configuration wizard in Device Manager to deploy Wi-Fi configurations to users. The users will not be aware of details, such as the WEP encryption key. Fill in the required fields according to your configuration and specifically the fields as follows.

1. On the Create a WiFi configuration page, in Configuration name, enter an name for the policy, and optionally enter a description.

2. In Authentication, click one of the following options:
   - Open
   - Shared
   - WPA
   - WPA-PSK
   - WPA2
   - WPA2-PSK
   - 802.1x EAP (WPA Enterprise)

   For the 802.1x EAP configuration, you can specify user identity through the Device Manager macro named ${user.username} to auto-populate in the configuration. Citrix recommends that you leave the password field blank, so the device user can enter the WiFi password from their device.

3. Click Create.
To declare a GPRS access point in an Android device

You can use the APN (Assess Point Name) configuration creation wizard to declare a specific General Packet Radio Service (GPRS) access point for Android devices, like an enterprise private APN.

1. On the General tab, enter a name and optionally a description.

2. Specify the APN resource, account credentials and type of authentication.

3. Optionally, specify proxy settings and then click Create.
Configuring App Monitoring for Android Apps

Android app monitoring in Device Manager provides a secure application-browsing environment on Android devices. You can define blacklisted or whitelisted applications and take action on applications, such as preventing the applications from opening or, in real time, selectively allowing applications to run.

You can define blacklisted or whitelisted applications in an XML file that you package and push to Android devices. Sample XML files are available for reference under <installation directory>/XenMobile Device Manager/samples/appmon/. For example, the default Android app monitoring policy XML file is located at: <installation directory>/XenMobile Device Manager/samples/appmon/appControlPolicyConfiguration.xml. The configuration tags that you can include in the XML file are as follows:

- `<whitelist>` and `<blacklist>`. These tags define applications to be blocked or allowed by package name. Some sample native application package names are as follows:
  - Camera. com.android.camera
  - Browser. com.android.browser
  - Email. com.android.email or com.htc.android.mail

- `<appblockmessage>`. This tag allows customized message to appear as part of the block screen to a user and when a blacklisted or non-whitelisted application opens.

- `<appcontrolpolicylogo>`. This tag allows you to add a custom image to your app block display message when a user is prevented from installing an app. When this element is set to true, the custom logo appears. You must name the custom image appControlPolicyLogo.png and upload the file to Device Manager and then deploy the image file to the device on which you want to display the image.

- `<enforceblacklist>` and `<enforcewhitelist>`. These tags enforce applications through `<blacklist>` or `<whitelist>` tags. In case both these tags are set to true, applications defined in a whitelist XML file take precedence, and the blacklisted applications are ignored.

- `<prevent_uninstall>`. This tag allows you to block a user from uninstalling the Citrix Mobile Connect app from their device. When set to true, a user cannot uninstall the app from their device.

  **Note:** If you set this option to true, you will not be able to uninstall any other apps from the device.

- `<password>`. This tag allows a device to access blacklisted or non-whitelisted applications by using an administrator-defined passcode. There are no restrictions on the length or type of characters in the passcode. You can choose to not include this tag as part of the XML file. As a result, the user cannot enter the passcode in a text box. Instead, block screen appears with a custom company logo file (optional), customized text that you define by using the `<appblockmessage>` tag, and a button that users tap to
<doresetart>. This tag defines if the application control service should be running or not running on the device. If set to false, the service does not run on the device.

**Multiple Configuration Files**

You can define multiple Android app monitoring policy files. For example, you can create a blacklist or a whitelist policy for different groups in your organization, such as a policy for your engineering group, a separate policy for your finance group, sales group, and so on. In order to create multiple app list configuration files, you need to retain the string appControlPolicyConfiguration in the file name. You can, however, modify the other part of the file name to help indicate the purpose of the file. For example:

- appControlPolicyConfigurationOff.xml. An app monitoring policy in which certain apps cannot to run on the device, such as the camera.
- appControlPolicyConfigurationDisable.xml. An app monitoring policy in which certain apps are blacklisted and cannot be installed on the phone.
- appControlPolicyConfigurationEnable.xml. An app monitoring policy in which certain apps are whitelisted and can run on the device.
Example XML Syntax for Blacklisting and Whitelisting Policies

The following code samples illustrate how to use App Monitoring to create application whitelists and blacklists for your Android devices. Blacklisting app use case. Block an native email app on Android devices that are running operating systems Version 3.0 and earlier.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<appcontrol>
  <appcatalog>
    <whitelist>
      <name>org.mozilla.firefox</name>
    </whitelist>
    <blacklist>
      <name>com.android.email</name>
    </blacklist>
  </appcatalog>
  <appblockmessage>This application has been blacklisted and blocked by your Mobile System Administrator. For further inquiries, please contact your IT department.</appblockmessage>
  <enforceblacklist>true</enforceblacklist>
  <enforcewhitelist>false</enforcewhitelist>
  <dorestart>true</dorestart>
  <password>P@ssw0rd</password>
</appcontrol>
```

Whitelisting app use case. Only allow a XenMobile app to run on the Android device, and block all other applications

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<appcontrol>
  <appcatalog>
    <whitelist>
      <name>com.citrix</name>
      <name>com.android.launcher</name>
      <name>com.android.launcher2</name>
      <name>com.htc.launcher</name>
      <name>com.android.email</name>
    </whitelist>
    <blacklist>
      <name>com.android.email</name>
    </blacklist>
  </appcatalog>
  <appblockmessage>This application has been blacklisted and blocked by your Mobile System Administrator. For further inquiries, please contact your IT department.</appblockmessage>
  <enforceblacklist>true</enforceblacklist>
  <enforcewhitelist>false</enforcewhitelist>
  <dorestart>true</dorestart>
  <password>P@ssw0rd</password>
</appcontrol>
```
To add a logo to a customized block screen on an Android device

In Device Manager, you can customize the block screen that appears on an Android device by using the `<appblockmessage>` XML tag defined in an App Monitoring policy. The screen can also include a company logo.

1. Save the logo file as `appControlPolicyLogo.png` on your computer.

2. In the Import a file to the Device Manager database dialog box, import the logo file and then save the file to a destination folder on the device.
   
   **Note:** Make sure you use the following format to name the destination folder has the following format: `%XenMobile folder%\files`.

3. Add the following line to your `appControlPolicyConfiguration.xml` file after the end of `<appblockmessage>` tags: `<appcontrolpolicylogo>true</appcontrolpolicylogo>`

4. Create a deployment package that includes the application monitoring policy XML file, as well as an optional company logo file.
Common Issues with the App Monitoring Policy Implementation

With the App Monitoring feature, you might encounter the following issues:

If you notice that XenMobile is not blacklisting an application you have defined as forbidden, you can try the following tasks to remedy the situation:

- Check the XML file name; it should be appControlPolicyConfiguration.xml.

- Make sure the package containing appControlPolicyConfiguration.xml policy is deployed to the device, and the device is connected to the server.

- Check the package name for the blacklisted application. Use XenMobile Remote Support to verify native application package names under “Task Manager”.

- Validate your appControlPolicyConfiguration.xml file XML syntax with a validator, such as XML Validation.

If you can verify the preceding information, but the issue persists, open a support case and attach the XML file as well as device logs. You can share device logs by using alogcat, a free Android marketplace application.

If you notice that your Company Logo is not included as part of the block screen, verify that logo PNG file is saved as appControlPolicyLogo.png and is saved under %XenMobile folder%\files.

If you need to reset an application passcode, modify the <password> XML tag value to include the new passcode.

If you are not sure if the App Mon service is running, please note that the service is not running by default. You must push the XML policy file (appControlPolicyConfiguration.xml) to the device.

If you need to revoke device access to blacklisted applications, you can modify the <password> XML tag value to include the new passcode. The user needs to obtain and enter the new passcode.
Device Manager leverages NitroDesk's TouchDown technology to enable you to push Exchange email configurations and security policies on Android devices using the ActiveSync protocol. It provides device administrator the ability to install TouchDown software on Android devices, configure device email settings, and apply corporate security policies to Android devices managed by Device Manager. Before you configure policies, do the following prerequisites:

- **Download NitroDesk TouchDown binary from the following locations:**
  - http://nitrodesk.com/tddownloads/nitroid-droid.apk (Smartphones with Android 2.x or 4.x)

You can visit the Android Marketplace and download the TouchDown app, or, you can download the software from your server running Device Manager. To download the TouchDown software to your Android device, make sure that you have either an internal or external SD card. Also, make sure that you enable the following setting on your Android device before attempting the download: Settings → Applications → Unknown Sources

**Obtain a license.** Contact XenMobile and provide them with your ActiveSync URL to get an encrypted license. For example, the ActiveSync URL for XenMobile is webmail.xenmobile.com. NitroDesk will provide the following information for licensing:

- **Server:** webmail.zenprise.com
- **LICENSE KEY:** ABCDEF
- **SERVER KEY** (for ECE Licensing): BC89OjU5MTR2

Once you have obtained your TouchDown license, you will need to add that license key to any Android TouchDown policies you create in Device Manager.

If you want to test TouchDown before purchasing a license, when you install the Touchdown application on the device, a one-month free trial license is included. With this trial license you can still send down the Touchdown policies and settings from the server running XenMobile. The only difference between the paid and free license is that when using the free trial license the mail sync is manual, so the user must initiate the sync to receive email on the device. When using the full paid license, all mail will automatically sync to the device.
Policy Combinations for Touchdown on Android Devices

The following policies combinations are common and useful ways to manage your Android devices with TouchDown.

License TouchDown App and Add Encryption Policy

License Key
RequireDeviceEncryption = true
RequireStorageCardEncryption = true

Require Passcode for TD App

DevicePasswordEnabled = true
MaxInactivityTimeDeviceLock =
MinDevicePasswordLength =
SuppressApplicationPIN = true

Prevent Attachments Download to SD Card

AllowStorageCard = false

Roaming and Custom Signature

RequireManualSyncWhenRoaming = true
SetSignature = Zenprise Protected Tablet

Update Device Type

DeviceTypeString = TouchDown
XenMobile-Certified TouchDown Policies

The following is a list of XenMobile-certified TouchDown policies that you can use with your Android device. Device Manager provides several other policies that are available but not officially certified.

Email Data Encryption Policies

The following two TouchDown policies are required to enable secure email data encryption:

- **RequireStorageCardEncryption = true.** If True, email attachments downloaded to an SD card will be encrypted. Also, the policy disallows moving a TouchDown profile/database to the SD card. Note that attachments prior to this policy will continue to remain in plain text, and all attachments after this policy is activated will be encrypted on the SD card.

- **RequireDeviceEncryption = true.** If True encrypts Contacts, Calendar and Email content; i.e., header as well as body, but not attachments.

TouchDown License Policy

**LicenseKey = <String>.** String value that specifies license key for the TouchDown application.

Individual Security Policies

- **SuppressApplicationPIN = true.** If True, the application will not show a PIN prompt and if you do not want the Exchange ActiveSync PIN to be enforced by TouchDown. This is useful if Device Manager decides to enforce a device level PIN. If set to False, then TouchDown will prompt for pin/passcode only once. To change that behavior, set this policy to false and add the policy named **MaxInactivityTimeDeviceLock,** which prompts the user for a pin/passcode after a period of inactivity.

- **MaxInactivityTimeDeviceLock.** Integer value (in seconds) that defines maximum inactivity time period before device auto locks.

- **DevicePasswordEnabled = true.** If this field is not present, TouchDown will honor the PIN policies that Exchange ActiveSync sends. If this field is present then True = Enable PIN prompting. False = TouchDown will not prompt for PIN (even if EAS policies require it). Please make sure to add the policy named **MinDevicePasswordLength** along with this policy.

- **MinDevicePasswordLength = 1,2,3...14.** Integer value that defines minimum password length for device passwords. Please make sure to add policy:**DevicePasswordEnabled** along with this policy.

- **AlphaNumericDevicePasswordRequired = true|false.** True, if you want TouchDown application to enforce alphanumeric codes for device passwords. Make sure to add policies:**DevicePasswordEnabled** and **MinDevicePasswordLength** along with this policy.
XenMobile-Certified TouchDown Policies

- AllowSimpleDevicePassword = true|false. If True, allows simple device passwords. Make sure to add policies: DevicePasswordEnabled and MinDevicePasswordLength along with this policy.

- AllowStorageCard = true|false. If false, prevents downloading of attachments to the SD card. Also, disallows moving of TouchDown profile/database to the SD card.

- AttachmentsEnabled = true|false. True, ability to send/receive email attachments via TouchDown

- RequireManualSyncWhenRoaming = true|false (to reduce data roaming charges). If True, ability to manually sync email when device is roaming thereby limiting data roaming charges.

- DisableCopyToPhoneBook = true (for data loss prevention purposes). If True, this will cause TouchDown to never copy contacts to the device phone book.
To configure a TouchDown policy to install and configure Exchange email accounts

You can use TouchDown in Device Manager to install and configure Exchange email accounts for users for your Android device users.

**Note:** For each TouchDown policy that you create, be sure to add the TouchDown license key to the policy, or the policy will not work

1. Click the Policies tab and then under Android, click Configurations.

2. Click New Configuration and then click TouchDown Email.

3. In the Add a TouchDown Email configuration dialog box, enter a configuration name for the policy (such as TouchDown Email) and then enter your Exchange email parameters.

   **Important:** While deploying this policy, XenMobile behaves as though the NitroDesk TouchDown application is already being installed from Android Marketplace. You can leave the password field blank, which will prompt the user to enter a password. Or, you can enter the variable `%EWPASSWORD%` which will auto-configure email on the Android device without prompting the device user for the password when they launch their email client.

4. Add the license key to the policy, so you can be sure you are deploying valid software. Click the Policies and Applications Settings tab, click New Configuration and then click Policy.

5. In the New Configuration dialog box, in Name, click LicenseKey and then, in the Value, enter the TouchDown license string.

6. Click Create and then click Add.

After you deploy the policy, the user needs to log in the Android agent and authenticate the users’ credentials in order activate the policy.
To create a deployment package for the TouchDown policy

In order to push the TouchDown email configuration policy to your Android devices, you need to create, configure, and run a deployment package in Device Manager to the devices you want to use the policy.

1. Click the Deployment tab, click New Package and then click New Android Package.

2. In the Create New Package wizard, define and deploy the TouchDown email package. On the Name page, enter a name for the email policy, such as TouchDown Email.

3. On the Groups page, select a group or groups to be the recipient of this package. Or, you can choose to create an anonymous deployment. Any users unable to authenticate themselves to the server can be connected in anonymous mode and still receive packages.

4. On the Resources to be deployed page, in the Policies list, select the TouchDown Email policy you want to deploy and then click the right arrow to add the policy to the package.

5. In Installation Files, select the appropriate Android and TouchDown software to add to the package, depending on if you are deploying to an Android tablet or a phone.

6. In the Deployment schedule page, choose a time to run the deployment, or click Now to run the deployment immediately.

7. On the Deployment rules page,

8. When you have configured the deployment package and are ready to deploy, click Finish.

After you deploy the policy, the user needs to log in on the Android agent to authenticate the user’s credential in order activate the policy.
To initiate a selective wipe of email data by using a TouchDown API

You can initiate a selective wipe of email data (emails and attachments) on an Android device by using a TouchDown application programming interface (API). You can initiate an elective wipe on the Security tab. Status updates are available on a per-device basis on the General tab.

If a device user saved an email attachment to a location outside of the TouchDown default attachments folder, TouchDown won’t be able to detect the action. XenMobile won’t, therefore, delete the data as part of the selective wipe. This is a limitation with the use of the TouchDown API.
Configuring Deployment Rules for an Android Device Size

You can use Device Manager deployment rules to differentiate between a smartphone and a tablet based on the size of the Android device, and then deploy the policies based on size of the target device. The screen size rules enable you to apply specific policies based on whether or not the device is a tablet or a smartphone. Because some deployment resources are tablet-specific, using the screen size property will ensure accurate deployments of tablet- or phone-specific policies.

You can create the rules, for example, if you want to deploy a TouchDown Android policy on all Android tablets except the Amazon Kindle, and you want to ensure that these policies do not get deployed to any smartphones that may happen to be running the same version of Android that the tablets are running. Conversely, you may want to deploy a similar Android package, but for smartphones.

You set the rules in the Edit package wizard, in Deployment rules, on the Simple or Advanced tabs.
List of TouchDown Policies for Android Devices

**AllowHTMLEmail**
Type: Boolean
If True, TouchDown will allow the device to receive email that uses HTML format.

**AllowSimpleDevicePassword**
Type: Boolean
If True, allows simple device passwords.
Please be sure to add the following policies in combination with this policy:

- DevicePasswordEnabled
- MinDevicePasswordLength

**AllowStorageCard**
Type: Boolean
If False, prevents downloading of attachments to a device's SD card. Also, this policy disallows moving a TouchDown profile/database to an SD card.

**AlphaNumericDevicePasswordRequired**
Type: Boolean
If True, TouchDown will enforce alphanumeric codes for device passwords. Please be sure to add the following policies in combination with this policy:

- DevicePasswordEnabled
- MinDevicePasswordLength

**AttachmentsEnabled**
Type: Boolean
If True, allows you to send/receive email attachments via TouchDown.

**DevicePasswordEnabled**
Type: Boolean
If this field is not present, TouchDown will honor the PIN policies that EAS sends. If this field is present, and if you set to True, PIN prompting is enabled and a PIN will be required to access the device. If False, TouchDown will not prompt for a PIN, even if the
**DevicePasswordExpirationDays**

Type: Integer

Value that defines when a device's password is about to expire, measured in days. 0 = no expiration.

**DevicePasswordHistoryCount**

Type: Integer

Value that defines device password where 0 = no history.

**DisableCalendarWidget**

Type: Boolean

If True, the Calendar widget will not show any data.

**DisableChangeSignature**

Type: Boolean

If true, TouchDown disallows user from changing email signature line.

**DisableCleanup**

Type: Boolean

If True, the user will be prevented from being able to wipe configuration settings on the device.

**DisableCopyPaste**

Type: Boolean

If True, users will not be able to copy data from email or paste data into email when composing messages.

**DisableCopyToPhoneBook**

Type: Boolean

If True, this will prevent the user from ever being able to copy contacts to the device phone book.

**DisableDatabaseBackup**

Type: Boolean

If True, the user cannot backup data to an SD card.

**DisableEasyPINRecovery**

Type: Boolean
If True, the user cannot use PIN Reset by entering a Microsoft Exchange account password.

**DisableEmailWidget**

Type: Boolean

If True, email widget will not display any data.

**DisableExportTo3rdPartyWidgets**

Type: Boolean

If true, device cannot export data to external content provider widgets.

**DisableReconfiguration**

Type: Boolean

Reconfiguration of device is disabled except through the MDM client.

**DisableSettingsBackup**

Type: Boolean

If True, user cannot back up device settings to an SD card.

**DisableSpeechNotification**

Type: Boolean

If True, notifications will not be read out loud.

**DisableTaskWidget**

Type: Boolean

If True, task widgets will not display any data.

**DisableUniversalWidget**

Type: Boolean

If True, Universal widget will not display any data.

**HideCalendarInfoOnNotificationBar**

Type: Boolean

If True, notifications will not show calendar data indicating which appointment is scheduled.

**HideEmailInfoOnNotificationBar**

Type: Boolean
If True, notifications will not show Email data.

**HideTaskInfoOnNotificationBar**

Type: Boolean

If True, notifications will not show Task data.

**hideWidgetDataWhenLocked**

Type: Boolean

If True, PIN lock will hide data in widgets.

**License Key**

Type: String

String value that specifies license key that enables running the TouchDown application.

*Note:* Configuring the LicenseKey policy is required in order to use TouchDown Android policies in Device Manager.

**MaxAttachmentSize**

Type: Integer

Integer value that defines maximum size of attachments.

**MaxCalendarAgeFilter**

Type: Integer

Integer value specifying maximum range of past events to sync.

Valid values are as follows:

0 = unlimited, 4 = 2 weeks, 5 = 1 month, 6 = 3 months, 7 = months

Note that this will not impact the currently set values by the user if the current values are more restrictive than this value.

**MaxDevicePasswordFailedAttempts**

Type: Integer

Integer value that defines maximum failed attempts to enter a correct device passcode before locking the user from accessing the device.

**MaxEmailAgeFilter**

Type: Integer

Integer value specifying maximum range of past emails to sync.

**MaxEmailBodyTruncationSize**
Type: Integer

Integer values that determines the maximum sized of an email body before it is truncated.

Valid values:

- 0 - No Body is fetched
- 1-4k
- 2-5k
- 3-7k
- 4-10k
- 5-20k
- 6-50k
- 7-100k
- 8 - unlimited

Raw integral values representing the size in bytes may also be used. For example, if you set to 3000 (above 8), TouchDown will limit to the closest kilobyte unit shown above. Also note, this ONLY limits the upper limit the user chooses, and does not enforce the exact value. For example, if you set the value to 7, the user can then choose to limit to any value less than 100k.

MaxInactivityTimeDeviceLock

Type: Integer

Integer value (in seconds) that defines maximum inactivity time period before device auto locks.

MinDevicePasswordComplexCharacters

Type: Integer

Specifies the number of complex characters required in a device password.

MinDevicePasswordLength

Type: Integer

Defines minimum password length for device passwords.

Please make sure to add the DevicePasswordEnabled policy along with this policy.

PhoneBookCopyFields

Type: Integer

Comma-separated list of fields that can be copied to phone book.
The following fields can be entered in this string, delimited by commas, without any spaces:

- org
- photo
- note
- title
- location
- dept
- wphone
- wphone2
- hphone
- hphone2
- mphone
- ofax
- hfax
- assistantphone
- radiophone
- carphone
- pager
- compphone
- email1
- email2
- email3
- homeaddress
- workaddress
- otheraddress

**RequireDeviceEncryption**

Type: Boolean

If True, encrypts Contacts, Calendar and Email content, such as header as well as body, but not attachments.
List of TouchDown Policies for Android Devices

**RequireStorageCardEncryption**

Type: Boolean

If True, email attachments downloaded to the SD card will be encrypted. Also, True disallows moving of TouchDown profile/database to the SD card.

Please note that attachments prior to this policy will continue to remain in plain text, and after this policy is activated all attachments will be encrypted on the SD card.

**SetPlainTextSignature**

Type: String

String values that specify the signature on the application to be used with plain text email.

**SetSignature**

Type: String

String value that sets the signature on the application.

**SetSuppressions**

Type: String

String value that specifies a list of suppression codes to apply to TouchDown. To prevent TouchDown from displaying certain options to the end user. The list of codes should be comma separated, with at least one comma in the string.

**SupressApplicationPIN**

Type: Boolean

Set to True if you do not want the application to show a PIN prompt, and you do not want the Exchange ActiveSync (EAS) PIN to be enforced by TouchDown. This is useful if the MDM decides to enforce a device level PIN. If False, TouchDown will prompt for pin/passcode only once.

To change that behavior, set this policy to False and add the policy named MaxInactivityTimeDeviceLock, which prompts the user for a pin/passcode after a period of inactivity.
List of TouchDown Application Settings for Android Devices

AlwaysBCCSelf

Type: Boolean

If True, sends a copy (BCC) of all outgoing emails to the configured email address.

AppointmentRemindersAtNonPeakTime

Type: Boolean

If True, reminds user of all appointments even if the appointment occurs during off hours or if the reminder is set to occur during off hours.

CalendarAllDayInWidget

Type: Boolean

If True, this option will show all-day events in the TouchDown Calendar Widget.

CalendarCustomWeekView

Type: Boolean

This option gives two additional options:

- Week starts on
- Week ends on

Using these options the user can change the Week starts on and Week ends on options to select the start and end dates for the week.

Selecting a custom week start and end days will change the way the week view is shown. It will not affect the month view unless your Week start day is before the weekend day (Monday to Saturday).

CalendarDefaultPrivacy

Type: String

Automatically places the same privacy status for each new event unless otherwise specified by the user.

CalendarDefaultReminder

Type: Integer
Automatically places the same reminder length for each new event unless otherwise specified by the user.

**CalendarDefaultStatus**

Type: String

Automatically places the same availability status for each new event unless otherwise specified by the user.

**CalendarEnableResources**

Type: Boolean

If True, gives the ability to specify a resource field when creating new meetings. The user may use the resources field to specify non-attendees such as conference room resources or equipment which are available using an email address.

**CalendarFirstWeekday**

Type: Integer

Specifies the first day of the week to show in the calendar.

**CalendarLastWeekday**

Type: Integer

Specifies the last day of the week to show in the calendar, where 1 - 7 represents Sunday - Saturday. For example, 1 = Monday, 2 = Tuesday, and so on.

**CalendarLightTheme**

Type: Boolean

If True, the day and week Views will be shown with a light theme.

**CalendarOverdueTasksInAgenda**

Type: Boolean

If True, shows overdue tasks in the agenda.

**CalendarShowUpcomingOnly**

Type: Boolean

If True, in the TouchDown Agenda view only current appointments that have not already passed for the current day are shown.

**CalendarSyncHistory**

Defines date range of appointments to synchronize.

Values:
-1 = All
4 = 2 Weeks
5 = 1 Month
6 = 3 Months
7 = 6 Months

**CalendarTasksInAgenda**

Type: Boolean

If True, shows the calendar tasks in the agenda.

**CalendarWorkEnd**

Type: String

Specifies the end of the work day.

**CalendarWorkStart**

Type: String

Specifies the start of the work day.

**CalnedarZoom**

Type: Integer

Indicates zoom size for showing the day and week views in larger size and fonts. A good recommended zoom size for high resolution devices is 150%.

**CleanSDCardonRemoteWipe**

Type: Boolean

Removes data from SD card when a remote wipe command is issued.

- If True, will delete the entire SD card on remote wipe.
- If False, remote wipe will delete only the TouchDown folder.

**CopyToPhoneNameFormat**

Defines how to copy TouchDown Exchange contacts to the phone book as First Last name or as Last First name. Values:

- 0 = First Middle Initial Last
- 1 = Last First Middle Initial
- 2 = File As

**DeferServerUpdates**
List of TouchDown Application Settings for Android Devices

- **Type**: Boolean

  Selected changes are deferred and batched to the server. This is selected by default and improves response time of the application as well as reduce the number of server updates.

- **DeviceTypeString**
  
  - **Type**: String
  
  Default is Android. Once this value is set, it should not be changed.

- **DisableSmartreplies**
  
  - **Type**: Boolean
  
  If True, Smart Replies are turned off. This option should only be selected if the server does not allow SmartReplies and SmartForwards. If forwards and replies are not working, then turn this option ON to determine if it works.

- **DisableTabletMode**
  
  - **Type**: Boolean
  
  If True, disables tablet mode even if it has detected that the user is working on a tablet. This option is specifically for tablet users who prefer the classic TouchDown view.

- **EmailAfterDeleteGoto**
  
  This option lets the user select the behavior when viewing a message and selecting to delete the message. Options include:

  - Email List. Go to the email list.
  
  - Next Email. Open the next email in the list. If none, go back to the email list view
  
  - Previous Email. Go to the previous email in the list. If none, go back to the email list view.

- **EmailAlwaysExpandFolders**
  
  - **Type**: Boolean
  
  If True, then when the user opens Choose Folders or taps the email folder bar to change folders, the folder tree will always appear uncollapsed until the user manually collapses them.

- **EmailBodyStyle**
  
  - **Type**: String
  
  Specifies different fonts, sizes, colors and styles to be used when composing new messages in HTML mode.

- **EmailConfirmDeletes**
  
  - **Type**: Boolean
If True, prompts user with a message each time the user deletes an email to confirm that the email should be deleted.

**EmailDownloadSize**

Defines the download size of the email messages from the server during synchronization. Zimbra users should set this to a value less than or equal to 10 KB.

- 1 = 4KB
- 2 = 5KB
- 3 = 7KB
- 4 = 10KB
- 5 = 20 KB
- 6 = 50 KB
- 7 = 100 KB
- 8 = Full
- 10 = No body

**EmailFetchEmbeddedImages**

Type: Boolean

If True, if using ActiveSync connection mode and HTML emails are enabled, embedded images within emails will automatically be downloaded and displayed. Note that this may cause some refreshing of the email message after each image is fetched and shown.

**EmailHighlightSender**

Type: Boolean

If True, makes the name of email sender of any email item larger and bold (as opposed to the subject).

**EmailHighlightUnread**

Type: Boolean

If True, any read items in the email list will appear grey, without subject or sender in bold, leaving only unread emails fully lit and bold.

**EmailMoveToAny**

Type: Boolean

If True, when the user selects to move email messages to other folders, the user is able to move messages to folders that have not been selected for synchronization. If this is False, then the user can only move emails to folders that have already synchronized.

**EmailMultiSelectors**
If True, each email message in the email list view will show a circle on the right side. The user can place a check mark against each message by tapping the circle. Once selected the user could perform operations like Delete, Mark as Read, Mark as Unread and Move to Folder on all the selected items at once by tapping the Menu button on the device and selecting the option from the menu that opens.

EmailPreviewAttachments

If True, view a sample thumbnail of email attachments after download but before attachments are opened with an attachment viewer.

EmailSearchAsYouType

If True, when the user searches for messages using the Menu/ Search option in the email list view, the messages are filtered according to the search string as typed. If this is False, the user must tap the green arrow next to the search string to perform the search.

EmailShowSummary

If True, displays an email summary.

EmailSyncHistory

Defines a date range of emails to synchronize. Default is 14 days.

EmailTextViewSize

Select the text size to use when viewing email messages. This can be set to 1 of 5 levels: smallest, smaller, normal, larger or largest.

EmailToolBarMode

Select how to display the toolbar. Values:

- 0 = Always show
- 1 = Hide
- 2 = Toggle on shake

EnableHTMLEmail

If True, TouchDown will attempt to download and display emails in HTML format. If False, emails will be retrieved as plain text.
**ExcludeAttachmentsFromGallery**

Type: Boolean

If True, ensures that media files are not scanned by the Android Gallery application when it scans the SD card for media files.

**FilteredTasksOnHomeScreenAndWidgets**

Type: Boolean

If True, displays tasks on the home screen window and on the task widget when they are viewed on the TouchDown Tasks Screen.

**HonorBackgroundDataSetting**

Type: Boolean

If True, honors the user's preference in the Android operating system if user has decided to turn off Background Data in device settings under the Accounts & Sync heading.

**IncludePhoneContactsInPickList**

Type: Boolean

If True, lists contacts from the Android Phone Book as contact options for new email or SMS items.

**ManualSyncWhenRoaming**

Type: Boolean

If True, supresses push and polling when on a roaming network.

**NoDeleteOnServer**

Type: Boolean

If True, deleting emails on the device will not remove them from the server.

**NoMarkReadOnServer**

Type: Boolean

If True, reading emails or marking them as read/unread on the device will not mark them as read/unread on the server.

**NormalizePhoneNumbers**

Type: Boolean

If True, changes contact phone numbers as follows:

- X and x, and extension will be replaced by a ; (semicolon)
List of TouchDown Application Settings for Android Devices

- P and p will be replaced with a ; (semicolon)
- W and w will be replaced with a , (comma)

NotifyAppointments

Type: Boolean

If True, shows a notification for reminders.

NotifyFailedPolling

Type: Boolean

If True, sends a notification when a periodic data refresh fails.

NotifyNewEmail

Type: Boolean

If True, sends a notification when new messages are received.

NotifyPasswordFailure

Type: Boolean

If True, sends a notification when an entered password is incorrect.

NotifySuccessfulPolling

Type: Boolean

If True, sends a notification when a successful data refresh is received.

OffPeakPollInterval

Type: Integer

Defines off-peak polling interval. Any integer >=0, which specifies the polling minutes if polling is enabled during off peak hours.

PollAtOffPeak

Type: Integer

If True, TouchDown will periodically poll for changes even during off peak times.

PollingFrequency

Type: Integer

Defines the frequency to check for changes from the server. An ideal value is 15 minutes. Keep in mind that smaller polling intervals can increase battery drain. (Note: This only applies if Push is not enabled.)

PushEnabled
List of TouchDown Application Settings for Android Devices

**Type:** Boolean

If True, push email is enabled.

**ReminderRepeat**

**Type:** Integer

Allows you to set interval of reminder repeats. Values:

- **0** = No repeat reminders
- **X>0** = repeat after X minutes
- **X<0** = Repeat X minutes before appointment

**ShowEmailsOnStartup**

**Type:** Boolean

If True, TouchDown will always open and display your email list.

**Suppressions**

**Type:** Integer

Comma-separated codes which will specify which fields to suppress.

**UpdateContactChangesToPhone**

**Type:** Boolean

If True, updates contact information on the device when detected on the server.
To update a new version of a custom Android app

Before you can update a custom app to a new version, the app must meet the following requirements:

- The new app package name must be the same as the previous version.
- The app version number must be later than the previous version.

When a new version of a custom Android app (not available on Google Play) is available, you can update the app by adding the new .ipa file to the Files tab. The next time the device connects to the server running Device Manager, the app will be updated to the new version.

**Note:** The app file name can be the same or different and doesn't affect the new version update.

1. On the Files tab, select the iOS app you want to update and then click Edit.

2. To upload an iOS application with a .apk extension, click Choose File and then browse for the app.

3. Click Update.
Configuring Policies for Samsung SAFE Devices

Device Manager supports policy configurations for Samsung SAFE devices so that you can successfully manage your Samsung Android devices. Device Manager Samsung SAFE configurations are compatible with Samsung API Levels Version 2 and 3.

You can access all of the new Samsung configurations on the Policies tab. Under Android, click Configurations and then click New Configuration.

Restrictions Configuration

The Android Restrictions policy allows you to Allow or Disallow the following on Samsung device configurations:

- Common Apps/App Store. YouTube, Browser, Google Play Marketplace, Non-Google Play App Install
- Hardware controls. Factory reset, backup, OTA, clipboard, camera, power off, screenshot capture, SD card, and so on.
- Network settings. Bluetooth, BT tethering, WiFi, WiFi tethering, cellular data, roaming, and so on.
- USB settings. Debugging, mass storage, tethering, and so on.

In the New Samsung restriction configuration dialog box, you configure settings on the following tabs:

- General. On this tab, you enter a name and description for the configuration.
- Applications. This tab allows you to block or allow specific apps and app marketplaces. When you select an option, the app or app store will be allowed on the device. You clear the option if you do not want the device user to be able to access these apps or app stores.
- Hardware controls. This tab allows you to block or allow user control of specific hardware settings on the device. When you select an option, the device user will be able to change the hardware settings. You clear the option if you do not want the device user to be able to change these settings.
- Network. This tab allows you to block or allow user control of specific network settings on the device. When you select an option, the device user will be able to change the network settings. You clear the option if you do not want the device user to be able to change these settings.
- USB. This tab allows you to block or allow user control of specific USB controls on the device. When you select an option, the device user will be able to change the USB controls. You clear the option if you do not want the device user to be able to change these settings.

Exchange ActiveSync Control Configuration
The Samsung Exchange Active Sync control configuration allows you to remotely configure Exchange Email settings, such as server configuration and advanced mail server settings (SSL, synchronize contacts, synchronize calendar, make default email account).

**Note:** In order to push an ActiveSync policy to a SAFE device, you need a SAFE device that is running the XenMobile for Samsung agent.

In the Edit a Samsung Exchange ActiveSync configuration dialog box, you configure settings on the following tabs:

- **General.** On this tab, you can define your Exchange Active Sync configuration you want your Samsung devices to use.

- **Advanced.** On the Advanced tab, you can select or clear the following Exchange Active Sync settings:
  - Use SSL
  - Is default account
  - Synchronize contacts
  - Synchronize calendar

**Firewall Configuration**

The Samsung Firewall configuration allows you to remotely configure firewall settings for your Samsung devices.

In the Edit Samsung firewall configuration dialog box, you configure settings on the following tabs:

- **General.** On this tab, you enter a name and description for the configuration.

- **Allow/Deny Hosts (Blacklisting/Whitelisting).** The Allow/Deny Hosts tab allows you to enter IP addresses or host names you want to either allow (whitelist) or block (blacklisting) from allowing the device to access.

- **Proxy Configuration.** Allows you to remotely configure proxy settings for the device.

- **Re-route Configuration.** Allows you to configure proxy reroute configurations for your devices.

**App Uninstall Configuration**

The App uninstall configuration allows you to block or permit specific apps from being uninstalled from a Samsung device. You can choose from a prepopulated list of apps derived from all software inventories taken from all managed Samsung devices.

In the Edit Samsung App Uninstall Restriction dialog box, you configure settings on the following tabs:

- **General.** On this tab, you enter a name and description for the configuration.

- **Application.** The list of apps that you are blocking or allowing to be uninstalled from an Android device. Click New Application to add a new app to the list.

**Password Policy**
Configuring Policies for Samsung SAFE Devices

The Password Policy configuration allows you to optionally enforce a single device to use the same password for any user accessing the device plus complex flexible password parameters: numeric/alpha numeric restrictions, length requirements, expiration, wipe device after X number of failed attempts, plus encryption for device storage.

In the Password policy configuration update dialog box, you configure settings on the following tabs:

- General. On this tab, you enter a name and description for the configuration.

- Password complexity. This tab give you great flexibility in configuring password complexity parameters for Android devices.

- Encryption. You can choose to enable encryption on the Android devices storage.

  Note: Available for Android 3.0 and later. The Android 3.0 encryption operation will prompt the user to accept the action. It also requires the device to be plugged in and the device will not be usable for up to an hour while the encryption operation takes place. This is a function of the Android 3.0 encryption capability.

- Samsung SAFE. This setting allows you to set a single password for multiples users on a device.

Silent App Un-Installation Configuration

The Silent App Uninstall configuration allows you to initiate software un-installation without requiring user intervention. You can choose from the list of apps derived from all software inventory of all managed Samsung devices in your network. When you deploy the policy, the apps selected will be uninstalled quietly and seamlessly.

Selective Wipe for SAFE Devices

You can use Selective Wipe on Samsung SAFE devices to remove email data, document data, and application data.
Configuring Exchange ActiveSync Policies for HTC devices

Device Manager supports Exchange ActiveSync policy configurations for Android HTC devices. Device Manager supports HTC API version 0.5.0.

To access the HTC Exchange ActiveSync policy, in the Device Manager web console, select Policies tab. Under Android, click Configurations and then click New Configuration > HTC Exchange ActiveSync Configuration.

HTC Exchange ActiveSync Control Configuration

The HTC Exchange Active Sync configuration allows you to remotely configure Exchange Email settings, such as server configuration and advanced mail server settings (SSL, synchronize contacts, synchronize calendar, make default email account).

**Note:** In order to push an ActiveSync policy to an HTC device, you need be running the Citrix Connect agent on an HTC device.

In the Create an HTC Exchange ActiveSync configuration dialog box, you configure settings on the following tabs:

- Configuration Name. Type a name for the Exchange ActiveSync email configuration policy so it is easily identifiable in Device Manager.

- Description. Type an optional description.

- Configuration Display Name. Type a unique name for the email account configuration as it will appear on the device.

- Server Address name. Server address of the Exchange ActiveSync server.

- User ID. Type the email account user name.

- Password. Type the email account user password.

- Domain. Type the domain for the Exchange ActiveSync server.

- Email address. Type the user's email address.

**Note:** In this field, you can use Device Manager system macros `${user.username}` and `${user.mail}`, which will automatically look up specific users and their email accounts based on the format listed.
Citrix Worx Store for Android Apps (MDM-only)

You can provide your users a list of recommended or required apps on their Android devices. You can add external (free or for cost) apps hosted on Google Play or Amazon Application Store as well as internal, in-house apps that your team has developed. All apps appear inside the Citrix Worx Store for Android apps on your users’ Android devices.

Populating the Citrix an Enterprise Application Store for Android requires performing the following tasks:

- Add the custom-made or external apps from the Android Google Play or Amazon app store to Device Manager.
- Create and push a deployment package containing the apps to the device.
- Brand the Citrix Worx store with your company logo or image (optional)

Users view the apps you deploy on their Android devices by opening the Citrix MDM Connect and then tapping the Apps icon.
To add Android apps to Device Manager

To add external apps to Device Manager, you will need the app Web addresses. For example, you can distribute the following apps to your users:

- Nitrodesk TouchDown For Smartphones (if you have a phone).

- Nitrodesk TouchDown HD for Tablets (if you have an Android tablet).

- Project Viewer.
  https://play.google.com/store/apps/details?id=cintelic.project.pro&hl=en

To add internal Android apps, you only need to upload Android app files with the .apk extension.

1. In Device Manager, select the Files tab, click New and then click External APK App.

2. Click Credentials to add your Google Play Market access credentials. Device Manager uses the credentials to access the proper version of the app you will distribute to your employees.

3. Click Save.

4. In the Add an external APK application dialog box, in URL, enter the app Web address and then click Go button to find the app data.

5. Click Add.

6. Repeat Steps 1 through 5 for the other apps. The apps appear on the Files tab in Device Manager.

7. To upload internal apps to Device Manager, click New and then select New App or File.

8. In the Import a file to the Device Manager database dialog box, click Choose File and then browse to select the Android app file (.apk) to import.

9. Select from the following options:
   - Execute APK file. Select this option to execute the installation automatically when the file transfer is done.
   - After installation. Select this option to avoid deleting the installation file from the device when the installation is done.
   - Destination folder. Write down the folder where the file should be uploaded. Only Flash Storage and Device Manager Installation folder are available as path prefix for Android devices.
To add Android apps to Device Manager

- If the file already exists. Copy it if the files are different or do not overwrite the existing one. You can also decide to register a comment if needed.

10. Click Import.
To create a deployment package for Android apps

1. Select the Deployment tab, click New Package and then click New Android Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the package, such as Android App Store, and then click Next.

3. On the Groups of users window, select the group you created earlier and then click Next.

4. On the Resources to be deployed window, under Available Resources, scroll to the Enterprise App Store - , select the check boxes for the two external and two internal apps you want to add, click the right arrow button and then click Next.

5. On the Deployment schedule window, select the If not deployed Start Now option and then click Next.


8. In the packages list, click Deploy.
Managing Windows Phone 8 Configurations

You can create a variety of policy types and configurations for your Windows Phone 8 devices to help manage user and company data, including Windows Phone app distribution through the Enterprise Hub Company store, storage policies (to encrypt stored data and the ability to prevent storage card usage), password policies, Exchange ActiveSync email policies so your users can seamlessly connect to corporate email accounts, as well as your own custom policies.
To create Windows Phone 8 storage policies

1. In the Device Manager web console, click the Policies tab, click to expand Windows Phone 8 and then click Configurations.

2. In the New Configuration menu, click Storage Policy.

3. In the Create a storage policy configuration dialog box, General section, type a name for the policy.

4. Next, select one or both of the policy options:
   
   a. Require Device Encryption. Select if you want all data stored on the device to be encrypted. Selecting this option ensures that no one will be able to access the data without the PIN code, even if the device is cracked and the chip is removed.

   b. Disable Storage Card. Select if you want to prevent a user from storing data on an external storage card.

5. Click Create.
To configure Windows Phone 8 password policies

1. On the Policies tab, under Windows Phone 8, click Configurations.

2. In the New Configuration menu, click new Configuration > Password Policy.

3. In the Create a password policy dialog box, on the General, enter a name for the policy and a brief description.

4. In the Policies section, configure your Windows Phone 8 password policy according to the standards of your IT department. The password policy options are as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Require a password on the device</td>
<td>Enables password protection on the device. If cleared, the device does not require a password on the device (unless the device user sets it manually).</td>
</tr>
<tr>
<td>Allow simple password</td>
<td>Allows the use of a simple password, which is one consisting only of repeated “2222” or sequential “abcd” characters.</td>
</tr>
<tr>
<td>Password complexity</td>
<td>· Alphanumeric: Requires that at least one character of the password is a letter.</td>
</tr>
<tr>
<td></td>
<td>· Alphanumeric or Numeric: Requires that the password contain either at least one letter or one number (but not both).</td>
</tr>
<tr>
<td></td>
<td>· Alphanumeric, Numeric, or none: Password can contain both alphanumeric and numeric characters.</td>
</tr>
<tr>
<td>Minimum password length</td>
<td>Allows you to set the minimum overall length (in characters) required for the password.</td>
</tr>
<tr>
<td>Minimum password complex characters</td>
<td>Allows you to the number of characters that are required to be present in the password. The characters are defined as: lower case alphabetical characters, upper case alphabetical characters, numbers, non-alphanumeric characters. For example, if the value is 2, a password with both upper case and lower case alphabetical characters would be sufficient, as would a password with lower case alphabetical characters and numbers.</td>
</tr>
<tr>
<td>Password expiration (in days)</td>
<td>Allows you to specify the number of days for which the password can remain unchanged. After the set number of days, the user is forced to change the password before the device is unlocked.</td>
</tr>
<tr>
<td>Password history</td>
<td>Allows you to specify the number of previously used passwords to store. When a user creates a new password, the user can't reuse a stored password that was previously used.</td>
</tr>
</tbody>
</table>
To configure Windows Phone 8 password policies

<table>
<thead>
<tr>
<th>Inactivity before device lock (in minutes)</th>
<th>Allows you to specify the length of time that the phone can be inactive before the password is required to reactivate it. You can specify any interval between 30 seconds and 1 hour. The default is 15 minutes. The format of the setting is hh:mm:ss; for example, 15:00 = 15 minutes.</th>
</tr>
</thead>
</table>

5. Click Create. The new policy appears in the Policies list.
To configure Windows Phone 8 Exchange ActiveSync policies

You can use this policy to preconfigure and deploy your corporate Exchange ActiveSync configuration to your Windows Phone 8 device users. Note, however, that the policy does now allow you to set the user password. The device user will need to set that parameter from the device once the policy is pushed.

1. On the Policies tab, under Windows Phone 8, click Configurations.

2. In the New Configuration menu, click New Configuration > Exchange ActiveSync.

3. In the Create a new Exchange ActiveSync configuration dialog box, on the General tab, enter the following information:

   a. Configuration Name. Type a name for the policy.

   b. Description. Type an optional description.

4. In the Email parameters section, enter the following information:

   a. Account name. Name of the Exchange ActiveSync account.

   b. User name. Type the account user name.

   c. Domain. Type the domain for the Exchange ActiveSync server.

   d. Email address. Type the user's email address. In this field, you can use Device Manager system macros ${user.username} and ${user.mail}, which will automatically look up specific users and their email accounts based on the format listed.

   e. Server name. Type the name of the Exchange ActiveSync server.

5. Click the Advanced tab and enter the following information:

   a. Synchronization Frequency. Select the frequency with which you want the email account on the device to sync with the Exchange server. This setting specifies how often any new data from the server will be sent to the device.

   b. Synchronization Items. Select which items you want to be synced, such as email, contacts, calendar, and so on.

   c. Loggin. Specify the level of detail for logging of Exchange activity (or no loggin).

6. Click Create.
To configure Windows Phone 8 custom policies

1. On the Policies tab, under Windows Phone 8, click Configurations.
2. In the New Configuration menu, click New Configuration > Custom Policy.
3. Enter your own custom XML configuration for Windows Phone 8, and then click Validate.
Creating Windows 8 Tablet Registry Configurations

You can create Windows 8 Tablet registry configurations to allow for a single point for device registry management. A set of registry keys can only exist in a configuration. You can create different settings and then deploy them selectively to some or all of the mobile devices under management. Once you create the registry settings, you can then deploy them to your Windows 8 devices using a Device Manager deployment package.
To create a Windows 8 tablet registry configuration

1. In the Device Manager web console, on the Policies tab, under Windows 8 Tablet, click Registry Configurations.

2. Click New > Configuration.

3. In the New registry configuration dialog box, type a name and then click Create. This will create a default, blank registry value set, for which you can create custom registry entries to suit your requirements.

4. To create a new registry key, select a registry folder and click New > Key. Type a name for the new key and then click Create.

5. To enter a registry key value, click New Value.

6. In the Create a new registry value dialog box, enter the following information:
   a. Name. Type a name for the registry key.
   b. Type. Choose the registry type, such as String, DWORD, Expandable String, or Executable.
   c. Value.

7. Click Create.
Managing Windows Mobile Configurations

You can create several types of device management policies and configurations for your Windows Mobile devices such as App Tunnels for secure connections to your corporate network at the app level, registry setting configurations, server settings policies and custom XML configuration policies.
About the Windows Mobile Server Collection

The Servers collection in Device Manager is used to both administer existing and to add new servers to your Device Manager deployment. A default server configuration is created during the Device Manager installation.

In the Server Groups collection for Windows Mobile, you can add new access points to the server and create backup servers. You can also configure server groups. You can use server groups to do the following:

- Create one or more backup servers (valid only if strong authentication is not enabled for the product).
- Define several access points for connection to the server running Device Manager.
- Provide logical grouping for multiple deployment locations.
To add a new server in the Windows Mobile server collection

To create other servers running Device Manager, you first need to create a new server group. To do so, click New group. After you create at least one group, you can then create a new server.

If a device cannot connect to the selected server, the device will attempt to connect through other servers in the same group, one after the other, following the defined order, through to the default server. To change the order in which servers are listed, you right-click a server and then click Options (or click Down).

1. Click New server.
2. Enter a display name for the server.
3. Enter the IP address or fully qualified host name (FQDN) of the server.
4. Enter the host port.
5. Choose optional settings of SSL, use a proxy server, or use as the default server.
To edit or delete a server

1. Click the server whose settings you want to modify and then click Update.
2. Modify the settings and then click Update.

To delete a server

1. Click the server whose settings you want to modify and then click Delete.
2. When prompted, click to confirm the deletion.
To configure device IP address ranges

By design, devices connect to the default server running Device Manager at the provided host name or IP address. For situations in which you have a LAN, WiFi, or USB connection, you can specify IP address ranges. If the Windows Mobile device has an IP address within this range, the device will connect to the server running Device Manager. Specifying IP address ranges is useful when uploading new software because you can lower data charges.

1. Click an existing server create a new server.

2. Click Update.

3. Click the Device IP ranges tab.

4. Click New IP address range.

5. Enter the starting and ending IP addresses and then click Create. When you select a server, you can also update the server settings, such as the IP address, default server, and IP address ranges.

You must also deploy a server group to a user group on the Deployment tab.
About the Registry Collection for Windows Mobile

The Registry collection is used to configure the Windows Mobile or Windows CE registry base of the mobile devices, thus allowing for a single point for device registry management. Device Manager includes a series of step-by-step wizards, allowing for rapid setup and deployment of registry configurations.

In addition to configuring both new and existing registry values, either for the operating system or installed third-party applications, you can manage the Device Manager client configuration options. This allows for multiple client backup settings, and control over network connectivity, and is included with a dedicated wizard.

A set of registry keys can only exist in a configuration. You can create different settings and then deploy them selectively to some or all of the mobile devices in the fleet. You can create a new registry configuration either manually or via the wizard. The configuration wizard includes pre-configured options for the following applications:

- Device Manager options
- Uninstall Device Manager client from a device
- Scheduling
- Connect on SMS reception / Connect on call
- MS Exchange configuration for MS Outlook
- Security rules

**Note:** This feature is available only with the Device Manager Secure Device option.

- Configure devices when roaming
Configuring Registry Keys by Using the Device Manager Options Wizard

Device Manager uses registry keys to store its own data in each mobile device. You can configure these options by using the Device Manager Options wizard.

Device Manager configuration backup. Device Manager client settings can be stored on the removable memory card in the mobile device. If a mobile device has to be hard reset, it will automatically retrieve the settings required to reconstruct the configuration, such as Device Manager agent, registry keys, Device Manager-related security certificates, and network configuration. For devices with more than one memory card, you can also configure a backup to a specific card.

Connect to these networks. Device Manager is authorized, if necessary, to activate connections as defined in Network Management on the mobile device:

- User-defined Office
- User-defined Internet
- Built-in Office (My Work Network connection)
- Built-in Internet (My ISP connection)

The operating mode depends on the status of the mobile device's current network connection at any given time:

<table>
<thead>
<tr>
<th>Authorized and Active connection</th>
<th>If the server running Device Manager is accessible through this network, the Device Manager agent connects to the server running Device Manager.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unauthorized but Active connection</td>
<td>The mode is the same as for Authorized and Active connection.</td>
</tr>
<tr>
<td>Authorized and Inactive connection</td>
<td>The Device Manager agent will activate the connection and then connect to the server running Device Manager.</td>
</tr>
<tr>
<td>Unauthorized but Active connection</td>
<td>The Device Manager agent will not attempt to activate the network connection.</td>
</tr>
</tbody>
</table>

- Device Manager icon. This option hides or displays the Device Manager icon in the mobile device's traybar.
- Connection time-out. This option sets the connection time-out for the device's connection to the server running Device Manager, in seconds. If the device does not connect, cancel the connection attempt.
- Keep-alive interval. This option sets the frequency that the device will ping the server, in order to keep the connection alive.
Ask the user before allowing remote control. When a connection is established with the helpdesk, the remote device prompts the user to allow the helpdesk to take remote control over the mobile device through a confirmation dialog box.

Ask the user before allowing file transfer by the remote control tool. File transfers from a device to the server can be configured for anonymous mode on the Device Manager web console, with user confirmation of the request or with only presentation of a message to inform the user.
To uninstall Device Manager from a Windows Mobile device

It is preferable to create a special group, such as UninstallGroup, on the Users tab to uninstall Device Manager and then create a package of Device Manager option registry keys containing the uninstall option. You can then deploy this package to UninstallGroup on the Deployment tab. Thereafter, you can add a user to the UninstallGroup in order to uninstall Device Manager from the remote device.

1. Select Uninstall XenMobile from devices.

2. On the Deployment tab, deploy this configuration to selected users.
Configuring a Connection to Device Manager on SMS Reception or Call

This feature allows for Windows Mobile devices to be forced to connect back to the server running Device Manager when either a call or SMS from a preconfigured number is received by the device. To enhance security, a keyword must be included within the SMS message. This is particularly useful if a device is lost or stolen, and needs to be remotely disabled or wiped.

To use this feature, in the Connect On SMS reception / Connect on Call dialog box, select to either connect to server when receiving a SMS message or phone from a specific number.
Configuring Exchange Server for Windows Mobile Devices

Using the MS Exchange configuration for MS Outlook wizard in Device Manager, you can configure mobile email settings easily and automatically across your entire fleet of devices. These settings will generate the appropriate registry keys to synchronize with an Exchange server:

- Exchange server name
- Settings for receive emails and attachments
- Calendar settings
- Other settings

On the General tab, elect the appropriate device operating system type because different configuration options are available depending on the operating system release. For instance, tasks synchronization is available for Windows Mobile 6 devices but not for Windows Mobile 2003 or Windows Mobile 5.

If you create an Exchange tunnel, the value you enter in Server address has to be the same as the value you enter in Specify a local alias on the Tunnel tab, if you specified an alias. The server running Device Manager manages and optimizes the data stream and communication between the Exchange server and the mobile device.
Configuring Windows Mobile Devices for Roaming Situations

The roaming settings in Device Manager for Windows Mobile devices will generate the appropriate registry keys for a better control of the wireless communications costs while traveling abroad and using other mobile operator networks than your default mobile operator (for example, the name of the mobile operator stored on the SIM card of the mobile device). In roaming situations, when the device has a cellular connectivity setup, the device will connect to the server running Device Manager according the following settings:

Note: You can select more than one setting.

- Use on demand connection only. The device will only connect to the Device Manager Server if the end-user manually triggers the connection using the Device Manager Agent screen on his device, or if a mobile application requests a forced connection (such as a push mail request if the Exchange server has been set accordingly). Note that this option temporarily disables the default device connection schedule policy as defined in the Scheduling wizard within the Registry tab.

- Block all cellular connections except the ones managed by Device Manager. except for the data traffic officially declared in a Device Manager application tunnel or other Device Manager device management tasks, no other data will be sent or received by the device. For example this option will disable all connections to the Internet via device web browser (Pocket IE).

- Block all cellular connections managed by Device Manager. All "application" data transiting through a Device Manager tunnel will be blocked (including the Device Manager Remote Support application). However the data traffic related to pure “device management” (such as the deployment of a new Device Manager package) will not be blocked.

- Block all cellular connections to the Device Manager Server. In this case, until the device is either reconnected via USB, Wi-Fi or via its default mobile operator cellular network, there will be no traffic transiting between the device and the Device Manager Server.

In the Deployment tab, you can also configure rule to avoid deploying a specific package (say “XYZ”) when roaming. In this case, if the Block all cellular connections managed by Device Manager option has been selected, all packages except “XYZ” will still be deployed even in roaming situations.

In the Tunnel tab, a given application tunnel can be forced to block all data traffic when roaming. For example, if the Block all cellular connections except the ones managed by Device Manager option has been selected, the “CRM_App” data traffic will still be blocked although it is managed as a Device Manager tunnel.
To configure a new registry manually

1. Click New configuration.

2. Enter a name. This will create a default, blank registry value set, for which you can create custom registry entries to suit your requirements.
To delete a registry key configuration

1. Select the registry configuration to be deleted.
2. Click Delete.
3. When prompted, click to confirm the deletion.
To use the schedule wizard to configure connections for Windows Mobile devices

1. On the Policies tab, click Registry Configurations, and then on the Wizard menu, click Scheduling.

2. In Scheduling configuration parameters, select the following options:

   - Do not define connection policy. The device will not reconnect unless the user clicks Connection in Device Manager.

   - Keep connection permanently live. If the connection is permanent, Device Manager on the mobile device will attempt to reconnect automatically to the server running Device Manager after a network connection loss and will monitor the connection by transmitting control packets at regular intervals (This configuration is not recommended because it consumes more battery charge and generates more network traffic.)

   - Define a permanent and/or occasional connection schedule within a given time range. Keep the connection live during the following time range:

     - Define a period in which the device will stay connected to the server. Device Manager on the device will attempt to reconnect to the server running Device Manager after a network connection loss and will monitor the connection by transmitting control packets at regular intervals.

     - Force one connection during the time range below. The connection will automatically shut down once updates have taken effect. This option forces a scheduled, one-time connection to the server, in particular to check for availability of new deployments. To avoid a connection peak at the beginning of the selected range, the relevant devices will connect randomly during the defined range. Device Manager on the device will only reconnect after a network connection loss if an operation was in progress. The server running Device Manager will likewise terminate the connection after an inactive period.

     **Note:** Both of the preceding options include an option to see the schedule to the local device clock or to UTC time.
Managing SharePoint Configurations

Citrix data loss prevention (DLP) solution enables access from your mobile workforce to your SharePoint content. You can apply access control rules to content to prevent unauthorized usage depending on document classification. In XenMobile, you can use the DLP-SharePoint/Encrypted Email Attachment Viewing feature to manage SharePoint configurations. If the SharePoint configuration item is unavailable, your license does not include the SharePoint feature. To check your license features, view the About tab in the web console.

If you are planning to use the Device Manager SharePoint access management feature, make sure your deployment meets the following Windows requirements:

- SharePoint 2010 or Office 365.
- Windows 2008 R2 - SharePoint 2010 SP1 is required or KB976127.
- Windows 2008 - Rest API calls will fail unless KB976217 is also installed.

**Note:** Make sure that your SharePoint folders on the SharePoint server do not use special characters such as commas (,), semicolons (;), or periods (.), or those folders will not appear on your users' devices.
To configure a SharePoint resource configuration site in Device Manager

When you configure a SharePoint resource configuration site, you define the SharePoint server settings and specific directories (folders) that you want to expose to the device user.

**Note:** Make sure that your SharePoint folders on the SharePoint server do not use special characters such as a comma(,), semicolon(;;), or period(.), or they will not appear on your users’ devices.

1. In Device Manager, click the Policies tab.


3. Click New Configuration.

4. In the Create a resource configuration dialog box, on the Site/Folder Config tab, enter a name for the SharePoint site and then configure the following:

   - Name. The name of the resource definition.
   - Description. A free text description describing the resource.
   - Site. Enter the SharePoint site Web address.
   - Doc Library/Folder. Enter the list of path relative to the base site that you want to publish.
   - Include Sub-folders. Enables the access to sub folders of the above defined path.
   - Document Control. Check all the document controls that are applied to the doc libraries.

5. In the Options dialog box, configure extra options you would like to apply to the SharePoint document folder on your users’ iOS devices, such as if the documents should be wiped from the device if the device is jailbroken, encryption and annotation of documents, and so on and then click Create.
To configure a SharePoint document control policy

You can enable your users to securely access corporate SharePoint content. You apply access control rules, on the content in order to prevent unauthorized usage or actions, depending on your company policy and document sensitivity. You can view this content on the Connect agent on a device in the Documents -> Shared Docs folder. You create a SharePoint control policy to define explicitly what a device user can and cannot do with documents in their secure document container on their devices, such as whether or not documents can be printed, if a user can copy and paste to and from documents, if document check in/check out is allowed, and so on.

1. Under XenMobile Policies, under SharePoint, click Control Policy.

2. Click New Control Policy.

3. In the New Control Policy dialog box, on the General tab, enter a name for the policy.

4. On the Document Control tab, you can set the control policies for all the documents in the folders specified in your SharePoint resource configuration.

5. You can define the following controls that will be applied to the documents by selecting them. Any options left unselected will not be allowed by document users.

   • Allowed features
     • Document synchronization. Allow the document to be synchronized to the device. If not checked, the document is only accessible online.

     **Note:** If you want to be able to annotate PDF files on your device, you need to make sure this option is selected, since the PDF annotation tools only work with locally synced documents.

     • Copy/Paste of content. Allow copy/paste of document content.

     • Email link to document. Enable users to send a link to this document via email.

     • Email document. Allow users to send this document via email.

     • Print document. Allow users to print this document.

     • Document check in. Allow users to check in this document from SharePoint.

     • Document check out. Allow users to check out this document from SharePoint.

     • Open document in another application. Allow users to open this document in a third-party application on the device. If not checked, only the internal viewer can be used.

     • Time expiration
To configure a SharePoint document control policy

- Expires on a date. Specify a date after which the document is not be accessible. If on the device, it will be deleted.

- Expires after x Days. Specify the duration of validity of the document. After the specified period, the document is not accessible.

- Authentication expiration
  - Specify an authentication timeout. If the user does not authenticate regularly to SharePoint, the documents become inaccessible.

6. Click Create.
Configuring SharePoint on Android Devices

To configure a SharePoint data loss prevention (DLP) connection for Android, you need to do the following:

- Create an application tunnel that the SharePoint server will use to communicate securely with the device; use a client port.
- Create a SharePoint resource configuration that configures the SharePoint site server address; be sure to use the client port configured in the application tunnel.
- Create a SharePoint policy to configure the security and access parameters for the SharePoint site.
- Deploy the policy to the device.

To create the application tunnel

1. On the Policies tab, click Android, click Tunnels and then click New Tunnel.
2. In the Create a Tunnel dialog box, enter the follow app tunnel parameters:
   - Name. Give the app tunnel a name that indicates it is going to be for a SharePoint connection.
   - Application Device Parameters Client Port. The port number that will be used by the XenMobile client application on the device.
     
     **Note:** You will need to use this same port when you configure the SharePoint resource.
   - Application Server Parameters
     - IP address or server name. Address of the SharePoint server.
     - Server Port. SharePoint server port.
3. Click Create.

To create a new SharePoint control policy

A SharePoint control policy defines a set of actions that the user will be able to execute on documents. Document Control allows you to define all the features that will be applied to the documents:

1. On the lower-left, click Control Policies and then click New Control Policy.
2. In the Create new control policy dialog box, on the General tab, enter a name such as "Android Employee SharePoint Documents."

3. On the Document Control tab, configure the following settings:
   - Allowed features
     - Document synchronization. Allow the document to be synchronized to the device. If cleared, the document is only accessible online.
       **Note:** If you want to be able to annotate PDF files on your device, you need to make sure this option is selected, since the PDF annotation tools only work with locally synced documents.
     - Copy/Paste of content. Allow copy/paste of document content.
     - Email link to document. Enable the user to send a link to this document via email.
     - Email document. Allow the user to send this document via email.
     - Print document. Allow the user to print this document.
     - Document check in. Allow the user to check in this document from SharePoint.
     - Document check out. Allow the user to check out this document from SharePoint.
     - Open document in another application. Allow the user to open this document in a third-party application on the device. If not selected, only the internal viewer can be used.
   - Time expiration
     - Expires on a date. Specify a date after which the document is not be accessible. If on the device, it will be deleted.
     - Expires after x Days. Specify the duration of validity of the document. After the specified period, the document is not accessible.
   - Authentication expiration
     - Specify an authentication timeout. If the user does not authenticate regularly to SharePoint, the documents become inaccessible.

4. Click the Tag Mapping tab to enable deeper integration to your SharePoint libraries by applying document controls based on already defined tags on your SharePoint documents.

5. Click Create.

**To create a SharePoint resource configuration**

A SharePoint resource configuration defines a SharePoint document library access and the control policies that are tied to its documents.
1. Click the Policies tab and then from the left side, click SharePoint Resource Configuration.

2. Click New Configuration.

3. In the Create a Resource Configuration dialog box, enter a name and description of the SharePoint server connection.

4. Enter the SharePoint server name plus the port number you configured in your application tunnel for the Client Port (2500, for example).

5. Under Document Control, select the Control Policy you created. Other options:
   - Include Sub-folders. Enables the access to sub-folders of the above defined path.
   - Document Control. Check all the document controls that are applied to the doc libraries.

6. Click Create. The new SharePoint server appears as a resource in Device Manager. Now the SharePoint server and resource you configured is ready to be accessed by users.

**To deploy the SharePoint resource to your device**

To enable your client users to access the content in this SharePoint site, you need to create a deployment package that contains the SharePoint resource and then push that deployment to your device. Once on your device, you can launch the client application and access the documents folder to view the documents contained on the SharePoint server. Now, you will create a deployment package and push the new Android SharePoint resource to your Android device.

1. Click the Deployment tab, click New Package and then click New Android Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the SharePoint package (such as Android SharePoint Package) and then click Next.

3. In the Groups of users window, select a group to which you want to deploy the SharePoint package to and then click Next.

4. In the Resources to be deployed window, under Available Resources, scroll to the SharePoint folder, select the SharePoint Configuration you created in the last step and then click the right arrow to add the resource to the deployment package.

5. Scroll in the Available Resources list and then on the Tunnels folder, click the application tunnel you created for your Android SharePoint configuration.

6. Click the right arrow to add the resource to the deployment package and then click Next.

7. In the Deployment schedule window, select the If not deployed Start Now option and then click Next.


10. From the Packages list, click Deploy.

When the deployment has finished, select the deployment package, and then click Details to see information about the success of the package deployment. When the package shows as deployed, you can check the success of your deployment. Select the deployment package, open the Connect client on the Android device and then tap the Documents folder. From here, users can open documents from the SharePoint site.
Symbian devices configuration in Device Manager is done by sending OMA Device Management commands to the devices. The list of supported commands can be found from the Nokia Developer web site. A search for the “OMA Device Management” keywords in the document section of web site will return a number a documents describing Device Description File (DDF) for features that can be controlled using OMA-DM.

OMA DM will allow control of Symbian devices by:

- Defining Wifi or GPRS Access Points.
- Defining Mail for Exchange parameters.
- Encrypting device and/or SD cards.
- Customizing devices.
- Configuring VoIP parameters.

That list is not an exhaustive list of what can be configured on Symbian devices, and features may depends on the device model. For instance, device encryption was supported on S60 3.2 devices only on the E-Series devices.

Device Manager support <Alert>, <Add>, <Replace> and <Exec> OMA-DM commands.

The following example will display the message "Management in progress..." on the user device during 30 seconds.

```
<Alert>
  <CmdID>_cmdid_</CmdID>
  <Data>1100</Data>
  <Item>
    <Data>MINDT=30</Data>
  </Item>
  <Item>
    <Data>Management in progress...</Data>
  </Item>
</Alert>
```

The structure of an OMA-DM command must always contain a command ID. In this case, the command ID is interpreted on the fly. It is then replaced by the placeholder "_cmdid_". The following example will configure the Mail for Exchange client.

```
<Replace> <CmdID>_cmdid_</CmdID> <Item>
  <Target><LocURI>./MailForExchange/Server</LocURI></Target>
  <Data>webmail.mycompany.com</Data>
</Item>
</Replace>
```

```
<Replace> <CmdID>_cmdid_</CmdID> <Item>
  <Target><LocURI>./MailForExchange/UseDefaultPort</LocURI></Target>
</Item>
```
Creating Symbian Configuration Profiles

Note that this is just an example; a lot more options are available to configure in the Mail for Exchange client. As you can see, several commands can be chained in the same command block. User attributes can be used using $user.attributename macro. Those macro will be replaced on the fly by the actual user data.

Special config can be created to enforce configuration on Symbian devices. Those configurations will have the following format.

<Exec> <CmdID>_cmdid_</CmdID> <Item> <Target> <LocURI>am_policy</LocURI> </Target> </Item> </Exec>

The <LocURI> parameter can be set with the following values:

- am_policy / am_policy_del: This will set or remove the application management policy enforcement to control application installation and removal.

- ap_policy / ap_policy_del: This will set or remove the Access Points policy management enforcement.

- custo_policy / custo_policy_del: This will set or remove the device customization policy management enforcement.

- ds_policy / ds_policy_del: This will set or remove the Data Synchronization policy management enforcement.

- email_policy / email_policy_del: This will set or remove the Email policy management enforcement.

- im_policy / im_policy_del: This will set or remove the Instant Messaging policy management enforcement.

- wlan_policy / wlan_policy_del: This will set or remove the WLAN Access Points policy management enforcement.
To create a new Symbian configuration profile, on the Policies tab, click New configuration in the Configurations section of Symbian.
To use the schedule wizard to configure connections for Symbian devices

1. On the Policies tab, click Configurations, and then on the New Configuration menu, click Scheduling.

2. In the Scheduling dialog box in Device Manager, enter a name for the configuration and optionally, a description.

3. In Scheduling configuration parameters, select the following options:

   · Do not define connection policy. The device will not reconnect unless the user clicks Connection in Device Manager.

   · Keep connection permanently live. If the connection is permanent, Device Manager on the mobile device will attempt to reconnect automatically to the server running Device Manager after a network connection loss and will monitor the connection by transmitting control packets at regular intervals (This configuration is not recommended because it consumes more battery charge and generates more network traffic.)

   · Define a permanent and/or occasional connection schedule within a given time range. Keep the connection live during the following time range:

       · Define a period in which the device will stay connected to the server. Device Manager on the device will attempt to reconnect to the server running Device Manager after a network connection loss and will monitor the connection by transmitting control packets at regular intervals.

       · Force one connection during the time range below. The connection will automatically shut down once updates have taken effect. This option forces a scheduled, one-time connection to the server, in particular to check for availability of new deployments. To avoid a connection peak at the beginning of the selected range, the relevant devices will connect randomly during the defined range. Device Manager on the device will only reconnect after a network connection loss if an operation was in progress. The server running Device Manager will likewise terminate the connection after an inactive period.

   Note: Both of the preceding options include an option to see the schedule to the local device clock or to UTC time.
Scheduling Connections to Device Manager

Scheduling provides essential control over devices that are subject to compliance rules. The schedule feature directs the device to automatically connect to the server running Device Manager at predetermined intervals. During these connections, a policy audit automatically occurs and missing or modified policies are automatically reapplied. Additionally, scheduling ensures that Device Manager has the most up-to-date device information available.

**Note:** Flexible scheduling is available on Android and Windows Mobile devices only. iOS devices use a predetermined schedule defined by the iOS operating system.

The Scheduling Wizard is located in the Device Manager web console under the Policies tab. Scheduling provides Registry keys for managing scheduled connections between a device and the server. This is useful for devices that require the ability to connect back for data synchronization between a Line-of-Business, ERP, or CRM-type system.

To define an hourly range in the scheduling table, you can either click in a square or you can drag and drop with your mouse to define a range. (First left-click on a square and then, keeping the button pressed, move the cursor over another square and release the button.)
Configuring Macros

Device Manager provides powerful macros that provide a method to populate user or device property data within the text field of any profile or policy or notification/enrollment templates (for some automated actions), to name a few usages. With macros, an administrator can configure a single policy and deploy it to a large user base and have user-specific values appear for each targeted user. For example, an administrator could pre-populate the mailbox value for a user in an Exchange profile across thousands of users.

This section provides an overview on the use of macros in Device Manager.

This feature is currently only available in the context of configurations and templates for iOS and Android devices.
A macro can take the following form:

- ${type.PROPERTYNAME}
- ${type.PROPERTYNAME ['DEFAULT VALUE'] [ | FUNCTION [(ARGUMENT1, ARGUMENT2)] ]}

As a general rule, all syntax following the dollar ($) sign must be enclosed in curly brackets ({}).

- Qualified property names reference either a user property, a device property or a custom property.
- Qualified property names consist of a prefix, followed by the actual property name.
- User properties take the form ${user.[PROPERTYNAME] (prefix="user.")}.
- Device properties take the form ${device.[PROPERTYNAME] (prefix="device.")}.

For example, ${user.username} populates the username value in the text field of a policy. This is useful for configuring Exchange ActiveSync profiles and other profiles used by multiple users.

For custom macros (properties that you define), the prefix is ${custom}.. You can omit the prefix.

**Note:** Property names are case-sensitive.
Defining User Macros

The following user macros are always available:

- loginname (username + domainname)
- username (loginname minus the domain, if any)
- domainname (domain name, or the default domain)

The following administrator-defined properties may be available:

- c
- cn
- company
- companyname
- department
- description
- displayname
- distinguishedname
- facsimiletelephonenumber
- givenname
- homecity
- homecountry
- homefax
- homephone
- homestate
- homestreetaddress
- homezip
- ipphone
- l
- mail
Defining User Macros

- middleinitial
- mobile
- officestreetaddress
- pager
- physicaldeliveryofficename
- postalcode
- postofficebox
- telephonenumber
- samaccountname
- sn
- st
- streetaddress
- title
- userprincipalname
- domainname (overrides property described above)

Additionally, if the user is authenticated by using an authentication server, such as LDAP, all the properties associated with the user in that store are available.
Managing Devices

You can manage devices by using the following:

- Tagging devices to identify ownership of the device. You can tag devices with a script or by using the Device Manager web console.

- Adding devices to Device Manager either manually or by using the Device Provisioning tool.

- Locking and unlocking devices by using the Device Manager web console.

- Revoking device certificates to prevent devices from accessing Device Manager.

- Wiping information from devices that includes removing some or all data on the device.
Tagging User Devices Automatically

You can tag your users’ devices as either corporate-owned or employee-owned to keep track of your company's Bring Your Own Device (BYOD) program, either automatically with a script, or manually by using the Device Manager web console. To enable employee and corporate device tagging, you will need to download a Microsoft PHP, add device IDs to a CSV file, and execute the given XenMobile scripts that will automate the device tagging process. After setting up the device tagging, you will schedule the script as a repeating Windows Task to run every minute.

Note: For on-premise deployments, the tagDevices.php script is located at C:\Program Files (x86)\Citrix\XenMobile Device Manager\samples\WebServices.
To set up device tagging

1. In a browser, go to the Windows PHP download site at http://windows.php.net/download/.

2. Download the installer package named php 5.3 (VC9 x86 Thread Safe (2012-Feb-02 21:56:19).

3. Install the package on your local system at c:\php5.

4. Copy the two files named tagDevice.php and devices.csv to c:\temp (this PHP script is host, location and platform agnostic).

5. Open the tagDevice.php file in a text editor and replace the default information (highlighted) with the following parameters:

   • For an on-site Device Manager implementation:

     ```
     $soap_url = "<servername>/zdm/services/EveryWanDevice?wsdl"
     $client = new SoapClient(null, array(  
        'location' => $soap_url,
        'url' => "<servername>",
        'login' => "demo",
        'password' => "XXXXX"));
     ```

     For example:

     ```
     $soap_url = "mdm.zenprise.com/zdm/services/EveryWanDevice?wsdl"
     $client = new SoapClient(null, array(  
        'location' => $soap_url,
        'url' => "mdm.zenprise.com",
        'login' => "demo",
        'password' => "XXXXX"));
     ```

     where mdm.zenprise.com is the name of the Device Manager server and zdm is the Device Manager instance name.

   • For a cloud deployment implementation:

     ```
     $soap_url = "<instance>.zc.zenprise.com/<instance>/services/EveryWanDevice?wsdl";
     $client = new SoapClient(null, array(  
        'location' => $soap_url,
        'url' => "<instance>.zc.zenprise.com",
        'login' => "demo",
        'password' => "XXXXX"));
     ```

     For example:

     ```
     $soap_url = "abc.zc.zenprise.com/abc/services/EveryWanDevice?wsdl";
     $client = new SoapClient(null, array(  
        'location' => $soap_url,
        'url' => "abc.zc.zenprise.com",
        'login' => "demo",
        'password' => "XXXXX"));
     ```
6. Edit the devices.csv file and add the serial numbers of all corporate devices, on separate lines.

7. Open a DOS command prompt and cd to c:\temp and run the following command tagDevice.php as follows:

   c:\temp>c:\php5\php.exe tagDevice.php
   device:7R043870A4S is a personal asset
device:82835PLWY7K is a personal asset
device:88025X9PA4T is a personal asset
device:880277VSA4S is a personal asset
device:99000052027603 is a personal asset
device:A1000013555FD9 is a personal asset
device:A10000138B2613 is a personal asset
device:A1000017B0A311 is a personal asset
device:C329030326CC33E is a corporate asset
device:GB0262YCETV is a personal asset
device:GB0289L3ETV is a personal asset
c:\temp>

To configure a device tagging script to run as a repeating task

1. Create a file named tagDevice.cmd under c:\temp (where you previously had copied tagDevice.php and devices.csv) and add the following line: cd c:\temp &&
c:\php5\php.exe tagDevice.php

2. Create an MS Scheduled task to execute this command once every minute (/MO 1). For example: c:\> schtasks /create /TN tagDevice c:\temp\tagDevice.cmd
   /MO 1

3. Query the tasks to verify that it exists by executing the following command: c:\
   schtasks /query /TN tagDevice

4. To delete the task, execute this command: c:\ schtasks /delete /TN tagDevice
Tagging User Devices Manually

There are three ways you can manually tag a device:

- Tag the device during the invitation-based enrollment process (iOS-only).
- Tag the device during the Self Help Portal enrollment process.
- Tag the device by adding device ownership as a device property (any device).

When you enroll an iOS device. You have the option of tagging the device as either corporate- or employee-owned. When using the Self Help Portal to self-enroll a device, you can also tag the device as either corporate- or employee-owned. You can also tag a device manually by adding a property to the device from the Devices tab in Device Manager, creating the property named Device Ownership and choosing either Corporate or Employee.
To add a device to Device Manager manually

The Device Manager server repository database stores a list of mobile devices. Each mobile device is defined by a unique serial number and/or IMEI. There are a number of methods to populate Device Manager with your devices:

- Adding devices manually.
- Import a list of devices from a file by using Device Provisioning tool (Windows Mobile and Symbian devices only) or Device Auto Discovery (only available with the Secure Device option).

1. Click New device and then select the device type.
To import a list of devices by using a file

Develop a text file according to the following format by using a utility application such as a text editor, spreadsheet application, or note taker.

<table>
<thead>
<tr>
<th>Element</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial Number</td>
<td>Device serial number (required if IMEI is not given)</td>
</tr>
<tr>
<td>IMEI</td>
<td>Device IMEI identifier (required if serial number is not given)</td>
</tr>
<tr>
<td>Operating System Family</td>
<td>Required to be WINDOWS, ANDROID, or iOS.</td>
</tr>
<tr>
<td>Property name 1</td>
<td>Optional</td>
</tr>
<tr>
<td>Property value 1</td>
<td>Optional</td>
</tr>
<tr>
<td>Property name (n)</td>
<td>Optional</td>
</tr>
<tr>
<td>Property value (n)</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Many mobile operators or device manufacturers provide lists of authorized mobile devices, and you can utilize these to avoid having to enter a long list of mobile devices manually. Device Manager supports an import file format that is common to all three of the supported device types.

Note the following:

- File charset must be UTF-8/
- Semi-colon (;) is used as the field delimiter so it must be escaped if it is present in the data.
- For iOS device import, Serial Number is mandatory. Serial Number is the identifier for iOS devices.

For example:

```
1050BF3F517301081610065510590391;WINDOWS;propertyN;propertyV\;test\;1\;2;prop 2;prop2 value
2050BF3F517301081610065510590392;25244201625379902;ANDROID;propertyN;propertyV$*&&ééétest
3050BF3F517301081610065510590393;iOS;test;
4050BF3F517301081610065510590393;;iOS;test;
55244201625379903;ANDROID;test.testé;value;
```
To import a task file

1. Click the Import tab.

2. Browse to the corresponding provisioning file.

3. Click Import.
Locking a Device Remotely

If the device is lost, but you are not sure it was stolen, you can remotely "lock" the device. To do so, select the device in Device Manager and then on the Security menu, click Lock.

For Android and Windows Mobile devices, the system will then generate a PIN code that will be set in the device if the user had not set a PIN code already. To access the device, the user will have to type that PIN code.

When the device is found again, you can remove the lock by using the Cancel the lock option.
Revoking a Device Certificate

To prevent access from a device to the server, select the Revoke option and then click Yes in the confirmation window.

To re-authorize a device that was previously forbidden to connect to the server, select the Authorize option and then click Yes in the confirmation window.

**Note:** This capability requires the Device Manager Secure Device option.

- To revoke an Android device. Revoking an Android device prohibits any further connection from the device. The action blocks the device from connecting to the Device Manager server and corporate network, but will not remove any corporate data. To remove all corporate data and information from a device, you also need to selectively wipe the device.

- To revoke an iOS device. Revoking an iOS device prohibits any further connection from the device. If the device attempts to connect to Device Manager, the mobile device management (MDM) profile and all packages deployed to the device are removed. The device is barred from re-enrollment unless it is re-authorized by an administrator.

  **Note:** Revoking an iOS device effectively performs a selective wipe on the device.

- To revoke a Symbian device. Revoking a Symbian device prohibits any further connection from the device.
Selectively Wiping a Device

You can perform a selective wipe in Device Manager if you only want to clear corporate data from the device while retaining personal information and selected settings. A selective wipe removes the mobile device management (MDM) profiles. All packages pushed by Device Manager to the device are also removed. The device can be re-enrolled at a future time.

Select Selective Wipe command from the Devices tab > Security menu > Selective wipe. Selective Cancel Wipe to undo the operation request.

Selective Wipe for iOS and Android Devices

Performing a selective wipe from the if you only want to clear corporate data from the device while retaining personal information and selected settings. The MDM profiles and all packages pushed by Device Manager to the hand held are removed. The device can be re-enrolled at a future time.

Note: Selectively wiping an Android devices does not completely disconnect the device from Device Manager and a user's corporate network. In order to break the connection between the device and the corporate network, you also need to revoke the Android device.

Selective Wipe for Windows 8 Devices

When you perform a selective wipe on a Windows 8 device from Device manager, it will remove all contents from the currently logged on user’s profile folder.

Selective Wipe for Windows Phone 8 Devices

When you selective wipe a Windows Phone 8 device using Device Manager, the following is removed from the device:

- The enterprise token that allows apps to be installed on the device by Device Manager.
- All Device Manager certificates.
- All Device Manager configurations that have been deployed to the device.
Requesting a Full Wipe for a Device

If a device is stolen or lost, you can send a request to have all data on a device be erased. For Android devices, this also includes the option to include any memory cards.

To fully wipe a device, from the Devices tab inside the XenMobile Device Manager web console, select Security > Full Wipe.

**Note:** Erasing a device may not complete in full if the “current holder” of the device has time to turn the device off before the content of the memory card is completely deleted. As such, they may still have access to data on the device.

If the wipe of the device is not done and it is retrieved, you can cancel the wipe command by selecting the Cancel wipe menu item.

For Android devices, you can choose to wipe only the device, which removes any internally stored data, or choose to wipe the device, plus any externally connected storage data (memory cards).

For Windows Phone 8 Devices, a full wipe removes all MDM information plus all user data, including all personal content such as apps, emails, contacts, and media files.

For Windows Mobile devices that are not running Windows Mobile 6 or later, after wiping, it may be required to send the device back to the manufacturer to reload the original operating system and/or software.
Managing Security and Identity

In Device Manager, you use certificates to create secure connections and authenticate users.

To establish a secure connection, a server certificate is required at one end of the connection. A root certificate of the Certificate Authority (CA) that issued the server certificate is required at the other end.

- Server certificate. A server certificate certifies the identity of a server. Device Manager requires this type of digital certificate.

- Root certificate. A root certificate identifies the CA that signed the server certificate. The root certificate belongs to the CA. The user device requires this type of digital certificate to verify the server certificate.

You can submit certificates for signing to a CA who signs the certificate and returns it to you.

In addition to certificates, you can configure security and identity in Device Manager in the following ways:

- Configure Device Manager by using Microsoft Certificate Services to generate user certificates for certificate-based authentication with WIFI, VPN, and Exchange ActiveSync profiles. You can also configure Device Manager as the CA to generate requests and to issue device identity certificates with Microsoft Certificate Services.

- Configure your own SAML service and identify provider in Device Manager. The SAML-based infrastructure can authenticate users and their mobile devices.

- Include Secure Device in your license that is activated automatically when you install Device Manager. Secure Device provides a strong level of security for user devices.

- Enable Strong ID that is a form of two-factor authentication. This provides extra security when enrolling devices in Device Manager.

- Configure filters in Device Manager that work with Network Access Control. Filters set users devices to be either compliant or not compliant. If a device is not compliant, the device is blocked from accessing the internal network.
About XenMobile PKI

The XenMobile Public Key Infrastructure (PKI) Integration feature allows you to manage the distribution and life-cycle of security certificates used on your devices with great flexibility.

The main feature of the system is the PKI Entity. A PKI entity models back-end component for PKI operations. That component may be either local to XenMobile (internal) or a part of your corporate infrastructure (external, such as a Microsoft, RSA, or OpenTrust PKI). The PKI entity handles the back-end certificate issuance and revocation. It is the authoritative source for the certificate’s status. The XenMobile configuration will normally contain exactly one PKI Entity per back-end PKI component.

The second feature is the Credential Provider. A Credential Provider is a particular configuration of certificate issuance and life-cycle. It will control things like the certificate’s format (subject, key, algorithms) and the conditions for its renewal or revocation, if any. The Credential Providers delegate operations to the PKI Entities. In other words, while Credential Providers control when and with what data PKI operations are undertaken, PKI Entities control how those operations are performed. The XenMobile configuration will normally contain many Credential Provider per PKI Entity.

The third feature of the system are Server Certificates. Server Certificates are X.509 certificates used functionally by the PKI Entity or the Credential Provider configurations.
Server Certificates

Server certificates are certificates used functionally by the XenMobile server that are uploaded into the Device Manager web console in the PKI integration section of the Options dialog box. They include CA (Certificate Authority) certificates, RA (Registration Authority) certificates, certificates for client authentication with other components of your infrastructure. In addition, you may use it as a storage for certificates you wish to deploy to devices. This will especially apply to CAs used to establish trust on the device.

XenMobile may or may not possess the private key for a given certificate. For some certain usages, XenMobile will require the private key, whereas for others, it will not. Each certificate you upload will be represented by an entry in the Server Certificates table, summarizing its contents. Later on, when you configure PKI integration components that require a certificate, you will be prompted to choose from a list of those Server Certificates that satisfy the context-dependent criteria.

For example, you might want to configure XenMobile to integrate with your Microsoft CA. The connection to the Microsoft CA should be authenticated using a client certificate.

You can upload the CA certificate (without the private key) the CA will use to sign requests, and an SSL client certificate (with the private key) client authentication. When configuring the Microsoft CA entity, you need specify the CA certificate, which you can then select from a drop-down list with all Server Certificates that are CA certificates (context-dependent criterion). Likewise, when configuring client authentication, you can select from a drop-down list with all the Server Certificates for which XenMobile has the private key (context-dependent criterion).
To import a server certificate

XenMobile supports the following input formats for certificates:

- PEM or DER-encoded certificate files
- PEM or DER-encoded certificate files with associated PEM or DER-encoded private key file
- PKCS#12 key stores (P12; also known as PFX on Windows)
- Java Key Store (JKS) and Extended Java Key Store (EJKS)

Key stores, by design, can contain multiple entries, so when you loading from a key store, you will be prompted to specify the entry alias identifying the entry you wish to load. If you do not specify an alias, the first entry from the store will be loaded. Since PKCS12 files usually contain only one entry, you should leave the alias empty for those files.

When importing a certificate, either from a file or a key store entry, XenMobile will attempt to construct a certificate chain from the input, and will import all certificates in that chain (creating a Server Certificate entry for each). This will only work if the certificates in the file or key store entry really do form a chain, such as if each subsequent certificate in the chain is the issuer of the previous one. You can add an optional description for the imported certificate for heuristic purposes. The description will only be attached to the first certificate in the chain (you can update the description of the remainders later on).

1. From the Device Manager web console, click Options.
2. In the XenMobile Server Options dialog box, from the left side select PKI > Server Certificate.
3. Click Upload Certificate to import a certificate.
4. From the Certificate Type list, select either Certificate or Keystore.
5. Next, click Choose File to select a certificate.
6. Next, click Choose File to select a private key file for the certificate.
7. Enter an optional description, and then click Upload.
Updating a Certificate

XenMobile only allows one certificate per public key to exist in the system at any given time. If you attempt to import a certificate for the same key pair as an already imported one, you will be presented with the option to either replace the existing entry or to delete it.

To most effectively update your certificates, simply upload the new one in the Device Manager web console’s Options dialog box, under PKI > Certificates. When a Server Certificate entry is updated, components that were using the previous one will automatically switch to using the new one. Likewise, if you have deployed the Server Certificate on devices, it will automatically be updated on the next deployment.
PKI Entities

A XenMobile Public Key Infrastructure (PKI) Entity configuration represents a component performing actual PKI operations (issuance, key escrow, revocation, status information). These components may either be internal to XenMobile, in which case they’re called discretionary, or external to it, if they are part of your corporate infrastructure.

XenMobile supports the following three types of PKI entities:

- Discretionary CAs
- Generic PKIs (GPKI)
- Microsoft Certificate Services
Common PKI Concepts

Regardless of its type, every PKI Entity is said to have a subset of the following capabilities:

- **sign** Issuing a new certificate, based on a Certificate Signing Request.
- **fetch** Recovering an existing certificate and key pair.
- **revoke** Revoking a client certificate.

Table 1. PKI Capabilities

<table>
<thead>
<tr>
<th>PKI Type</th>
<th>Capability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discretionary</td>
<td>Sign</td>
</tr>
<tr>
<td>PKI</td>
<td>The adapter is capable of taking Certificate Signing Requests, transmitting them to the PKI and returning newly signed certificates.</td>
</tr>
<tr>
<td>Microsoft</td>
<td>Sign</td>
</tr>
</tbody>
</table>

About CA Certificates

When configuring a PKI entity, you will have to inform XenMobile which CA certificate is going to be the signer of certificates issued by (or recovered from) that entity. One and the same PKI entity may return (fetched or newly signed) certificates signed by any number of different CAs; the certificate of each of these CAs will have to be provided as part of the PKI entity configuration, by uploading it to the Server Certificates repository and then referencing them in the PKI entity. For discretionary CAs, this will implicitly be the signing CA certificate, but for external entities, you will have to specify it manually.

**Note:** XenMobile will verify that the actual issuer of a certificate newly obtained through a sign or fetch operation matches the purported, configured issuer. An error will be raised if this is not the case.
Discretionary CAs

A Discretionary CA is created by providing XenMobile with a CA certificate and the associated private key. XenMobile will handle certificate issuance, revocation, and status information internally, according to the parameters you specify. However, XenMobile will never store the private keys of issued certificates, and so will not offer escrow services. Status information for certificates issued by a discretionary CA.

When configuring a Discretionary CA, you will have the option to activate OCSP support for that CA. If, and only if, OCSP support is enabled, the CA will add an id-pe-authorityInfoAccess extension to the certificates it issues, pointing to XenMobile’s internal OCSP Responder located at:

https://[server]/[instance]/ocsp

When configuring the OCSP service, you will have to specify an OCSP signing certificate for the Discretionary Entity in question. You can use the CA certificate itself as the signer. If you wish to avoid the unnecessary exposure of your CA’s private key (recommended), you will have to create a delegate OCSP signing certificate, signed by the CA certificate and including an id-kp-OCSPSigning extendedKeyUsage extension.

The XenMobile OCSP Responder service supports Basic OCSP responses and the following hashing algorithms in requests:

- SHA-1
- SHA-224
- SHA-256
- SHA-384
- SHA-512

Responses are signed with SHA-256 and the signing certificate’s key algorithm (DSA, RSA or ECDSA).
Generic PKI (GPKI)

The Generic PKI (GPKI) protocol is a proprietary XenMobile protocol running atop a SOAP Web Service layer for purposes of uniform interfacing with various PKI solutions. The GPKI protocol defines three fundamental PKI operations:

- **sign** The adapter is capable of taking Certificate Signing Requests (CSR), transmitting them to the PKI and returning newly signed certificates.

- **fetch** The adapter is capable of retrieving (recoverying) existing certificates and key pairs (depending on input parameters) from the PKI.

- **revoke** The adapter is able to cause the PKI to revoke a given certificate.

The receiving end of the GPKI protocol is the GPKI Adapter. The adapter translates the fundamental operations to the specific type of PKI for which it was built (in other words, there is a GPKI Adapter for RSA, another for OpenTrust, and so on).

Support for each of the PKI operations in an adapter is optional. If an adapter supports a given operation, it is said to have the corresponding capability (sign, fetch or revoke). Each of these capabilities may be associated with a set of user parameters.

User parameters are parameters that are defined by the GPKI adapter for a specific operation and for which you need to provide values to XenMobile. Which operations the adapter supports (which capabilities it has) and which parameters it requires for each of them is determined by XenMobile by parsing the WSDL. The connection between XenMobile
Microsoft Certificate Services

XenMobile interfaces with Microsoft Certificate Services through its web enrollment interface. It only supports issuing new certificates through that interface (the equivalent of the GPKI sign capability).

To create a Microsoft CA PKI Entity in XenMobile, you must specify the base URL of the Certificate Services web interface. The connection between XenMobile and the Certificate Services web interface may optionally be secured using SSL client certificate authentication.

**Note:** This integration method is historical and limited. It will be migrated to the GPKI protocol in the future.
Migrating Previous PKI Configurations

Since the new XenMobile PKI integration capabilities have been significantly enhanced, migrating to the new system is not automatic. If you had used PKI configurations in previous versions, you will be able to continue to use these in 8.0 without changes, but if you wish to make use of the new capabilities, you will have to manually upgrade existing PKI entities.

Your pre-8.0.1 PKI entities (Microsoft CA or GPKI) will appear in the list of PKI entities, but will be marked as not ready to be used, indicated by a red icon in the Valid column in the Options dialog box, under PKI > Entities.

To ready the entity for 8.10 usage, edit the entity and provide the missing settings (the system will indicate which settings are missing when you try to save the configuration). This process requires providing the CA certificate(s) for the entity.
Credential Providers

Credential Providers are the actual configurations you will use in the various parts of the XenMobile system. They define the sources, parameters, and life-cycles of your certificates, whether these are part of device configurations or stand-alone, that is pushed as is, to the device.

Figure 1. Certificates Lifecycle

The certificates’ life-cycle is constrained by the device enrollment. That is, no certificates are issued before enrollment, although some may indeed be issued as part of enrollment, and all certificates issued within the context of one enrollment are revoked when the...
One Credential Provider configuration may be used in multiple places, to the effect that configuration may govern any number of certificates at the same time. The unicity, then, is on the deployment resource and the deployment: if the Credential Provider P is “deployed” to device D as part of the configuration C, then P’s issuance settings will determine the certificate that is deployed to D, its renewal settings will apply when C is updated, and its revocation settings will apply when C is deleted or D is revoked.

With the aforementioned in mind, the Credential Provider configuration:

- Determines the source of certificates — that is, which PKI Entity certificates will be obtained from
- Determines the method using which certificates are obtained — signing a new certificate or fetching (recovering) an existing certificate and key pair
- Determines the parameters for the issuance or recovery (for example, CSR parameters such as key size, key algorithm, distinguished name, certificate extensions, and so on)
- Determines the manner in which certificates are delivered to the device
- Determines revocation conditions. While all certificates are revoked when the management relationship is severed, the configuration may specify an earlier revocation, for instance when the associated device configuration is deleted. In addition, under some conditions the revocation of the associated certificate in XenMobile may be sent to the back-end PKI; that is, its revocation in XenMobile may cause its revocation on the PKI
- Determines renewal settings. Certificates obtained through a given Credential Provider may be automatically renewed when they near expiration, or, separately from that, notifications may be issued when that expiration approaches.

To what extent various configuration options are available will mainly depend on which type of PKI Entity and issuance method are selected for a Credential Provider.
Methods of Certificate Issuance

There are two fundamental methods of obtaining a certificate, which in this context shall be called methods of issuance:

- **SIGN.** With this method, the issuance involves creating a new key pair, creating a Certificate Signing Request (CSR) for the key pair, and submitting it to a CA for signature.

- **FETCH.** With this method, the issuance (from the point of view of XenMobile) is in actuality a recovery of an existing certificate and key pair.

A Credential Provider uses exactly one of these methods; which method is selected impacts which configuration options are available. Notably, CSR configuration and distributed delivery are only available if the issuing method is sign. If the certificate is fetched, it is always sent as a pkcs#12 to the device (equivalent to centralized delivery mode for the sign method).

Which issuing methods are available for a Credential Provider will depend on the capabilities the PKI Entity it uses supports.
Certificate Delivery

An important notion is the delivery mode of certificates. The delivery is independent of the issuance, although it only applies when the issuing mode is newly issued [sign], not recovered [fetch] from the PKI).

Two modes of certificate delivery are available: centralized and distributed. Distributed mode uses the SCEP protocol and is only available in situations where the client supports the protocol, and is even mandatory in some situations.

For a Credential Provider to support distributed (SCEP-assisted) delivery, a special configuration step is necessary: setting up Registration Authority (RA) certificates. Those are required because when using the SCEP protocol, XenMobile acts like a delegate (a registrar) to the actual CA, and must prove to the client that it has the authority to act as such. That authority is established by providing XenMobile with the aforementioned certificates.

Two distinct certificate roles are required (although one and the same certificate can fulfill both): RA signature and RA encryption. The constraints for these roles are as follows:

- The RA signing certificate must have the X.509 key usage digital signature.
- The RA encryption certificate must have the X.509 key usage key encipherment

To configure the Credential Provider’s RA certificates, you must first upload them to the Server Certificates repository, and then link to them in the Credential Provider.

A Credential Provider is considered to support distributed delivery if, and only if, it has a certificate configured for each of the aforementioned roles. Each Credential Provider can be configured to either prefer centralized mode, to prefer distributed mode, or to require distributed mode. The actual result will depend on the context: if the context does not support distributed mode, but the Credential Provider requires it, deployment will fail. Likewise, if the context mandates distributed mode, but the Credential Provider does not support it, deployment will fail. In all other cases, the preferred setting will be honored.

Table 1. SCEP Distribution Availability

<table>
<thead>
<tr>
<th>Context</th>
<th>SCEP supported</th>
<th>SCEP required</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS Profile Service</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>iOS MDM enrollment</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>iOS configuration profiles</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>SHTP enrollment</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>SHTP configuration</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Windows Phone enrollment</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Windows Phone configuration</td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>
Certificate Revocation

There are three separate aspects to a certificate’s revocation, three types of revocation: internal revocation, externally propagated revocation and externally induced revocation.

- **Internal revocation** Internal revocation affects the certificate’s status as maintained by XenMobile (in its database). This status is taken into account when XenMobile evaluates a certificate presented to it, or when it has to provide OCSP status information for some certificate). The Credential Provider configuration determines how this status is affected under various conditions. For instance, the Credential Provider may specify that certificates obtained through it should be (flagged as) revoked when they have been deleted from the device.

- **Externally propagated revocation** Also known as “Revocation from XenMobile”, this type of revocation applies to certificates obtained from an external PKI, and means that the certificate will be revoked on the PKI when it is internally revoked by XenMobile (under the conditions defined by the Credential Provider configuration). The call to perform the revocation requires a revoke-capable GPKI Entity.

- **Externally induced revocation** Also known as “Revocation from PKI”, this type of revocation also only applies to certificates obtained from an external PKI, and means that whenever XenMobile evaluates a given certificate’s status, it will query the PKI as to that status, and, if the PKI returns that the certificate is revoked, will internally revoke it. This mechanism uses the OCSP protocol.

These three types are not exclusive, but rather apply together: the internal revocation is caused either by an external revocation or by independent findings, and in turn it potentially effects an external revocation.
Certificate Renewal

A certificate renewal is the combination of a revocation (of the existing certificate) and an issuance (of another certificate).

Note that XenMobile will first attempt to obtain the new certificate before revoking the previous one, in order to avoid discontinuation of service if the issuance fails. If distributed (SCEP-supported) delivery is used, the revocation will also only happen once the certificate has been successfully installed on the device; otherwise, the revocation will occur before the new certificate is sent to the device and independently of the success or failure of its installation.

The revocation configuration requires that you specify a certain duration (in days); when the device connects, the server verifies whether the certificate’s NotAfter date is later than the current date minus the specified duration. If it is, a renewal is attempted.
To create a credential provider using discretionary CA entities

Configuring a Credential Provider varies mostly as a factor of which issuing entity and which issuing method are selected for it. You can distinguish between Credential Provider using an internal entity, such discretionary, and those using an external entity, such as Microsoft CA or GPKI.

This task shows you how to create a discretionary entity. The issuing method for a discretionary entity is always sign, meaning that with each issuing operation, Device Manager will sign a new key pair with the CA certificate selected for the entity whether the key pair is generated on the device or on the server will depend on the selected distribution method.

1. In the Device Manager web console, click Options.

2. In the Options dialog box, select PKI > Credential Provider.

3. In the Define a new credential provider dialog box, on the General tab, enter the following information
   a. Credential Provider name. Type a unique name for the new provider configuration. This name will be used subsequently to refer to the configuration in other parts of the administration console.
   b. Description. An optional description for the configuration.
   c. Issuing method. The method that the system should obtain certificates from the configured entity. Select a discretionary entity.

4. On the CSR tab, you configure the parameters for the key pair that will be created during issuance, as well as the parameters of the new certificate. Enter the following information:
   a. Key algorithm. The key algorithm for the new key pair. Available values are RSA, DSA and ECDSA.
   b. Key size. The size, in bits, of the key pair. Note that which values are permissible depends on the type of key (for instance, the maximum size for DSA keys is 1024 bits). To avoid false negatives (which will be dependent on the underlying hardware and software), Device Manager will not enforce key sizes. You should always test Credential Provider configurations in a test environment before activating them in production.
   c. Signature algorithm. The signature algorithm for the new certificate. Values are dependent on the key algorithm; in this case, Device Manager will limit your choices to matching values.
   d. Subject name. The Distinguished Name (DN) of the new certificate’s subject. For example: CN=${user.username}, OU=${user.department}, O=${user.companyname}, C=${user.c}
e. Subject alternative names. X.509 subject alternative names. To create a new entry, click on New alternative name and then click on the first column to select the type of alternative name from those available. Last, enter a value in the second column. Note that as for the subject DN, you can use Device Manager macros in the value field.

5. Next, click on the Distribution tab. Because the Credential Provider uses a Discretionary CA Entity, the CA certificate for the Credential Provider will always be the CA certificate configured on the entity itself; it will be presented here for mere consistency with configurations that use external entities.

The second element on this tab is the configuration of certificate delivery. If you have defined RA certificates at the entity level, they will be filled by default here, but you can change them if you desire (but that the constraints on RA certificates still apply).

You can then select the delivery mode for certificates obtained from this entity. If you select the Prefer centralized delivery mode, RA certificates are optional; otherwise, they're mandatory.

6. Next, click on the Revocation tab. In this tab, you can configure under what conditions Device Manager should (internally) flag certificates issued through this provider configuration as revoked. You can also instruct the system to send a notification when it flags a certificate as revoked. Do this by selecting a template for the event type Certificate revoke. Device Manager will create a default template for that type, but you can edit it or create others. Note that the revocation configured here will be what determines the responses from the Device Manager OCSP Responder for certificates created with this configuration, if OCSP support is enabled for the PKI Entity.

7. Next, click on the Renewal tab. In this tab, you can configure the renewal of certificates obtained through this configuration. Two basic operations can be configured:

a. Renewing the certificate, optionally sending a notification when this is done (notification on renewal), and optionally excluding already expired certificates from the operation. Note that “already expired” in this case means that their NotAfter date is in the past; not that they already have been revoked. Device Manager will not renew certificates once they have been internally revoked.

b. Issuing a notification for certificates that near expiration notification before renewal).

To have notifications sent for either case, simply specify a Notification Template for the appropriate event type. The event type for the former is Certificate is renewed; for the latter, Certificate will expire. Device Manager will create default Notification Templates for both these event types, but you can modify them or create new ones.

Note that renewal takes precedence over notification before renewal. That is, if at a given moment Device Manager determines that a certificate must be renewed, it will not also send a notification before renewal (instead, the notification on renewal, if any configured, will be used). You should configure a greater period for the notification before renewal if you imperatively need both to be sent. Notifications before renewal will only be sent at most once for a given certificate.

8. Click Create.
To create a credential provider using external PKI entities

When you create a Credential Provider using an external (Microsoft or GPKI) entity, the main difference in configuring the two is the issuing method for the provider; which methods are available depends on the capabilities of the selected PKI entity:

- If you opt for using a Microsoft CA entity for your Credential Provider, your choice of issuing method will be limited to sign. The sign method of issuance involves creating a new key pair, creating a Certificate Signing Request (CSR) for the key pair, and submitting it to a CA for signature. (You will be prompted to select which certificate template to use for issuance). You must choose a value from those you have defined during the creation of the PKI entity. This is the template name that will be sent to the Microsoft CA along with the Certificate Signing Request during issuance. A Credential Provider can only use one template. If you want to issue certificates based on different templates, create a Credential Provider configuration for each of them.

- If you opt for using a GPKI entity for your Credential Provider, your choice of issuing method will depend on which capabilities are supported by the adapter. If the GPKI adapter defines user parameters for the selected capability, you will be presented with an interface to specify values for each of those. The parameters are specific to a capability; different capabilities have different sets of user parameters.

Note: For information on configuring Microsoft Certificate Services to work with Device Manager, see Configuring Device Manager with Microsoft Certificate Services

1. In the Device Manager web console, click Options.

2. In the Options dialog box, select PKI > Credential Provider.

3. In the Define a new credential provider dialog box, on the General tab, enter the following information

   a. Credential Provider name. Type a unique name for the new provider configuration. This name will be used subsequently to refer to the configuration in other parts of the administration console.

   b. Description. An optional description for the configuration.

   c. Issuing method. The method that the system should obtain certificates from the configured entity. Select a Microsoft or GPKI entity.

4. On the CSR tab (sign method only), you can configure the parameters for the key pair that will be created during issuance, as well as the parameters of the new certificate:

   a. Key algorithm. The key algorithm for the new key pair. Available values are RSA, DSA and ECDSA. Key size The size, in bits, of the key pair. Note that which values are permissible depends on the type of key (for instance, the maximum size for DSA keys is 1024 bits). To avoid false negatives (which will be dependent on the underlying hardware or software), Device Manager will not enforce key sizes. You should always test Credential Provider configurations in a test environment before
b. Signature algorithm. The signature algorithm for the new certificate. Values are
dependent on the key algorithm; in this case, Device Manager will limit your
choices to matching values.

c. Subject name. The Distinguished Name (DN) of the new certificate’s subject. The
format the system expects is as described in [5]. Note that you can use Device
Manager macros for the the DN field values. For example:

\[CN=${user.username}, OU=${user.department},
O=${user.companyname}, C=${user.c}\]

d. Subject alternative. names X.509 subject alternative names. To create a new entry,
click on “New alternative name”; then click on the first column to select the type
of alternative name from those available; and finally enter a value in the second
column. Note that as for the subject DN, you can use Device Manager macros in the
value field.

5. Next, click the Distribution tab. Here, you are required to specify the issuer CA of the
certificates returned by the PKI entity in the configuration you have selected. You will
be offered to choose one of the CA certificates defined on the entity. If the issuing
method for this Credential Provider is sign, you will also be able to configure the
delivery method, since the fetch method retrieves the key pair from the PKI server and
hence there is no key generation involved at all, distributed key generation is not
available with that method.

6. Next, select the Revocation XenMobile tab. On this tab, you can configure the
conditions and actions for the internal revocation of certificates. You can opt to have
certificates revoked under the following conditions:

a. When they are removed from the device, that is, either when the system detects
that they have been removed from the device without server interaction, or when
the server has removed them subsequent to the scheduling of a removal command.

b. When they are updated on the device, that is, when they are replaced with a newer
certificate for the same function.

c. When the enrollment is revoked by the administrator.

d. When the device is deleted.

e. Or any combination of these.

You can further opt to have a notification sent when the revocation action is
undertaken; to do so, simply configure a notification template for the appropriate
event type (Certificate revoke).

In addition to these conditions, since the certificates obtained through this
configuration will have come from an external source, you can opt to propagate the
revocation status externally (the common case would be to propagate it to the PKI that
issued the certificates, but your choice is not restricted in that matter). The
propagation is achieved using a GPKI entity with the revoke capability; the interface
will propose you the list of revoke-capable GPKI entities that exist in the system. If the
selected entity defines user-parameters for the revoke operation, you will be prompted
to enter values for them. You can use Device Manager macros for the values.

7. Next, select the Revocation PKI tab.
In this tab, you can configure the system to perform external certificate status checks for certificates issued through this CredentialProvider configuration. The checks are performed using the OCSP protocol [1] and take place when a deployment is initiated. For the checks to occur, the back-end PKI must insert corresponding OCSP responder address extensions (ASN.1 OID: 1.3.6.1.5.5.7.1.1) in the certificates it issues. If that is not the case, the setting will be silently ignored.

As part of the OCSP protocol, the initiator of the OCSP request (in this case, XenMobile) must be able to validate the OCSP responder’s (likely your PKI server) signing certificate. To that effect, as part of the external revocation check configuration, you must specify the CA certificate of your PKI’s OCSP Responder’s signing certificate. The CA certificate must be uploaded to the Server Certificates repository so that you can select it in the drop down. Its private key is not required for this purpose.

Note that OCSP Responder certificates are usually either the CA certificate itself (that is, the CA that signed the certificate the status of which is queried), or a certificate signed directly by that CA. It that sense, specifying that CA certificate in this section will usually be adequate.

You can further define what actions XenMobile should undertake in the event that the OCSP verification yields a status indicating that the certificate in question was revoked. If that is the case, you can opt to:

- Do nothing.
- Remove the corresponding configuration from the device, that is, the configuration the certificate in question was deployed as part of.
- Revoke the enrollment and wipe the device.

In addition to the action you opt for, you can choose to have a notification sent in that case, by selecting a notification template for the appropriate event type (Certificate revoked by PKI). The external revocation and the internal revocation configured in the tab before are complementary, in the sense that if the external revocation check yields a revoked status and you have opted, for instance, to revoke the entire enrollment in that case, then the settings you have specified in the Revocation XenMobile tab will apply to all other certificates present on the device. The same thing goes for all certificates that were part of the same configuration if you have merely chosen to remove the configuration the certificate was deployed as a part of.

8. Next, select the Renewal tab.

On this tab, you can configure the renewal of certificates obtained through this configuration. Two basic operations can be configured:

- Renewing the certificate, optionally sending a notification when this is done (notification on renewal), and optionally excluding already expired certificates from the operation. Note that ‘already expired’ in this case means that their NotAfter date is in the past; not that they already have been revoked. XenMobile will not renew certificates once they have been internally revoked.
- Issuing a notification for certificates that near expiration (notification before renewal).

To have notifications sent for either case, simply specify a Notification Template for the appropriate event type. The event type for the former is Certificate is renewed; for the latter, Certificate will expire. XenMobile will create default Notification Templates for both these event types, but you can modify them or create new ones.
To create a credential provider using external PKI entities

It is important to note that renewal takes precedence over notification before renewal. That is, if at a given moment XenMobile determines that a certificate must be renewed, it will not also send a notification before renewal (instead, the notification on renewal, if any configured, will be used). You should configure a greater period for the notification before renewal if you imperatively need both to be sent. Notifications before renewal will only be sent at most once for a given certificate.

9. Click Create.
Configuring a SAML Service Provider

Device Manager supports configuration of your own Security Assertion Markup Language (SAML) service and identity provider and SAML-based infrastructure to authenticate users and their mobile devices. With your own SAML configuration, you do not need to pre-provision user account information in Device Manager, such as user names, group association, or other directory attributes. SAML implementations allow network administrators to provide single sign on access to servers, web sites, and apps.

SAML Use Cases

Initial Registration of Mobile Device

The Device Manager agent should be able to register the device with the Device Manager server using the SAML token. No pre-provisioning of the user name, group association, or other directory attributes in the Device Manager server should be required.

Ongoing Authentication and Authorization for Policy Updates and Device Controls

Once the mobile device manager (MDM) agent registers the device and receives the initial policy updates, the mobile device must be able to re-authenticate with the IDP server each time the SAML token expires to receive policy updates and allow for security actions, such as lock, revoke, wipe devices, and so on, including know when the user has changed groups that might impact MDM policies, or proper authorization.

Single Sign-On With Other SAML-Enabled Applications

After the mobile device registers with the SAML token, other SAML-enabled applications should be able to authenticate the user without prompting for the corporate credential to provide a single sign-on user experience. It needs to be determined whether all SAML enabled applications, including popular ones, such as SF.com, Google Apps, Microsoft365, Box.net, and so on can be supported or only applications that are managed by Device Manager or written to the App SDK.

Decommissioning Devices and Removing Users

When a user is removed from the corporate directory; for example, the user leaves the organization, there must be a mechanism to deactivate users and decommission the devices in the Device Manager server.
SAML Test Requirements

- Establish a "relying party trust" between the iDP server and the Device Manager Service Provider server, including required certificates for the trust relationship.

- Develop claim attribute mapping with User ID, Group Membership, Email Address, and other directory attributes.

- Device Manager agent requests the SAML token from the customer iDP server and redirects back to Device Manager server for mobile device registration.

- SAML token on mobile device is presented to Device Manager server for device registration; Device Manager server validates the SAML token and extracts directory attributes; the device is registered and the user identity is created properly.

- Device configuration appears as expected in the Device Manager console; for example, as the software inventory.

- All reports list devices and inventory properly.

- Lock and revoke device using the Device Manager console security commands.

- Change the users group association from Group A to Group B. Push out different Device Manager policy updates to the devices for Group A and Group B. Verify that device gets the proper (Group B) policy updates.

- Access other SAML-enabled applications using HTML-based mobile apps to determine if user is prompted for corporate directory credentials to issue a separate SAML token.

- Access other SAML-enabled application using native mobile apps to determine if user is prompted for corporate directory credentials to issue a separate SAML token.

- Remove user from directory, ensure device state is changed to inactive, and user is removed automatically.

- User is able to reactivate by re-registering the device using the same SAML-based process for initial registration.
To add a SAML service provider

1. Click Edit.

2. In the Service Provider Configuration dialog box, click the General tab and then enter the following information:
   - Entity ID. Enter the ID of the SAML Service Providers Entity ID (globally unique name given to a SAML entity). An entity ID is typically rooted in the organization's Primary DNS Domain.
   - Base URL. URL of the SAML Service Provider.
   - Organization name. The name of your company (optional).
   - Organization Description. Description of your company (optional).
   - Organization URL. The URL of your company (optional).

3. On the Main Parameters tab, select the following options:
   - Supported Bindings
     - SAML Redirect. Select if your SAML server has implemented a URL redirect binding.
   - General
     - Sent SAML Request must be signed.
     - Reserved Assertion must be signed.
     - Received assertion must be encrypted.
     - Passive mode enabled (anonymous connection).

4. On the Contacts tab, you can enter the email addresses for the technical, support, and administrative contacts in your organization.

5. On the Certificates tab, you can enter upload a certificate for the SAML connection, as well as the Keystore password for SAML server authentication.

6. Click Save.
To configure a SAML identity provider

1. In the Identity Providers tab, click New.

2. In the General dialog box for the SAML Identity Provider, enter the following information:
   - Metadata URL. Web address used to access the SAML service provider metadata.
   - User domain. Domain under which the SAML metadata URL resides.

3. Click Create.
Configuring General Security Options

You can configure security in the Options dialog box to customize the security features of the service. By default, when Secure Device is included in the license, it is automatically activated during installation, with a strong level of security. If you need to change those parameters, use that dialog box.

- Enforce SSL: Forces devices to communicate using an SSL transport. All HTTP requests from devices will be rejected.
- Strong Authentication: Enabling Strong Authentication generates a Strong ID for devices that is then used as a second factor of authentication during the enrollment process.
- Strong ID Valid Once: Allows Strong ID passcodes to only be used once. When the Strong ID is used once to generate a device certificate, it cannot be reused. Device has to be revoked and re-authorized.
- Certificate Renewal: Sets the renewal time frame for certificates used in Strong Authentication mode. '0' disables the certificate renewal process.
- Always Add Device: Allow to automatically register devices into Device Manager even when Secure Device is activated.
- Block Rooted Android and iOS Enrollment: Enabling this function will block rooted or jailbroken devices from enrolling.
- 8 Char Strong ID: Enables a Strong ID character string that is limited to 8 characters.
- SHP Console for Users: Enables or Disables the Self-Help Console for user management of devices.
- XDM/SHP console max inactive interval: The time (in minutes) between client requests before the server will invalidate a session. 0 means that a session will never timeout.
- iOS agent auto logout (minutes): Length of time before an iOS agent user is logged due to inactivity.
- Enable client cert authentication for iOS: If enabled, iOS enrollment agent will use certificate authentication. If disabled, iOS enrollment agent uses session based authentication.

To enable Strong ID

Strong ID is a form of 2 factor authentication used to provide an extra layer of extra security when enrolling a device.

1. Enable Strong ID from the Options menu on the Security tab in the Device Manager web console. Citrix also recommends that you enable 8 Char Strong ID). At this point, no devices will be able to enroll until the device's serial number or IMEI is known.
2. Add the devices manually (or import) from the Devices tab using the device's serial number or IMEI, which will generate a Strong ID for the device.

3. When a user is ready to enroll, the user needs to call their administrator and give their Serial/IMEI, so the administrator can provide the Strong ID from the device properties.
Configuring Network Access Controls

If you have a Network Access Control (NAC) appliance set up in your network (such as a Cisco ISE), you can enable filters to set devices as compliant or not compliant for NAC-based on rules or properties. If a Device Manager managed device does not meet the specified criteria, and thus is marked Not Compliant, the device will be blocked on your network by the NAC appliance.

To set unmanaged devices as not compliant, enable the associated filter and set to "Not Compliant". The "Implicit Compliant / Not Compliant" filter sets the default value only on devices that are managed by XenMobile. For example, any devices that have a blacklisted app installed and/or are anonymous (not enrolled) are marked as Not-Compliant and will be blocked from your network by the NAC appliance.

The NAC compliance filters are as follows:

- Blacklisted Apps. Device has a blacklisted app installed.
- Rooted Android/Jailbroken iOS Devices
- Revoked Status. Device has been revoked.
- Unmanaged Devices. Device is in an unmanaged state.
- Suggested Apps Only. Device has "suggested" app installed.
- Inactive Devices. Device is in an inactive state.
- Anonymous Devices. Device is anonymous.
- Out of Compliance Devices. Device has property of Out of Compliance set to True.
- Encryption. The device has disk encryption enabled.
- Implicit Compliant/Non-Compliant. Indicates that if none of the above filters match, return device to be compliant or not (according to the option selected).
Configuring Device Manager with Microsoft Certificate Services

You can configure Device Manager with Microsoft Certificate Services to generate user certificates for certificate-based authentication with WIFI, VPN, and Exchange ActiveSync profiles. You can also configure Device Manager as a Registration Authority to generate requests and to issue device identity certificates with Microsoft Certificate Services.

In addition, you can configure Device Manager to use external SSL server certificates and digital signature certificates from other PKI-trusted certificate authorities.

**Caution:** Changing the digital signature certificate or the SSL certificate authority will disable the management of currently enrolled devices and require a re-enrollment across all devices.

Device Manager can make certificate requests to Microsoft Certificate Services through web enrollment to enable certificate-based authentication for WIFI, VPN, and Exchange ActiveSync profiles. Device Manager does this by acting as a client to Microsoft Certificate Services and requesting certificates on behalf of users with enrolled devices. This section describes how to create a Microsoft Certificate Server entity and configure Device Manager to request certificates for users enabling certificate-based authentication.

**Prerequisites**

- Port 443 (default) open from Device Manager to Microsoft Certificate Services server.
- Microsoft KB 980436 patch needs to be installed on Microsoft Certificate Services server.
- Microsoft KB 272175 - Guidelines for configuring client certificate authentication mode for IIS 6.
- Microsoft KB 953461 patch needs to be installed on Microsoft Certificate Services server on Windows 2008 Server Enterprise.
- Web enrollment for Microsoft Certificate Services needs to be enabled.
- SSL enabled on Microsoft Internet Information Services (IIS).
- IIS configured to accept client certificate authentication.
- The client certificate in .p12 format which is used to authenticate against Microsoft Certificate Services should be copied to the Device Manager server and made accessible.
To enable Web enrollment for Microsoft Certificate Services

1. In Administrative Tools, click Server Manager.

2. Under Active Directory Certificate Services, check to see if Certificate Authority Web Enrollment is installed.

3. Select Add Role Services to install Certificate Authority Web Enrollment, if needed.

4. Select Certificate Authority Web Enrollment and then click Next.

5. Click Close or Finish when the installation is complete.
To enable IIS Web services

1. Go to Administrative Tools and click Server Manager.
2. Select Server Roles on the left side.
3. Select the Active Directory Certificate Services role and the Web Server IIS role, and click Install.
4. Close the Server Manager.
To configure Microsoft Internet Information Services for self-signed or external certificates

1. Go to Administrative Tools and click Server Manager.

2. Under Web Server (IIS), under Internet Information Services (IIS), select the host or top of the root and then click Server Certificates.

3. Create a self-signed certificate or import an external certificate.
To configure Microsoft Internet Information Services

1. In Administrative Tools, select Server Manager.

2. Under Web Server (IIS), under Role Services, verify that Client Certificate Mapping Authentication and IIS Client Certificate Mapping Authentication are installed. If not, install these role services.

3. In Administrative Tools, click Internet Information Services (IIS) Manager.

4. In the left-hand pane of the IIS Manager window, select the server running the IIS instance for web enrollment and then click Authentication.


6. Click Sites and then in the right pane, click Bindings.

7. Add an HTTPS binding if one does not exist.

8. Go to Web Server (IIS) > Sites > Default Web Site > CertSrv

9. Click SSL Settings and then click Accept for Client Certificates.
To create a certificate template for XenMobile certificate requests

1. Open the an MMC Console with a domain administrator account and then add a Snap-In for Certificate Templates.

2. Open Certificate Templates.

3. Right-click the User template and then click Duplicate Template.

4. Select Windows 2003 Server for the template type and then click OK.

5. In Template Display Name, enter a certificate. Note the actual Template Name because you will need it later in the configuration.


7. Click the Request Handling tab and then specify Signature and Encryption.

8. Enable or disable Allow private key to be exported.

9. Select Enroll subject without requiring any user input.

10. Select Supply in the request.

11. Click OK on the warning window.

12. Click the Security tab.

13. Grant Enroll permissions to a user account that will be making the certificate requests from Device Manager.

14. Open MMC and add a Snap-In for Certification Authority. Expand the CA server and right-click Certificate Templates.

15. Make sure that User template as shown in the screenshot below exists within Certificate Templates. Make sure that User template exists, otherwise the server will be unable to issue a user certificate.

16. Click New and then click Certificate Template to Issue. Select the certificate template you created in the preceding steps.
To generate the XenMobile client certificate

You can request certificate from any system in the domain; however, make sure to logon using domain service account credentials. The domain account must have local administrator rights to the system requesting a certificate from the Certificate Server.

1. Either Run As a Domain User or initiate a Remote Desktop session to a system using Domain User credentials.

2. Open a web browser and open the web enrollment page for Microsoft Certificate Services. This page is usually https://server.company.com/certsrv (certsrv is case-sensitive).

3. Click Request a Certificate.

4. Click User Certificate and the click Submit.

5. Click Install the Certificate.
To export the client certificate

The client certificate that you request must be exported as a .p12 or PKCS12 certificate and copied to the Device Manager server.

1. Export the certificate as a .p12 or PKCS12 certificate from the web browser used or from the Certificates console on the CA server.

2. Open an MMC Console and add the Certificates Snap-in.

3. Right-click the certificate that you requested and then click All Tasks and Export.

4. In the Certificate Export window, click Next.

5. Click Yes to export the private key.

6. Enter a password for the exported certificate. You will need to remember this password.

7. Enter a file name for the certificate export and then click Next.

   **Note:** The file name cannot contain spaces.

8. Click Finish.

9. Copy the filename.pfx or filename.p12 to the Device Manager server and specify a location.
To configure a Microsoft certificate server entity

1. In the Device Manager web console, click Options.

2. In the Options dialog box, from the left side select PKI > Entities.

3. Click New > New MsCertSrv entity.

4. In the Add a MsCertSrv entity dialog box, on the General tab enter the following information:
   a. Entity name. Type a name for your new entity, which you’ll use later on to refer to that entity. Entity names must be unique.
   b. Service root URL. The base URL of your Microsoft CA’s web enrollment service; for example, https://192.168.2.113/certsrv/ (the URL may use plain HTTP or HTTP-over-SSL).
   c. certnew.cer page name. The name of the certnew.cer page, if you have renamed it for some reason. If not, then you can leave this field empty.
   d. certfnsh.asp page name. The name of the certfnsh.asp page, if you have renamed it for some reason. If not, leave this field empty.
   e. Authentication type. Select No authentication, HTTP-Basic Authentication or SSL client certificate authentication. For the latter, you will have to upload the SSL client certificate to the repository (with its private key) and select it here.

5. Next, select the Templates tab. On this tab, you will need to list the Certificate templates for your Microsoft CA. Note that those must be the internal names, not the display names.

6. Next, select the Custom HTTP parameters tab. On this tab, you can specify custom parameters that XenMobile should inject in the HTTP request to the Microsoft Web Enrollment interface. This will only be useful if you have customized scripts running on the CA.

7. Next, select the CA Certificates tab. On this tab, you will be required to inform XenMobile of the signers of the certificates the system will obtain through this entity. When your CA certificate is renewed, all you need to do is update it in the repository and then the change will be effected to the entity transparently.

8. Click Create.
To configure a Microsoft certificate services policy

Before you can configure a Microsoft certificate services policy, you need to configure a Microsoft CA credential provider in the Device Manager Options dialog box. Once the Microsoft CA credential provider has been configured, then you can create the policy that references the provider. For instructions, see To create a credential provider using external PKI entities.

1. Click the Policies tab in the Device Manager console.

2. On the left-hand pane, under iOS, click Configuration profiles.

3. Click New Configuration > Profiles and Settings > Credentials.

4. In the Credential configuration creation dialog box, on the General tab, enter the following information:
   a. Identifier. Type a name for the profile that identifies it uniquely to the user. This name must be unique and not in use by any other profile, or if this name matches the name of another policy, the first policy will be overwritten.
   b. Display name. Type a name of the profile as it will appear in the Device Manager web console.
   c. Organization. Type your company or organization name.
   d. Description. Type an optional description to describe the policy.
   e. In the Allow Profile Removal section, choose one of the following:
      · Always. Allows the profile to always be removable.
      · Authentication. Allows you to enter a required password that is used when profile is removed. Requires a password.
      · Never. Prevents the profile from ever being removed.
   f. Select the Automatic Removal Date check box if you want to select a specific date on which to remove the profile.
   g. Select the Duration until removal (in days) check box to specify a set a period of time after which the profile will automatically be removed.

5. Next, select the Credential tab, and configure the following settings:
   a. Credential Type. Select Credential Provider.
   b. Credential Provider. Select the Microsoft CA credential provider you previously configured in the Device Manager Options dialog box.

6. Click Create.
To configure a Microsoft certificate services policy

This policy can now be deployed to iOS devices. For information, see Creating Deployment Packages
Configuring an OpenTrust PKI Adapter for Device Manager

XenMobile OpenTrust Adapter was validated with OpenTrust PKI Version 4.7.1 (r131349).

The XenMobile OpenTrust Adapter is a web application running on Tomcat:

- Windows 2008 R2
- Java 1.6.0_29 or above, 32 bits version
- Apache Tomcat 7.0.27

Note: You only need Tomcat core features, not the manager or the documentation, unless you need it. After installation, you can also delete the directory <tomcat_dir>/webapps/ROOT.

The XenMobile OpenTrust Adapter provides an interface that allows Device Manager to submit certificate requests for a signature to an OpenTrust Certificate Manager server. Device Manager submits a request to the OpenTrust adapter to sign a certificate. The OpenTrust Certificate Manager receives the request, signs the certificate and returns it to Device Manager. Device Manager makes these certificate requests in order to generate device identity for mobile device management mutual authentication, or user credential certificates to be used in conjunction with WiFi, VPN, and Exchange ActiveSync profiles for iOS devices. XenMobile recommends that the OpenTrust Adapter is installed on a separate server from the Device Manager host, using its own instance of Tomcat 7.0.

To install OpenTrust Adapter

1. Copy the provided WAR file to the Tomcat webapps directory. You can change the WAR file name to fit the usage of this adapter instance (wifi_certificate, exchange_certificate, and so on).

2. Start Tomcat. It will automatically expand and install the web application in its directory.

3. To check that the adapter is properly running, connect to:
To obtain an authentication certificate from OpenTrust PKI

The authentication between the OpenTrust Adapter is secured by using a client certificate that needs to be generated from OpenTrust PKI server.

1. Log in to the OpenTrust PKI server, browse to Enrollment Entity and then click Request a Certificate.

2. Select Other and then click Next.

3. Select Authentication and then click Next.

4. Enter the required parameters and then click Next.

5. You now need to validate the certificate request. Navigate to Registration Authority > Enrollment > List Certificate Requests.

6. Select your certificate request and then click Process selected requests.

7. Click Approve.

8. You now need to retrieve the certificate. Navigate to Enrollment Entity > Search for a Certificate > Enrollment.

9. Enter your search criteria and then click Search.

10. Find your certificate and then click the name.

11. Click Integrate this certificate into your browser (or smartcard).

12. Open the certificate store of your web browser. For example, with Firefox, navigate to Options, click the Encryption tab and then click View Certificates.

13. In the Certificate Manager, click the Your Certificates tab.

14. Select your certificate and then click Backup.

15. Enter the password and save the resulting p12 file. You will need the file and password when you configure the adapter.
To set up access rights on OpenTrust PKI

You need to provide the required access rights to the generated identity.

1. Navigate to Access Control.
2. Select your User.
3. If you already have a group defined to allow SOAP access to the Registration Authority, you can add this user to this group. Select the group and then click Save.
4. To give individual rights to that user, click the Rights on Modules tab.
5. Select the Execute check box to give access rights to the Registration Authority.
6. Click the Rights on Zones & Profiles tab.
7. For each profile you want the user to be able to control, next to Enrollment, select the Execute check box.
8. Click Save.
To configure the OpenTrust adapter

1. Open the file opentrust_adapter.properties in `tomcat/webapps/<adapter_name>/WEB_INF/classes` and edit it accordingly:

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>OpenTrust.RA.Url</code></td>
<td>Web address used to access the SOAP interface of the OpenTrust PKI server</td>
</tr>
<tr>
<td><code>Enrollment.Profile</code></td>
<td>OpenTrust Profile name used by this instance</td>
</tr>
<tr>
<td><code>KeyPair.FileName</code></td>
<td>Path to the keypair used to authenticate to OpenTrust PKI SOAP interface</td>
</tr>
<tr>
<td><code>KeyPair.Psw</code></td>
<td>Password of the above mentioned keypair</td>
</tr>
</tbody>
</table>
To set the connection to the adapter

1. To configure Device Manager with your adapter, on the Options menu, click PKI Entity.

2. Click New and then enter the required information:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Entity Name</strong></td>
<td>Name your adapter connection.</td>
</tr>
<tr>
<td><strong>URL</strong></td>
<td>Enter the URL of the adapter web services interface:</td>
</tr>
<tr>
<td></td>
<td>http://&lt;server&gt;:&lt;port&gt;/&lt;adapter_name&gt;/GpkiAdapter?wsdl</td>
</tr>
<tr>
<td><strong>Certificate path</strong></td>
<td>If you are using an authenticated HTTPS connection, select your client cert (p12).</td>
</tr>
<tr>
<td><strong>Certificate password</strong></td>
<td>Enter the password for the above p12.</td>
</tr>
</tbody>
</table>

3. Click Load to initiate the connection with the adapter.

4. Click Ping to check the connectivity.

5. Click Create to save the adapter configuration.
To configure an iOS profile to deliver certificates to iOS devices

To deliver certificates to iOS devices, you need to configure an iOS profile in Device Manager. For more information on configuring PKI integration with Device Manager, see About XenMobile PKI.

1. Click the Policies tab.
2. On the left side, under iOS, click Configurations.
3. Create a new policy for the PKI authority that you installed by clicking New Configuration > Profiles and Settings > Credentials.
4. On the General tab, enter the following information:
   a. Identifier. Enter a unique identifier to distinguish the certificate policy.
   b. Display name. Enter a name that will be used to label the policy on the device.
   c. Organization. Enter your company name here.
   d. Descriptions. Type an optional description.
5. In Allow profile removal operation, click one of the following options:
   - Always: This option allows the profile to always be removable.
   - Authentication: Allows you to enter a required password that is used when profile is removed. Requires a password
   - Never: Prevents the profile from ever being removed.
6. Select the Allows you to select a specific date check box to specify a date you want to remove the profile.
7. Select the Duration until removal (in days) check box to enable you to set a period of time after which the profile will automatically be removed.
8. Next, on the Credential tab, enter the following information:
   a. Credential name. Provide a unique name for the credential.
   b. Description. Optionally, you can type a description for the credential.
   c. Credential Type. Select a credential type according to the PKI configuration you have set up for Device Manager, such as a certificate, a keystore, a server certificate, or a credential provider.
   d. Credential file path, Server certificate, or Credential provider. Select the path or the name of the credential you are adding to the policy. If you are using a Keystore file, then you need to provide the keystore password.
To configure an iOS profile to deliver certificates to iOS devices

9. Click Create.
To configure an OpenTrust adapter to use HTTP by using a self-signed certificate

If you want the adapter to be accessible using HTTPS, you need to configure the Tomcat connector accordingly. You can configure the adapter by using a self-signed certificate. This process uses openssl and java keytool.

1. Create a directory called certs. In that directory, create another directory called ca.

2. Create a root CA. You need to adapt the subject name and passwords to fit your needs.
   In the certs directory, issue the following commands:
   ```
   openssl genrsa -aes256 -passout pass:zenprise -out ca/ca.key 1024
   openssl req -new -x509 -passin pass:zenprise -key ca/ca.key -out ca/ca.pem -days 3650 -subj "/C=US/ST=CA/L=RWC/O=Zenprise/OU=Zenprise/CN=ZenTestCA/emailAddress=none@zenprise.com"
   openssl x509 -inform PEM -in ca/ca.pem -outform DER -out ca.crt
   ```

3. Create an HTTPS certificate using that CA. Change at least the CN to fit the XenMobile OpenTrust Adapter server name. For example:
   ```
   openssl genrsa -aes256 -passout pass:zenprise -out server-key.pem 1024
   openssl req -new -passin pass:zenprise -subj "/C=US/ST=CA/L=RWC/O=Zenprise/OU=Zenprise/CN="MyServerName.zenprise.com"/emailAddress=none@zenprise.com" -days 3650 -key server-key.pem > server.csr
   openssl x509 -req -passin pass:zenprise -in server.csr -out server-crt.pem -CA ca/ca.pem -CAkey ca/ca.key -CAcreateserial -CAserial ca.srl
   ```

4. Create a p12 containing your key and certificate.
   ```
   openssl pkcs12 -export -in server-crt.pem -inkey server-key.pem -out MyServerName.p12 -name server
   ```

5. Create a java keystore containing that PKCS12 file.
   ```
   keytool -importkeystore -deststorepass changeit -destkeypass changeit  -destkeystore keystore.jks -srckeystore MyServerName.p12  -srcstoretype PKCS12 -alias server
   ```

6. Modify the Tomcat server.xml file to create the HTTPS connector. The file needs to reference the keystore previously created.
   ```
   <Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
   maxThreads="150" scheme="https" secure="true"
   clientAuth="false" sslProtocol="TLS"
   keystoreFile="C:\Zenprise\Apache Software Foundation\Tomcat 7.0\conf\keystore.jks" keystorePass="changeit"/>
   ```

7. Import the root cert in the java keystore of DeviceManager so that this server certificate can be trusted. On the Device Manager server, issue the following command:
   ```
   keytool -import -trustcacerts -alias root -file ca.crt -keystore cacerts
   ```
To configure an OpenTrust adapter to use HTTP by using a self-signed certificate

The keystore file used by Java (cacerts) is usually located in: C:\Program Files\Java\jdk1.6.0_22\jre\lib\security
To configure Device Manager to generate identity certificates from OpenTrust adapter

You will need to generate a certificate from OpenTrust with the following keyUsage:

- keyEncipherment
- digitalSignature

Furthermore, you will need an OpenTrust root certificate and a CA certificate.

**Caution:** This procedure will invalidate all certificates used previously by Device Manager. All devices using a certificate to authenticate, such as iOS and Android, Symbian, and Windows Mobile using Strong Authentication mode will need to be re-enrolled.

1. Modify pki.xml. This file is located in tomcat/webapps/zdm/WEB-INF/classes. Open it with a text editor, and modify it as follows (the modified parts are in bold text). Keep in mind the following considerations:

- Path to the certificates.
- keyUsage of the certs.
- Name of the OpenTrust connector in the console.
- The CSR template that has to match your profile definition on the OpenTrust PKI Server.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:p="http://www.springframework.org/schema/p"
  >

  <bean id="legacyRoot" class="com.sparus.nps.pki.def.PublicCertFileParams"/>
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<bean id="legacyIosDevicesCa" class="com.sparus.nps.pki.def.KeyStoreParams"
    p:keyStoreType="${ios.mdm.pki.ca-mdm.keystoretype}"
    p:keyStorePath="${ios.mdm.pki.ca-mdm.certificatefile}"
    p:entryAlias=""
    p:keyStorePass="${ios.mdm.pki.ca-mdm.privatekey.password}"
    p:publiclyTrusted="false"
    p:issuerParams-ref="legacyRoot"
/>  

<!-- SHTP is the proprietary protocol ZDM uses to communicate with Windows and Android devices -->

<bean id="legacyShtpDevicesCa" class="com.sparus.nps.pki.def.KeyStoreParams"
    p:keyStoreType="${secure.device.keystore.type}"
    p:keyStorePath="${secure.device.certificate.file}"
    p:entryAlias="${secure.device.alias}"
    p:keyStorePass="${secure.device.private.key.password}"
    p:publiclyTrusted="false"
    p:issuerParams-ref="legacyRoot"
/>  

<alias alias="legacyDigitalSigner" name="legacyIosDevicesCa" />

<bean id="legacySslCert" class="com.sparus.nps.pki.def.KeyStoreParams"
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<bean id="OT_Root_cert" class="com.sparus.nps.pki.def.PublicCertFileParams"
  p:certificateFilePath="C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\otroot.cer"
  p:publiclyTrusted="false"
/>

<bean id="OT_CA_cert" class="com.sparus.nps.pki.def.PublicCertFileParams"
  p:certificateFilePath="C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\otinter.cer"
  p:publiclyTrusted="false"
  p:issuerParams-ref="OT_Root_cert"
/>

<bean id="OT_RA_cert" class="com.sparus.nps.pki.def.KeyStoreParams"
  p:keyStoreType="PKCS12"
  p:keyStorePath="C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\otadmin.p12"
  p:entryAlias=""
  p:keyStorePass="opentrust"
  p:issuerParams-ref="OT_CA_cert"
/>
```

```xml
<bean class="com.sparus.nps.pki.spi.impl.GpkiCa" id="OT_CA">
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

    <property name="caCertificate">
        <description>
            This CA's certificate.
        </description>
        WARNING! In order for tomcat to accept clients presenting identities
        issued by this CA, tomcat's truststore has to be modified accordingly
        (e.g. installing in it the certificate referred to here).
    </description>
    <bean factory-bean="certFactory" factory-method="buildPublic">
        <constructor-arg ref="OT_CA_cert" />
    </bean>
    </property>

    <property name="entityName" value="OTAdapter">
        <description>
            This is the GPKI entity name as defined in the console.
        </description>
    </property>

    <property name="requestProperties">
        <description>
            If the adapter defines user parameters (i.e., non-injected parameters),
            then they can be defined here. EMC adapter currently does not define
            any parameters.
        </description>
        <bean class="com.sparus.nps.pki.gpki.util.SimpleRequestProperties">
            <constructor-arg index="0" type="java.util.Map">
                <map key-type="java.lang.String" value-type="java.lang.String">
                    <!--entry key="[PARAMETER NAME]" value="[PARAMETER VALUE]" -->
                </map>
            </constructor-arg>
        </bean>
    </property>
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<constructor-arg/>
</bean>
</property>

<property name="raEncryptionCert">
<description>
RA encryption cert. MUST be issued by the certificate referred to in property caCertificate, i.e. the CA certificate, i.e. the certificate that will sign device identities.

This cert MUST have keyUsage: keyEncipherment.

RA encryption cert may be the same one as RA signing cert.
</description>
<bean factory-bean="certFactory" factory-method="buildPrivate">
<constructor-arg ref="OT_RA_cert" />
</bean>
</property>

<property name="raSigningCert">
<description>
RA signing cert. MUST be issued by the certificate referred to in property caCertificate, i.e. the CA certificate, i.e. the certificate that will sign device identities.

This cert MUST have keyUsage: digitalSignature.

RA signing cert may be the same one as RA encryption cert.
</description>
<bean factory-bean="certFactory" factory-method="buildPrivate">
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

    <constructor-arg ref="OT_RA_cert" />

    </bean>

    </property>

    <property name="csrTemplate">

    <bean class="com.sparus.nps.pki.spi.impl.CsrMacroTemplate">

        <description>

            Template for the CSR.

            WARNING! Macros have to be specified using ‘%{...}’, instead
            of ‘${...}’, in XML files.

        </description>

        <property name="dnFields">

            <list>

                <description>

                    The following are samples. Remove or add others as you like.

                </description>

                <bean class="com.sparus.nps.pki.def.DNFieldBean" p:oid="CN" p:value="%{user.loginname} />

                <bean class="com.sparus.nps.pki.def.DNFieldBean" p:oid="OU" p:value="aeotn" />

                <bean class="com.sparus.nps.pki.def.DNFieldBean" p:oid="O" p:value="noise" />

                <bean class="com.sparus.nps.pki.def.DNFieldBean" p:oid="C" p:value="DE" />

            </list>

        </property>

        <property name="altnames">

            <list>

                <description>

                    The following are samples. Remove or add others as you like.

                </description>

                <bean class="com.sparus.nps.pki.def.AltNameBean" p:sanType="rfc822Name" p:value="%{user.mail} />

            </list>

    </bean>

</property>
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<bean class="com.sparus.nps.pki.def.AltNameBean" p:sanType="userPrincipalName" p:value="%{user.username}@home.net"/>

</list>

</property>

</bean>

</property>

</bean>


<!--

The new PkiSpi infrastructure is designed to support all the PKI
capabilities we can reasonably be expected to need in the average term.
However, the rest (installer / business process) isn't up to par
yet; as such, we're retrofitting this infrastructure to work with
our current setup. That's the meaning behind the word "legacy"
in this context.
-->

<bean id="certFactory" class="com.sparus.nps.pki.def.ZdmCertificateFactory">
  <description>
  The ZdmCertificateFactory builds public key certificate objects
  from either PublicCertFileParams, PrivateCertFileParams or
  KeyStoreParams; and private key certificate objects (public
  key + private) from PrivateCertFileParams and KeyStoreParams.

  Factory method for the former is: buildPublic; for the latter: buildPrivate.
  
  </description>
</bean>
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<bean id="serialNumberGen" class="com.sparus.nps.pki.gen.CertificateSerialNumberSequenceImpl" />

<bean id="com.everywan.security.PkiSpi.internal" class="com.sparus.nps.pki.spi.impl.PluggablePki" lazy-init="true">
  <property name="digitalSignatureRoot">
    <bean factory-bean="certFactory" factory-method="buildPublic">
      <constructor-arg ref="legacyRoot" />
    </bean>
  </property>

  <property name="sslRoot"><null /></property> <!-- We don't have the config for this... -->

  <property name="digitalSigningCertificate">
    <bean factory-bean="certFactory" factory-method="buildPrivate">
      <constructor-arg ref="legacyDigitalSigner" />
    </bean>
  </property>

  <property name="sslCertificate">
    <bean factory-bean="certFactory" factory-method="buildPrivate">
      <constructor-arg ref="legacySslCert" />
    </bean>
  </property>

  <property name="shtpCa" ref="OT_CA" />

  <property name="iosMdmCa" ref="OT_CA" />
</bean>
```
To configure Device Manager to generate identity certificates from OpenTrust adapter

```xml
<bean id="com.everywan.security.PkiSpi.factory" class="com.sparus.nps.pki.def.PkiSpiFacade">
  <property name="enabled" value="${zdm.pki.enable}" />
  <property name="enabledBeanId"><idref local="com.everywan.security.PkiSpi.internal" /></property>
</bean>
</beans>

To add certificates to the Device Manager keystore

You now need to add the intermediate and root ca certificates to the Device Manager keystore.

1. Use the java keytool command (adapt the path to your environment): "C:\Program Files\Java\jdk1.6.0_23\jre\bin\keytool" -importcert -trustcacerts -alias "externalCA" -file "C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\mycert.cer" -keystore "C:\Program Files\Zenprise\Zenprise Device Manager\tomcat\conf\cacerts.pem.jks" -storepass notMeaningFul

2. Restart the Device Manager service to activate the new PKI usage.
To activate logging on Device Manager for the adapter

Logs from the adapter can be found in the tomcat/logs directory of the adapter.

1. Add a new logger in the log4j configuration to ensure proper error handling and auditing. In Internet Explorer, navigate to the following URL based on your installation: http://<host>/<instance>/log.jsp

2. Navigate to the bottom of the table and in Add New Logger, add an entry for the com.sparus.nps.pki

3. Set the logging level to TRACE.
Configuring the XenMobile RSA Adapter

The XenMobile RSA Adapter provides an interface that allows Device Manager to submit certificate requests for a signature to an RSA Certificate Manager server. Device Manager submits a request to sign a certificate to the RSA adapter. The RSA Certificate Manager receives the request and uses the RSA Xuda Libraries to sign the certificate. The Certificate Manager returns the signed certificate to Device Manager.

Device Manager makes the certificate requests in order to generate device identity for mobile device management (MDM) mutual authentication, or to generate user credential certificates to be used in conjunction with WiFi, VPN, and Exchange ActiveSync profiles for both iOS and Android devices.

Prerequisites

Citrix recommends the following prerequisites:

- Install the RSA Adapter on its own server, separate from the server running Device Manager and that you use a 32-bit instance of Tomcat 6.0.
- Device Manager Versions 7.0, 7.1, or 8.0.1.
- JAVA SDK 1.6 or later.
XenMobile RSA Adapter Certificate Manager Requirements

To install the XenMobile RSA Adapter, the following RSA Certificate Manager configurations are required. For the proper settings, consult your RSA Certificate Manager Installation Guide.

RSA Certificate Manager Installable Elements

- RSA CA Manager version 6.8 build 519 or later
- RSA Certificate Authority Version 6.8 Build 519 or later
- No special OSI-level privileges

RSA Certificate Manager Configurable Elements

- Configuration of CRL publishing: N/A
- Configuration of OCSP responder: N/A
- Configuration of certificate publishing: N/A

Partner Product Installable Elements

- Tomcat 6.0 or later, 32 bit
- Java SDK 1.6 or later

Partner Product Configurable Elements

- CRL checking mechanism: N/A
- OCSP checking mechanism: N/A
- Trust validation: N/A
- Enrollment: N/A
- General modifications to the partner product: N/A
Installing and Configuring the XenMobile RSA Adapter

The XenMobile RSA Adapter provides a mechanism for Device Manager to sign and revoke certification against an RSA Certificate Authority Version 6.8. The RSA Adapter enables device identity for mobile device management (MDM) mutual authentication and user credential certificates for use in conjunction with WiFi, VPN, and Exchange ActiveSync profiles. You perform the following tasks to install the RSA Adapter:

1. Set the Java SDK path on the Windows-based computer where you will install the RSA Adapter.
2. Configure the correct port (80) on your Tomcat server.
3. Copy the RSA Adapter installation and configuration files into a target installation directory.
4. Edit the RSA Adapter properties file with values obtained from the RSA Certificate Authority Manager Console.
5. Copy the RSA Certificate Authority Manager .cert and .key files to the installation computer.
6. Execute the RSA Adapter installation executable to install the software.
7. Verify the installation in a browser.

You perform the following tasks to configure the RSA Adapter:

1. Create and configure a PKI entity profile in Device Manager to be able to connect to the RSA Web Services Description Language (WSDL).
2. Create an iOS profile to enable use of the Certificate Authority.
3. Add a new logger in the log4j configuration to ensure proper error handling and auditing.
4. Configure the new PKI profile so it can be deployed to an iOS device and validated.
To install the RSA Adapter on Windows Server

Make sure you have access to the zenadapter.war file that is included as part of the RSA Adapter product distribution.

1. On the Windows server where you are installing the RSA Adapter, set the path to include the JAVA SDK 1.6+. For example, \Program Files (x86)\Java\jdk1.6.0_29\bin.

2. Next, configure the Tomcat server to run on port 80, instead of the default port of 8080:
   a. Navigate to %TOMCAT_HOME%/conf directory.
   b. Edit the server.xml file as follows:
      · Change non-SSL to:
        <!-- Define a non-SSL HTTP/1.1 Connector on port 8080 --
        <Connector port="80" ...
      · Change SSL to:
        <!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
        <Connector port="443" ...

3. On the installation computer, create a new directory named C:\Xenmobile.

4. Unzip and copy the entire contents of the RSA Adapter zip package to the directory named C:\Xenmobile.

5. Create a passphrase file that stores a passphrase that will be used by the RSA Adapter.
   *ote: Before you execute this command, make sure you are logged in as the Service Account user. The Service Account user you log in as must be the same Service Account that the Tomcat server runs as.

6. Open the Windows command prompt and change directories to the location of the C:\Xenmobile folder. For example, cd C:\Zenprise.

7. From this directory, execute the following command: java -jar WinDPHarness <passphrase filename> <passphrase>
   
   Note: Note the file path name used in this command <passphrase filename>, because you will need it when you edit the prop.txt file in the following step.

8. Open the C:\Xenmobile\prop.txt file in a text editor and set the following attributes in the file, for example:
   · ldapport=636
To install the RSA Adapter on Windows Server

- ldaphost= rsa1.kqe.xenmobile.com
- camd5= a2064dd584c7025f03ceb0443ca0fe9e
- keyfile=C:\xenmobile\admin.key
- certfile=C:\xenmobile\admin.cert
- protectFlag=0
- jurilID=fe109c4d64430fa6f6d614c08b75312b0b7e31226
- passphrasefile=c:\xenmobile\passcode.txt
- profileflag=1
- profileID=AC1E02D427C3D8
- keepldapopen=1

**Note:** These properties are available in your RSA Certificate Authority Manager console. Refer to your RSA Certificate Authority Manager guide for instructions on where to access these properties.

9. From the RSA Certificate Authority Manager server, copy the two RSA CA Manager files - .cert and .key - to the C:\xenmobile folder on the computer where you are installing the Zenprise RSA Adapter.

10. Copy the zenadapter.war file to the %TOMCAT_HOME%\webapps folder.

11. From your command prompt, execute the following commands:

    cd %TOMCAT_HOME%\webapps
    run jar xvf zenadapter.war

    Stop Tomcat
    Start Tomcat

12. Verify that the installation was successful. In Internet Explorer (8 or later), navigate to http://HOST WITH_ADAPTER_TOMCATINSTANCE/zenadapter.war. A page with the adapter WDSL and link should appear.
To configure the RSA Adapter in the Device Manager web console

To configure the RSA adapter in the Device Manager web console, you first configure a new PKI entity. Next, you create a new iOS profile to enable use of the Certificate Authority.

To configure a new PKI entity

1. Log on to the Device Manager web console and then click Options.
2. In the XenMobile Server Options dialog box, under PKI, click Entities.
3. In the PKI entities configuration screen, click New and then click New generic PKI entity.
4. Enter a name and then enter a URL for the WSDL that you installed when you finished the RSA Adapter installation. For example: http://zdm.zenprise.com/gpki/sample.
5. If the adapter is available over HTTPS/SSL, upload the SSL client certificate. If you are not using SSL, skip to the next step.
6. Click Load.
7. Test the connection to the adapter. Click the Capabilities tab and then click Ping. A "Ping Successful" message should appear.
To create a new iOS profile

1. Click the Policies tab, click to expand iOS and then click Configurations.

2. Create a new policy for the PKI authority that you installed by clicking New Profile.

3. On the same server running Device Manager, add a new logger in the log4j configuration to ensure proper error handling and auditing. In Internet Explorer, navigate to the following Web address based on your installation: http://<host>/<instance>/log.jsp

4. Navigate to the bottom of the table and add a new logger entry for the com.sparus.nps.pki.

5. Set the logging level to TRACE.

6. Test the deployment profile on a new iOS device by moving the new PKI package into the Resources to Deploy section so you can deploy the package to an iOS device.

7. Register a new device that is targeted with the package and verify that you see the new certificate on the iOS device. If the package does not deploy, check the log file and then contact IT support.
Configuring an External Certificate Authority by Using SSL

Before you configure an external Certificate Authority (CA) by using SSL, the following files should be on the Device Manager server and accessible by the Device Manager server:

- An external SSL certificate file in .p12 format issued by a trusted CA. The file name, externalSsl.p12, is used as an example in this procedure.
- A password for the .p12 certificate file should be known by the installing party.

You need to configure two XML files: The pki.xml file located in the \..\tomcat\webapps\zdm\WEB-INF\classes directory and the server.xml file located in the \..\tomcat\conf directory.

1. Locate the pki.xml file in \..\tomcat\webapps\zdm\WEB-INF\classes.

2. To configure the external SSL certificate, add an "externalSslCert" bean to the file as shown in the following example. Modify the bold fields appropriately. The keyStorePath should reference the .p12 certificate file located on the server. The keyStorePass should contain the password for the .p12 file.

   ```xml
   <bean id="externalSslCert" class="com.sparus.nps.pki.def.KeyStoreParams"

   p:keyStoreType="PKCS12"

   p:keyStorePath="C:\ExternalSSL_Cert\qamdm01\externalSsl.p12"

   p:entryAlias=""

   p:keyStorePass="xxxxxxx"

   p:publiclyTrusted="true"

   />
   
3. Set externalSslCert as the sslCertificate property. Replace the highlighted line with the proper bean name you specified in the preceding Step.

4. Locate the server.xml file located in the \..\tomcat\conf directory.

5. Locate the Connector port="443" and modify the following two parameters for this connector to bind the external SSL certificate to this port. The keystoreFile should point to the .p12 certificate file located on the server. The keystorePass parameter should contain the password for the .p12 file.

   ```xml
   <Connector port="443" maxHttpHeaderSize="8192" maxThreads="400" minSpareThreads="5" maxSpareThreads="100"

   scheme="https" secure="true" clientAuth="want" SSLEnabled="true" keystoreFile="C:\ExternalSSL_and_Signing_Cert\ExternalSsl.p12"

   keystorePass="xxxxxxxx"

   truststoreFile="C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\conf\cacerts.pem.jks"
   ```
6. Locate the Connector port="8443" as shown in the following example. The keystoreFile should point to the .p12 certificate file located on the server. The keystorePass parameter should contain the password for the .p12 file.

```xml
<Connector port="8443" maxHttpHeaderSize="8192" maxThreads="20" minSpareThreads="5" maxSpareThreads="5" ... scheme="https" secure="true" clientAuth="false" SSLEnabled="true" keystoreFile="C:\ExternalSSL_and_Signing_Cert\ExternalSsl.p12" keystorePass="xxxxxxxx" truststoreFile="C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\conf\cacerts.pem.jks" truststoreType="JKS" truststorePass="notMeaningFul" keystoreType="PKCS12">
```

7. Restart the Device Manager server.
Configuring a Certificate Services Entity by Using XML

1. Find the sample entity in the file. It will be located in a bean similar to the bold text in the following example.

```xml
<list value-type="com.sparus.nps.pki.conf.MsCertSrvEntity">
  <bean class="com.sparus.nps.pki.conf.MsCertSrvEntityBean">
    <constructor-arg index="0" type="java.lang.String" value="QA-CertSrv-On-SCEP-ClientCertAuth" /> ....
  </bean>
</list>
```

2. Rename the entity to a name for your organization (for example, Company-MS-CA). This name appears in Device Manager.

3. Uncomment the bean for the entity by deleting the comment characters before and after the bean. For example, <!-- including the exclamation and the ending comment -->.

4. Specify the serverBaseUrl value in the file which should be the certificate server URL to make a certificate request (for example, https://cert-server.company.com/certsrv).

```xml
<property name="serverBaseUrl" value="https://serverca.company.com/certsrv/" />
```

5. certFinishPageName/certNewPageName: The default values may be used unless the Microsoft Certificate Server is configured to use non-default pages.

6. Specify the Client Certificate Authentication file, which is the certificate export file that you copied when you exported the client certificate. Modify the following values shown in bold.

```xml
<property name="authentication">
  <bean class="com.sparus.nps.pki.conf.ClientCertAuthentication">
    <property name="keyStoreFile" value="C:\client-certificate-name.pfx" />
    <property name="keyStoreType" value="PKCS12" />
    <property name="keyStorePass" value="xxxxxxxx" />
  </bean>
</property>
```
7. Specify a template name to be used for making user certificate requests from Device Manager. The name should match the certificate template name you created for certificate requests. For details, see To create a certificate template for XenMobile certificate requests. Do not use the template display names. For example, “iPhone Encryption” is the display name whereas “iPhoneEncryption” is the template name. Use the template name without any spaces. The properties dialog box of the template should include both display name and the actual template name.

    <property name="availableTemplates">
    <list value-type="java.lang.String">
        <value>CertificateTemplateName</value>
    </list>
    </property>

8. Save the file and then restart the Device Manager server.
Configuring Location Services for Devices

You can use Device Manager to perform many GPS-related tasks with your devices, such as locating devices, tracking iOS devices, and creating a geo-fence around iOS devices. This section provides steps for setting up location services for Android and iOS devices.
To locate an Android device

You can locate Android devices by using the Locate command.

1. On the Device tab, select the Android device you want to locate.


3. A warning appears prompting you to confirm that you want to locate the device. Click Yes.

4. In the Device dialog box on the General tab, under Location, a statement shows that a location request has been sent to the device in order to locate it on a map. There will be a slight delay between the time the location is sent and the time the response is received.

5. Click the link to see the device location on a map.
To locate an iOS device by using GPS

For devices that have their GPS location enabled, you can locate the device by its geographic coordinates. You send a request from Device Manager to locate a device. As soon as the device is located, you can then view the location on Google Maps.

**Note:** You can only locate a device with Device Manager only if the device has the location services option set to On for the XenMobile client app.

1. On the Device tab, click a device you want to locate.
3. In the Edit an iOS device dialog box, on the General tab, under Location, a message states that a location request has been sent to the device in order to locate it on a map. There will be a slight delay between the time the location is sent and the time a response is received.
4. When the device is located, under Location, a link to Google Maps appears where you can locate the device.
5. Click the link to see the device on the map. The map results show the device location as well as the device serial number.
To locate an iOS device by configuring geotracking

If your iOS devices have a location services policy applied and the device is configured for geo-tracking, the Policies tab shows the locations of the device over the time that you configured in the location services policy. Geo-tracking enables you to track an iOS device over periods of up to a maximum of six hours at a time. You can view the geographical location of a device and its movement and display the device’s location on Google Maps. In addition to activating GPS tracking, you can specify individual parameters for GPS tracking by deploying a geo-tracking policy. If you are satisfied with the default values, however, you can enable tracking immediately.

To configure iOS geo-tracking

1. Click the Policies tab, click iOS and then click Configurations.

2. In New Configuration, click Profiles and Settings, and then click Location Services.

3. In the Locations Services - Configuration creation dialog box, enter the following information:
   - Name. Give the location services policy a name.
   - Description. Provide an optional description for the policy.
   - Location fix timeout. Enter the time Device Manager will wait before timing out if a device location cannot be fixed. If nothing is set, Device Manager attempts to locate the device according to the Poll interval value.
   - Tracking duration. The period of time that the device will be tracked after an Enable Tracking command has been sent to the device. Maximum is six hours.
   - Poll interval. Enter how often Device Manager attempts to fix a location on the device. If the device cannot be located, the attempt to locate will time out according to the Location fix timeout setting.
   - Accuracy. Enter the accuracy of the location point from the device.
   - Report if location services are disabled. The device reports to Device Manager if GPS is disabled and Device Manager shows the status of the device.

   **Note:** You must select this setting if you are using Automated Actions to trigger an action based upon a location-based trigger, such as Location Perimeter Breach or Location Services Disabled. Also, the device must be contacted for its location to trigger the action.

4. Click Create. To enable tracking of the device according to this configuration, you need to deploy the package to the devices you want to track.
To deploy a geo-tracking policy package

1. Click the Deployment tab, click New Package and then click Create new iOS Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the package and then click Next.

3. In the Groups of users window, select the group to which to apply the geo-fencing policy and then click Next.

4. In the Resources to be deployed window, scroll to the Configurations folder, select the geo-fence automated action and then click the right arrow to add the resource to the deployment package.

5. Click Next.

6. In the Deployment schedule window, select the If not deployed Start Now option and then click Next.


8. On the Package summary page, click Finish to deploy the package to the iOS devices of users in the selected group.

When the deployment has complete, select the deployment package and then click the Details button to see information about the success of the package deployment.

To track and view an iOS device

1. On the Devices tab, do one of the following:
   
   - Right-click to select a device, click Security and then click Enable Tracking.
   
   - Click the device and then under Security, click Enable Tracking.

2. To view the device, on the Devices tab, select the device and then click Edit.

3. Select the Geo-tracking tab. Each point on the map indicates when Device Manager fixes the location of the device. A green point indicates the first location point when tracking started, and the red point indicates the last device location point captured before tracking ended.

4. You can mouse over each point to see more detailed geographical information, such as latitude and longitude.

5. To see a longer range of tracking points if, for example, the device was tracked several times, change the Display Points From date range and then click Filter.
Working with Apps

You can add apps and files to Device Manager that you want to deploy to Android, and Windows devices. You can add proprietary apps apps you have developed innterally for your users and then depeloy those apps to the Worx Store in a deployment package. You can also add app defintions of publically available apps, so your users can access them from the iTunes, Google Play, or Windows Phone app stores and install them on their devices.

Apps you deploy appear to the iOS and Android device users in the Worx Home Store. Windows Phone 8 users access their apps from the Worx Home app.
To add a file to Device Manager

To deploy a file to a device, you first need to upload the file into the Device Manager repository database. When you add the files to deployment packages, you can deploy the files to users’ devices. You can add file types, such as documents, images, videos, presentations, and .pdfs.

**Note:** You cannot add files to iOS devices.

1. In the Device Manager web console, click the Files tab and then click New file.

2. In the Import a file to the XenMobile MDM database dialog box, browse to the file on your computer, click Open and then click Import

You can now deploy the file to a device in a deployment package.
Uploading Other Files

If you select a file type that has no ".cab" or ".mscr" extension, you will be presented with several options.

- **Macro must be replaced.** Select this option if you want to search and replace the macros inside the scripts.
- **Destination folder.** Write down the folder where the file should be uploaded.
- **Specify what to do if the file already exists.** Copy it if the files are different or do not overwrite the existing one.
- **Specify if you want to set the Read-Only option.**
- **You can also decide to register a comment if needed.**
Configuring Macro Substitution for Uploading Files

If you want to upload files and Mortscript script files, select the Macro must be replaced box. When you select this check box, it indicates that the script file must be checked before deployment to substitute macros with dynamic values that depend on the user and device.

An example of a macro in a Mortscript script is as follows:

```mortscript
result = Question( "Hello world, mail = \{ user.mail? || protect( '"', '"') | encode('UTF-8') \}!", "It's a start!", "YesNo")
```

The macro is located between the `%{...}%` tags. The server does not detect in advance the file format or its encoding. Instead, the files are analyzed in binary format. The “%”, “{“,... are checked in ASCII format. Those are compatible with most characters encoding (but not UTF-16).

The macros are made of several parts separated by the pipe character: xxx | yyy | zzz:

- The first element indicates a property: user.xxx for a user property, or device.xxx for a device property. You can use the ? character to indicate to use an empty string if the property is not defined.

- The next elements are filters used to encode, transform and mostly protect the string so that it can enter the file context.

In this example, the elements are:

```mortscript
user.mail? | protect('"', '"') | encode('UTF-8')
```

- `user.mail?`: insert the “mail” property of the user. Use an empty string if it is empty or not defined.

- `protect('"', '"')`: protect the quote character by adding another quote before it. This protection is specific to Mortscript that requires to double that character when one wants to define one. For instance: “This is a quote “” in Mortscript”.

- `encode('UTF-8')`: encode the string in UTF-8.

The last part is important since only a binary element can be inserted by Device Manager:

- `user.mail?`: give a character string.

- `protect('"', '"')`: takes a string in input and outputs a string.

- `encode('UTF-8')`: takes a string in input and outputs a binary array.

**Note:** Use other encodings if necessary, like CP1252 in France if you want to use accentuated characters.
This other available functions are:

- S64Encode: [binary => text] encodes a binary in S64.
- S64Decode: [text => binary] decodes a S64 string to binary.
- B64Encode: [binary => text] encodes a binary in Base64.
- B64Decode: [text => binary] decodes a Base64 string to binary.
- encode(encoding): [text => binary] encodes a string with a specific encoding.
- decode(encoding): [binary => text] decodes a binary array to a string.
- protect(searched_character, protection_character): [text => text] insert the protection_character before the searched_character.
- transform(searched_string, replacement_string): [text => text] replace searched_string by replacement_string.

You can use the following user properties in the scripts:

- cn
- company
- companyname
- property_country
- department
- description
- displayname
- distinguishedname
- facsimiletelephonenumber
- givenname
- homecity
- homecountry
- homefax
- homephone
- homestate
- homestreetaddress
- homezip
When users connect to an LDAP directory, most these properties are completed automatically. You can also manually enter the properties in the user properties dialog.

You can use the following device properties:

- ew_version
- ew_revision
- cpu_clock_speed
- sim_id
- memory
- freedisk
- tel_number
- system_oem
- system_platform
Configuring Macro Substitution for Uploading Files

- cpu_type
- system_os_version
- system_os_build
- memory_available
- total_disk_space
- system_language
- user_language
- screen_width
- screen_height
- screen_nb_colors
- main_battery_percent
- backup_battery_percent
- battery_charging
- external_storage1_name
- external_storage1_total_space
- external_storage1_free_space
- external_storage2_name
- external_storage2_total_space
- external_storage2_free_space
- user_defined_1
- user_defined_2
- user_defined_3

You can also use any custom property defined for the device or user.
Examples of Simple Mortscripts

Ask for Reboot Script Example

Result=Question("Your device needs to be rebooted. Do you want to reboot now?", "Hello %user.name? | protect("", "") | encode('UTF-8') }%!", "YesNo")

If ( Result=YES)
Reset
EndIf

This script opens a dialog box with the Yes and No buttons. It asks for the user to reboot now or later. If Yes is pressed, the device will reboot. If No is pressed, nothing happens except that the dialog is closed.

The title of this dialog displays the name of the user, as stored in a custom property of the user.

Data Upload Script

Here is an example of a Mortscript script used to upload a file to a FTP server:


This simple script will upload the file test.zip, located in the directory “My Documents” of the device, to the server ftp.mydomain.com, in the directory “incoming”. The file will be renamed file.zip. It will use the login “test”, and password “test”, and use the passive mode of the ftp protocol.

The synopsis of the FtpUpload function is the following:


Note: In this Device Manager release, the FtpUpload function is not yet part of the standard Mortscript program but is only available in the Device Manager release of MortScript.
To upload a MortScript file

You can add MortScript (.mscr) files to Device Manager to deploy to Windows Mobile devices. MortScript is a batch scripting language that allows you to perform basic functions, such as opening or closing apps, running processes, creating directories, establishing or closing network connections, and other basic device functions,

1. Click the Files tab and then click New file.

2. In the Import a file to the XenMobile MDM database dialog box, browse to the MortScript file on your computer and then click **Open**.

3. Enter the following script parameters:
   a. **Execute script.** Select this option to execute the script automatically when the file transfer is done.
   b. **Macro must be replaced.** Select this option if you want to search and replace the macros inside the scripts.
   c. **Specify what to do if the file already exists.** If the files are different, you can choose make a copy or not overwrite the existing one.

   **Note:** You must encode MortScript files by using ANSI character set if possible. Unicode is also supported with proper prefixes.

4. Click **Import**.
Adding iOS Apps

You can add iOS apps to Device Manager and make them available to your users. You can deploy apps to devices using a deployment package. You can make iOS apps available either through the Connect app in the Applications folder, or create a web clip application store to deploy to your iOS users’ home screen.

You can add iOS apps in two ways:

- Internally. Upload the application to the Device Manager database as an iOS .ipk file
- Externally. Create an application definition that references the App data through a URL to the Apple iTunes app store.
To add an internal iOS app

If you have internally developed iOS apps (.ipk) or iOS apps that you have licensed to distribute, you can upload those apps directly to the Device Manager database and then deploy those apps to users’ devices.

1. In the Device Manager web console, select the Applications tab.

2. Click New > New app.

3. In the Import an application into the XenMobile MDM database dialog box, click Choose File.

4. In the iOS app parameters section, enter the following information:
   
   a. Select the Remove App when MDM profile is removed (Application push only) if you want the app to be removed from any devices you deploy it to if the XenMobile MDM profile is removed from the iOS device.
   
   b. Select the Prevent backup App data (Application push only) option if you want to prevent the device user from backing up the app to an external device or application.

5. Click Import.
To add an external iOS app

For those iOS apps that are must have, or that you would like to recommend to your iOS users, you can define an iOS app definition and then push the app to your users’ devices. When your users open the Connect app on their device and tap the Applications folder, they can down the app to their devices.

In order to add an external iOS app to Device Manager, you will need the complete URL to the app from the iTunes app store.

1. In the Device Manager web console, select the Applications tab.
2. From the New menu select New > External iOS App.
3. In the Add an external iOS application dialog box, enter the following information:
   a. Specify the URL with a link to the Apple App Store.
   b. Click Go to validate the URL link and retrieve application information.
   c. Next, you can optionally select one or both of these app security policies (under the app description):
      · Remove App When MDM Profile is Removed. To ensure that certain any external apps (those not developed by your organization) are only installed on a devices that are managed by your IT department, you can choose to remove a pushed app on iOS device if the user removes their MDM profile.
      · Prevent App Data Backup. Before you push an iOS 5 app to an iOS device, you can select the Prevent Backup of App Data setting, which will prevent allowing the user to backup a specified app either on their computer (via iTunes) or through iCloud.
4. Click Add. The external iOS app definition is added to Device Manager. You can add as many apps or app definitions as you want to push to your users’ devices. These apps can be pushed to users’ devices when add them to a deployment package.
Adding Android Apps

You can add Android apps to Device Manager and make them available to your users. You can deploy apps to devices using a deployment package. You can make Android apps available either through the Connect app in the Applications folder.

You can add Android apps in two ways:

- Internally. Upload the application to the Device Manager database as an Android .apk file
- Externally. Create an application definition that references the App data through a URL to the Google Play or Amazon app store.
To add an internal Android app

If you have internally developed Android apps (.apk) or Android apps that you have licensed to distribute, you can upload those apps directly to the Device Manager database and then deploy those apps.

1. In the Device Manager web console, select the Applications tab.
2. Click New > New app.
3. In the Import an application into the XenMobile MDM database dialog box, click Choose File.
4. Select the file and click Open.
5. In the APK parameters section, enter the following information:
   a. Select the Execute APK File option if you want the app to launch immediately after it is installed on the device.
   b. Select After Installation if you want to prevent deleting the installation file from the device when the installation is done.
   c. In the Destination folder section, select the folder where you want to upload the installation file. Only Flash Storage and Device Manager Installation folder are available as a path prefix for Android devices.
   d. In the If the file already exists section, you can specify what to do if the file already exists. You can copy it if the files are different or do not overwrite the existing one.
6. When you are finished, click Import. Once imported, the app can be added to deployment packages and pushed to Android devices.
To add an external Android app

For those Android apps that are must have, or that you would like to recommend to your Android users, you can define an Android app definition and then push the app to your users’ devices. When your users open the Connect app on their device and tap the Applications folder, they can download the app to their devices. In order to add an external Android app to Device Manager, you will need the complete URL to the app from the Google Play or Amazon app store.

1. In the Device Manager web console, select the Applications tab.

2. From the New menu select New > External APK App.

3. In the Add an external Android application dialog box, enter the following information:
   a. In the Application store drop-down, select either Google Play or Amazon and then specify the URL with a link to the app store.
   b. Click Go to validate the URL link and retrieve application information.

4. Click Add. The external Android app definition is added to Device Manager. You can add as many apps or app definitions as you want to push to your users’ devices. These apps can be pushed to users’ devices when add them to a deployment package.
Adding Windows Mobile Apps

If you select a Windows Mobile app (.cab) to be uploaded, several options appear, as described below.

Note: This is possible only with signed applications. The installation will silently fail otherwise.

- **Execute CAB file**: select this option to execute automatically the installation when the file transfer is done.

- **Silent installation**: select this option to silently install the application, without prompting the end user. Frequently, for reasons of code signatures, messages may be generated asking the device user to confirm installation of applications. Likewise, by default under Windows CE, messages ask where applications should be installed. Device Manager allows applications to be installed in silent mode without the device user having to reply to confirmation messages.

- **After installation**: select this option to avoid deleting the installation file from the device when the installation is done.

- **Destination folder**: Write down the folder where the file should be uploaded.

- **Specify what to do if the file already exists**: copy it if the files are different or do not overwrite the existing one.

- **You can also decide to register a comment if needed.**
Creating Deployment Packages

You can remotely deploy a package of settings to a mobile device from the Deployment tab in the web console. You can use the Package building wizard to build out packages by using preconfigured objects. Connected devices receive the package as soon as scheduling rules are met. Reconnecting devices receive the package as they reconnect subject to other rule criteria.

Packages are compilations of previously created resources, prepared into configurations for the various user groups. Packages include the following:

- A package name
- Groups of users
- Resources, which, depending on the operating device, are a combination of the following:
  - A server group
  - App tunnels
  - Registry configurations
  - XML configurations
  - Software inventory
  - Applications
  - Files
  - Deployment schedule
- Deployment rules
How Base Packages Work

Device Manager contains pre-configured base deployment packages that automatically deploy to devices as soon as a user enrolls the device in Device Manager. The base packages are important for enabling basic device management.

The base packages in Device Manager contain the following policy configurations, categorized by device platform:

- **iOS.** Software inventory and MyAppStore (Citrix Worx Home web clip) policies, plus the following Citrix apps:
  - Citrix ShareFile
  - Citrix Receiver
  - Citrix Podio
  - Citrix GoToMeeting
  - Citrix Mobile Connect (for pre-existing customers upgrading to Citrix Worx Home)

- **Android.** Scheduling policies for connections to XenMobile, remote support tunnel, and software inventory policies, the Worx Home web clip, plus the following Citrix apps:
  - Citrix Receiver
  - Citrix GoToMeeting

- **Windows Phone 8.** Passcode policy.

- **Windows 8 Tablet.** Software inventory policies.

- **Symbian.** Passcode policy.

- **Windows Mobile.** Remote Support tunnel, scheduling, passcode, client config policies.

For more information about configuring policies, see Creating Policies. For more information about deployment packages, see Creating Deployment Packages.
To create and deploy a deployment package

1. In the Device Manager management console, click the Deployment tab.


3. On the Package Name page of the Create New Package wizard, enter a name for the app removal policy, and then click Next.

4. On the Groups of users page, select the users from whose devices you want to remove the app and then click Next.

5. On the Resources to be deployed page, in Available Resources, select the app removal policy you want to use for the package, and then click the right arrow button to add the resource to the package.

6. Click Next.

7. On the Deployment schedule page, configure to push the app no or at a specified time in the future.

8. Click Next.

9. On the Deployment rules page, specify any deployment rules you want to associate with the app and its deployment. For more detailed information, see Deployment Rules.

10. Click Next.

11. On the Package summary page, review the app removal package configuration and then click Finish.

12. Click Deploy in the toolbar.

All connected devices receive all configured packages as soon as scheduling rules are met. Reconnecting devices receive the package when they connect subject to other rule criteria.
Configuring Deployment Rules

You can set any number of parameters that will affect the deployment outcome of a package.

For example, your package deployment could be based on a specific operating system version, on a particular hardware platform, or some other combination. In this wizard, you will find both a Simple and Advanced rule editor, with the Advanced view being a free-form editor.

Simple Deployment Rules

Simple deployment rules are comprised of pre-defined tests and resulting actions. Where ever possible, the results are pre-built into the example tests. For example, when basing a package deployment on a hardware platform, all existing known platforms are populated into the resultant test, drastically reducing your rule creation time, and limiting possible errors.

Click on New rule to add a rule to the package.

Note: The rule builder includes further information, specific to each test.

To create a new rule, you select a rule template, select the condition type, and then customize the rule. Customizing the rule includes modifying the description. When you finish configuring settings, you add the rule to the package.

You can add as many rules as you want. The package is deployed when all of the rules match.

Advanced Deployment Rules

If you click on the Advanced tab, the Advanced Rule Editor appears.

In this mode, you can specify what relationship is set between the rules. The operators AND, OR, and NOT are available.
Configuring Deployment Schedules

The Deployment schedule allows you to define when to deploy a package.

You can schedule the deployment for:

- A future time (Do not deploy).
- A single deployment (one time).
- A permanent deployment to make sure that the devices always have the package content (On every connection). This is designed to ensure the devices initially and continue to comply with your application policies.

The available options might change depending on the platform type.

You can configure the schedule to make sure that a package is only deployed one time. For example, if users change deployment settings, the package will not deploy again.

The windows differ by device type; for Windows Mobile devices the window appears as follows:

**Important:** Some devices may not observe the schedule. When selecting a precise date for the deployment, the targeted devices receive the information to try to reconnect during that time frame, even if they do not have a connection scheduling in place during the specified time frame. However, if the device does not connect or connects later than the configured time frame, the device does not receive the package.
Configuring Package Hierarchy and Relationship Rules

Hierarchy rules apply to assignments between packages and sub-packages. The following table shows how user group and package structures work.

<table>
<thead>
<tr>
<th>User Group Structure</th>
<th>Package Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC (Group parent)</td>
<td>XYZ</td>
</tr>
<tr>
<td>• Marketing (Group child)</td>
<td>• Marketing</td>
</tr>
<tr>
<td>• R&amp;D (Group child)</td>
<td>• RD</td>
</tr>
<tr>
<td>• Admin (Group child)</td>
<td>• Admin</td>
</tr>
</tbody>
</table>

Scenario 1: If the assignment was made at the parent package level but not at that of the sub-package, the latter inherits its parent package's assignments. The conditions are:

- The XYZ package is not assigned to a specific group.
- The Marketing, R&D and Admin sub-packages are assigned to the ABC.Marketing, ABC.RD, and ABC.Admin subgroups, respectively.

The result is that the ABC Marketing, ABC RD, and ABC Admin subgroups inherit from the XYZ package solely because this package is not assigned.

Scenario 2: If the assignment was made at the parent package level as well as at that of the sub-package, the latter retrieves its own assignments alone. The conditions are:

- The XYZ package is assigned to the ABC group.
- The Marketing, R&D and Admin sub-packages are assigned to the ABC.Marketing, ABC.RD, and ABC.Admin subgroups, respectively.

The result is that the ABC. Marketing, ABC RD and ABC Admin subgroups do not inherit from the XYZ package.

Note: You can restrict the deployment of a package to a subset of devices within the selected user group by defining rules.
Configuring General Device Manager Options

You can use General Options to set general Device Manager device display settings, device access relative to number of users per device, device triangulation enablement, and the Enterprise App Store availability for iOS.

- Inactivity Days Threshold. Defines a time period in days within which a device must communicate back to the Device Manager server before changing the device status to "inactive".

  **Note:** If you are using Cisco ISE (or other NAC appliance) in conjunction with the Device Manager server to filter device access to your network, and if the Inactivity Days Threshold value changes, restart the Device Manager service on the Device Manager server for the changes to take effect.

- Number of Devices per User. Maximum number of devices a user can enroll. If you want to prevent device sharing, you can restrict the number of users per single mobile device, as well as restrict the number of devices that a single user can register and enroll. If you set the value to zero that means a user can own any number of devices. When a device or user limit is exceeded, the users receive an error that indicates that a connection or license limit is reached, which prevents the additional user or device from enrolling.

- Number of Users per Device. Maximum number of users that can share a single device. The default value is zero, which means an unlimited amount of users can share the device.

- Highlight Jailbroken or Rooted column, SMG Status column, Managed column. When enabled, these options provide status 'lights' to indicate a device status. When disabled, the status lights (red or green) will not display and text will be used to indicate status.

- Enable Device Triangulation. Enables devices to be located geographically for GPS geo-location and geo-fencing policies.

- Send Android Domain Users to Secure Mobile Gateway. When enabled, this option ensures that Device Manager sends Android device information to Secure Mobile Gateway in the event that Device Manager does not have the Android device user's ActiveSync identifier (ID).
Configuring Device Manager Security Options

The security options dialog box allows to customize the security features of the service. By default, when Secure Device is included in the license, it is automatically activated during installation, with a strong level of security. If you need to change those parameters, use that dialog box.

- **Enforce SSL.** Forces devices to communicate by using an SSL transport. All HTTP (unsecure) requests from devices will be rejected.

- **Strong Authentication.** Enables strong authentication by generating a Strong ID for devices that is then used as a second method of authentication during the enrollment process.

- **Strong ID Valid Once.** Allows Strong ID passcodes to only be used once. When the Strong ID is used once to generate a device certificate, it cannot be reused. The device has to be revoked and re-authorized.

- **Certificate Renewal.** Sets the renewal time for certificates used in Strong Authentication mode. A setting of zero disables the certificate renewal process.

- **Always Add Device.** Registers devices automatically into Device Manager even when Secure Device is activated.

- **Block Rooted Android and iOS Enrollment.** Enabling this function blocks rooted or jailbroken devices from enrolling.

- **8 Char Strong ID.** Enables a Strong ID character string that is limited to 8 characters.

- **Enable SHP Console for Users.** Enables or disables the Self-Help Console for user management of devices.

- **XDM/SHP console max inactive interval.** The time (in minutes) between client requests before the server invalidates a log on session. If you set the value to zero, log on sessions do not timeout. For example, if the console max timeout value is set to 1 (one minute) and a user logs on and does not interact with the UI for over one minute, then the user is logged off. The console might still appear as if the user is logged on until the user attempts to interact with the UI, but then the console will be refreshed and the user will see the log on page.

- **iOS agent auto logout (minutes).** Length of time before an iOS agent user is logged off due to inactivity.

- **Enable client cert authentication for iOS.** If enabled, iOS enrollment agent uses certificate authentication. If disabled, iOS enrollment agent uses session-based authentication.
To enable Strong ID

Strong ID is a form of 2 factor authentication used to provide an extra layer of extra security when enrolling a device. Devices cannot enroll until the device’s serial number or IMEI is known. When you enable Strong ID, Citrix recommends enabling the character string to be 8 characters in length.

1. In the Device Manager console, click Options > Security.

2. You can add the devices manually or import the devices from the Devices tab by using the serial number of IMEI, which generates a Strong ID for the device.

When users are ready to enroll their device, users need to call support personnel and give the serial number or IMEI. Support personnel can then prove the Strong ID from the device properties.
Scheduling Option for Hardware and Software Inventory

The Scheduling option enables you to globally enforce hardware inventory and software deployments for those devices that are always connected to Device Manager.

When you want your devices configured to always be connected to Device Manager (‘Always On’ or ‘Permanently Alive’); for example, you may want a device to be always connected to Device Manager in the event you need to remotely wipe the device in case of a data security breach. Using policies, you can configure your devices to always be connected to Device Manager.

If you want your devices to always be connected to Device Manager, you can configure options that the device is always on or permanently alive. For example, you want the device to be connected in case you need to remotely wipe the device for whatever reason. By using policies, you can configure device to maintain a constant connection to Device Manager.

Using this option allows you to set the time interval (in minutes) that a hardware inventory and a software deployment runs.

For more information, see Configuring Deployment Schedules.
Configuring System Settings for iOS

The following systems settings apply to your iOS Devices only:

- Store User Password. Provides the following options:
  
  - Enable. If you select Enable, a user's password on the iOS Connect app is securely stored and used for ongoing authentication with the Device Manager. On the user's device Connect app, the logon/logout button will be enabled, and the user will be required to log in again if the user manually logs out.
  
  - Disable. If you select Disable, Device Manager does not store a user's passwords and uses a certificate for all ongoing authentication with Device Manager. On the user's device Connect app, the logon/logout button will not display, and the user will never be logged out.

  **Note:** Note that when this setting is selected, you can allow users to register and authenticate with a domain password because an enrollment invitation overrides this setting when other enrollment modes are configured.

- User property for VPP country mapping. The mapping used to choose the property pool of the Apple Volume Purchase Plan. This code allows a user to download apps from app stores specific to country based on this mapping. For example if your user property is US, you will not be able to download the apps if the VPP code for the app is distributed in the UK.
Configuring Role-Based Access Control

You can configure the following settings for role-based access control:

- Access Role Based Access Control Settings
- Create a New Access Control Role (Associate Actions with Roles)
- Add Groups to a Role
- Associate Users with Roles

To configure role-based access control

1. In the Device Manager Options console, in the left pane, expand Access Control and then click Role Based Access Control.

2. In the right pane, click New.

3. In the Create a Role dialog box, enter a name for the role, select the features you want to enable for the role and then click Create.

To add groups to a role

When you create a new role, you can also associate a user group with the role as part of the role definition.

1. In the Device Manager Options console, in the left pane, expand Access Control and then click Role Based Access Control.

2. In the right pane, select a role and then click Edit.

3. In the Edit a role dialog box, select the feature access you want to associate with a role, and then select the group you want to have access to the role. Any group, and the group’s users, that you select receives access to the selected features.

4. Click Update to save the changes.

To associate users with a role

When you create a new role, you can associate users with the role.

1. Select the Users tab and double-click a user in the user table. Or, click New User.

2. In the New User dialog box, enter the user name and password, and then in Role, select the role you want to associate with the user.
3. Click Create.
Configuring Automated Actions

With Automated Actions, you can configure Device Manager to perform actions based on user or device properties, events, or the existence of applications on devices.

For example, you can configure the following Automated Actions:

- You can automatically notify users whose iOS or Android devices is jailbroken or rooted that they are in violation of company policy and that the device will be selectively wiped if the device is not brought into compliance.

- You can automatically enforce a geo-fencing policy whereby if a user’s device leaves a defined geographical perimeter, the device is blocked from accessing your organization’s email, is selectively wiped, or is revoked.

- You can alert users automatically when mobile devices are roaming domestically or internationally and that they may be charged extra for the service.

- You can wipe a user’s device automatically when the user leaves the company, and can disable the user’s Active Directory account, so that the user can no longer access your organization’s data.

- You can place a user’s device into an Out Of Compliance state automatically if the user installs a blacklisted app, and you can send the user a notification informing them that they have broken the organization’s mobile app policy.

**Note:** Before you can use Automated Actions to send automated notifications, make sure you have configured notification servers for SMTP and SMS so that Device Manager can send the message. For details, see To configure a Notifications SMTP Server and To configure an SMS Notifications Gateway.
How Automated Actions Work

You can configure Automated Actions in Device Manager to trigger an event when a user device is out of compliance. You configure the following settings when you configure Automated Actions:

- **Trigger.** The state that must exist to cause the event.
- **Condition.** The setting that defines the trigger explicitly.
- **Action.** The result that occurs if the trigger conditions are met.
- **Options.** The ability to delay an action to notify users of the policy violation and allow time for users to remedy the condition.

Before you start using Automated Actions, consider the following:

- If devices are shared between two users and you want to re-enroll the device to the second user, make sure that you delete the device entry from the Device Manager Devices tab before enrolling the second user.

- Automated Actions are only triggered when a device connects to Device Manager. For example, a notification is not sent to a device until the device attempts a connection back to the server. Likewise, if any of the managed devices are currently blocked by Secure Mobile Gateway, notifications are not sent to those devices until users initiate an Active Sync activity, such as receiving email or if the device synchronizes with Exchange.

- You can deploy Automated Actions to anonymous devices if you deploy the package to anonymous users. You cannot perform Notify (SMTP/SMS) Automated Actions on anonymous (unauthenticated) users.

- The only Automated Actions you can perform on unmanaged devices - that is, on devices that are revoked, have been selectively wiped, or are not enrolled - are the Notification and Set as Out Of Compliance actions.

- The Out Of Compliance action keeps a device in that state until another action explicitly changes the state of the Out of Compliance property.

- You cannot set the Secure Mobile Gateway block notification cannot on a device that is not enrolled.

- If you are using an Automated Action to detect when users disable their location servers on an iOS device and you want to send a notification, wipe, or revoke the device, you must enable Report if location services are disabled when you configure an iOS geo-tracking policy. For details, see To configure geo-tracking on iOS devices.

- If you want to create an Automated Action based upon a user whose Active Directory account is disabled, you can use the Event Trigger named 'AD Disabled User'.
How Automated Actions Work

- If you create custom notification templates of the following type - Out of Compliance and AD Disabled User - you cannot select the templates when you configure an Automated Notification.

- There is a default one-hour waiting period for event-based triggers. Recurring notifications may be delayed due to the original event that causes the notification to be sent. For example, if you configure Device Manager to send a recurring notification every hour, but users do not receive the notifications. The reason for the delay is due to the fact that recurring notifications are not sent until the configured trigger occurs again after the Repeated Wait time expires.
Choosing Automated Actions Trigger Types

Triggers are the states, events or properties that cause an automated action to occur. There are four categories of triggers: Device Property, User Property, Applications, Event. Each trigger can contain multiple types.

The following table provides a few examples of triggers and trigger events.

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device Property</td>
<td>Useful device properties you can use as triggers for automated actions:</td>
</tr>
<tr>
<td></td>
<td>· Jailbroken or Rooted. If users jailbreak their device, you can set an action to notify the user and if the user does not undo the jailbreak in a given amount of time, selectively wipe the device.</td>
</tr>
<tr>
<td></td>
<td>· Out Of Compliance. If a device is put into a state of being out of compliance, you can block that device from the SMG (and thus corporate email) and notify the user.</td>
</tr>
<tr>
<td></td>
<td>· Passcode compliant. If this trigger’s condition is false, then you can set the device to Out Of Compliance or selectively wipe the device.</td>
</tr>
<tr>
<td></td>
<td>· Perimeter Breach. If the device leaves the geo-perimeter defined in an iOS geo-fencing policy, this condition can set and used to notify the user, wipe the device, and so on.</td>
</tr>
<tr>
<td></td>
<td>· Many more: Look in the Automated Actions dialog to view all user properties.</td>
</tr>
<tr>
<td>User Property</td>
<td>Useful user properties you can use as triggers for automated actions:</td>
</tr>
<tr>
<td></td>
<td>· Active Directory failed login attempts. If an Active Directory user attempts to log in more times that allowed, you can notify the user that they will have to wait a certain time period before they can try to log in again.</td>
</tr>
<tr>
<td>Applications</td>
<td>This trigger allows you to specify whether or not an app is installed on a device, by name, and then set an appropriate action such as notify or set the device as out of compliance.</td>
</tr>
</tbody>
</table>
## Event

The following system events can be used as triggers in automated actions:

- Secure Mobile Gateway block. A user’s device has been blocked by the Secure Mobile Gateway and the device lost access to your organization’s email.

- Device unmanaged. The device lost its ability to connect to and communicate with (and thus be managed by) the Device Manager server.

- Device jailbroken. A user broke the iOS user agreement and warranty in order to install unauthorized software.

- Device not blacklist or whitelist app compliant. A user’s device breaks an app blacklist or whitelist policy, you can choose an action to perform.

- Device revoked. The device has lost ability to connect to the Device Manager server.

- Device international roaming. A user’s device is roaming internationally.

- Device domestic roaming. A user’s device is roaming domestically.

- Location perimeter breach. A user’s device has gone outside of a defined perimeter.

- Location services disabled. The location services on a user’s device is disabled.

- Active Directory disabled. If you disable a user’s Active Directory account, such as when an employee leaves a company.

## Types of Automated Actions

The following list details the types of actions you can configure to occur automatically based on trigger type.

- **Selective Wipe.** Clears organizational data from the device while retaining personal information and selected settings. The MDM profiles and all packages pushed by Device Manager to the device are removed. The device can, however, be re-enrolled at a future time.

- **Revoke.** Revoking a device prohibits any further connection from the device. If the device attempts to connect to Device Manager, the MDM profile and all packages deployed to the device are removed. The device is barred from re-enrollment unless it is re-authorized by an administrator.

- **Set as Out of Compliance.** The device is given a property named *Out of Compliance* and the property is set to True. When a device is out of compliance (has this property set to true), then it appears in the Out of Compliance on the Dashboard Alerts widget.
Automated Actions Example: Notifications for Blacklisted Apps

This topic is an example procedure that illustrates using Device Manager Automated Actions to set up an automatic notification to inform users when they install a forbidden (also known as “blacklisted”) app on their device. You can manage user devices to make sure that a work device installs the approved list of apps only, and that the device does not have any forbidden apps installed.

This example shows the following tasks:

- Configure the notification template you want to send.
- Create an Applications Access policy to designate an iOS app named Word with Friends for Free as forbidden (blacklisted).
- Create an Automated Action that sends a notification when a device violates a forbidden Applications Access policy.
- Deploy the Automated Action and Applications Access policy to your device in a deployment package.
- Install the Words with Friends for Free on your iOS device.
- Receive the Notification.

To configure a notification template

When users install a forbidden app on their device, you can send the correct notification by using a template for the message that is sent when the non-compliant blacklist or whitelist trigger is correctly configured.

By default, all notification templates are configured to use the \{user.mail\} macro, which uses the email address of the device owner who receives the notification. If you want notification emails to be sent to an administrative user; for example, to notify an administrator that a device has been jailbroken, you can enter the administrator email address in the To field.

1. In Device Manager, click Options.
2. In the Server Options dialog box, in the left pane expand Notification Templates.
3. In the right pane, under Notification Templates, click Non Compliant Blacklist / Whitelist.
4. In the Edit a Notifications Template dialog box, on the Settings tab, in Channels, select the channels of communication you want to use.
5. Click the SMTP tab and do the following:

   a. In From, enter the name or email address from whom the notification is sent. This is not a mandatory field, however Citrix recommends adding the name or email address.

   b. In To, leave the command ${user.mail}. If you modify the To field, the email might not be sent correctly.

   c. In Message, you can modify the message except for the macros
      ${firstnotnull(device.TEL_NUMBER,device.serialNumber)} and
      ${outofcompliance.reason whitelist_blacklist_apps_name)}. If you modify or remove the macros, the email might not be sent correctly.

6. Click Update. When you click Update the template is ready for the Automated Action.

Next you will create a blacklist for an app, so you can use the blacklisted app as a trigger for your automated action later. This example uses the Words with Friends Free app.

### To create an Applications Access policy for a forbidden app

1. In the Device Manager web console, click the Policies tab.


3. Click New Applications Access Policy.

4. In the Add a new Applications Access Policy dialog box, type Words with Friends for Free.

5. In Access policy, click Forbidden (blacklist).

6. In OS type, select the iOS.

7. Click New app.

8. In the Add a new application dialog box, enter the following:

   a. In App Name, type the name of the app. For example, type Words with Friends Free.

   b. In App bundle ID, type the bundle name of the app. For example, type com.zynga.WordsWithFriendsFree.

9. Click Create. This will create the application in the list. The app appears in the list in the Add a new application dialog box.

10. Click Create again to create the Application Access Policy. Once created, you can add this policy to a deployment package and deploy to the devices you want to manage.

Next, you create an Automated Action that sends a notification email to users when they install a blacklisted app on their device.
To create an Automated Action

1. In Device Manager, click the Policies tab.

2. In the left pane, under Global, click Automated Actions and then in the right pane, click New.

3. In the New automated action dialog box, do the following:
   a. In Name, enter Blacklist Notify.
   b. Under Trigger, in Trigger Type, select Applications and in Name, select Installed.
   c. Under Condition, in Condition, select Is and then in Value, enter WordsWithFriendsFree.
   e. Under Action, in Template, select Non Compliant Blacklist / Whitelist.
   f. Under Options, select Delay and then configure 10 minutes.
   g. Under Options, select Repeat wait and configure one hour. This option allows you to delay sending the notification message in the event that there is a communication failure between the device and Device Manager.

4. Click Create.

In the last task, you will create a deployment package that contains Automated Actions and then push that deployment to user devices.

To deploy automated action and Applications Access policy to devices

Once on your device, you install the blacklisted app to trigger both the notification message that your device is out of compliance, and to trigger the Secure Mobile Gateway block on the server.

Citrix recommends that you create separate deployment packages for your Automated Actions and deploy them separately from other packages. Additionally, make sure you configure Deploy to anonymous users in the Groups of users page of the package, to include those users who may have removed their agents, or who have had their Active Directory account disabled.

You run the Create New Package wizard to deploy packages. During the wizard, you select the following:

- Groups to which the policy is deployed.
- Resources that include the Automated Actions you created and the Software Inventory resource.
1. In Device Manager, click the Deployment tab, and then click New Package > New iOS Package.

2. In the Create New Package wizard, in the Package Name window, enter a name for the package and then click Next.

3. In the Groups of users window, select a group you want to deploy this policy to and then click Next.

4. In the Resources to be deployed window, under Available Resources expand the Automated Actions section and select the two Automated Actions you previously created in the last step. Then, click the right arrow to add the resource to the deployment package.

5. Next, in the Available Resources list, under Applications Access Policy, select the Forbidden policy you previously created and click the right arrow button to add it to the package. Click Next.

6. In the Deployment schedule window, select the If not deployed Start Now option. Click Next.

7. In the Deployment rules page, click Next.

8. In the Package summary page, click Finish.

9. When the wizard is complete, in Device Manager, click Deploy to deploy the packages.

When Device Manager finishes the deployment, select the deployment package, and then click the Details button to see information about the success of the package deployment. When the package shows as deployed, then you can move on to the next and step and install the blacklisted app on your iOS device.

When the users targeted in the deployment install the blacklisted app on their iOS device, Words with Friends Free, users receive a notification message that the app is not allowed.
Configuring Location Services Triggers

1.

2. In the Location Services - Configuration creation dialog box, select Report if location services are disabled and then click Create.

3. In the New automated actions dialog box, do the following:
   a. In Name, type a name for the automated action.
   b. Under Trigger, in Trigger type, select Event and in Event, select Location services disabled.

4. Click Create.
Showing Automated Actions That Have Run

You can view all of the automated actions that have run from inside of Device Manager at any time.

1. In Device Manager, click the Policies tab and then select Show Executions.
Automated actions allow you to change the status of a device from a state of compliance to a state of non-compliance based upon specific conditions. For example, you can set an Automated Action to change a device to a state of Out Of Compliance=True if the device has been jailbroken or rooted, if the user disabled location services on the device, or if the user installs a blacklisted application.

In a cases where a user’s device is put into an out of compliance state, and then the user fixes the device so that the device is in compliance, you will need to configure a policy to deploy a package that resets the device into a compliant state.

For example, let’s say you want to define the following three compliance policies in your organization by using Device Manager Automated Actions:

1. Location Services Policy. This policy states that if a user disables location services on their device, then the Automated Action should then set the device property Out of Compliance to True.

2. Blacklisted App Policy. This policy states that if a user installs a blacklisted app on their device, then the Automated Action should then set the device property Out of Compliance to True.

3. Jailbreak Policy. This policy states that if a user jailbreaks their device, then the Automated Action should then set the device property Out of Compliance to True.

**Naming and Setting the Order of Deployment Packages**

Device Manager deploys packages to target devices. When you create your Automate Action compliance policies, you need to name your policies in a very specific way, so that they are run in the correct order.

Device Manager deploys packages according to their name, deploying those packages in an alphanumeric order. Thus, you want to make sure that you deploy your Compliance Reset Automated Action package first and that it does not reset any of the other Automated Action packages that are designed to track compliance device compliance.

When you name your policies, make sure that the global compliance reset policies deploy first and then deploy your Automated Action compliance deployment packages.

In the example above, you might want to create three packages to track device compliance, the Geo-fence, Blacklist, and Jailbreak policies. Automated Actions tracks the devices and sets the devices to Out of Compliance=False when the user violates the policy. You also want to be able to reset the devices when the user brings the device back into compliance.
For example, you want to reset the device property when the Device Manager detects that the device is out of compliance. You want this policy to run before any other policy. You can provide the name `aaa-OOC-Reset` so that it will run before the policies that can set a device out of compliance.

You can create an Automated Action by setting the device property to out of compliance if users disables location services on the device. If you want this policy to run after the reset policy, you can give the policy the name `aab-location-services-disabled`. You can then set the delay for a specific number of minutes so this policy runs after the compliance reset Automated Action that runs before this policy.

You can also create an Automated Action that sets the device property to out of compliance if users install a blacklisted app on their device. You can give the policy the name `aac-blacklisted-app` and set the delay for four minutes so it runs after the two policies preceding this one.

You can create an Automated Action to set the device property to out of compliance if users jailbreak or root their device. You can give the policy the name `aad-jailbreak` and set the delay to five minutes so it runs after the three policies preceding this policy.

### Setting the Order of Compliance Packages

Your last step to make sure that your compliance policies run in the correct order. To do so, create Deployment Packages and use the same names you used for the Automate Actions. You follow the same principles in naming that you use for Automated Actions. When you use this naming conventions, you can make sure that the packages deploy in the same order as the Automated Actions.
Troubleshooting Automated Actions

To check whether or not automated notification was sent to a user, you can try a few things:

- Check the Deployment of the package that contained your Automated Action to make sure is actually deployed.

- Check the Device Manager Device Event Log and see what if any of the events specified in your automated actions have run.

- Check the Device Manager Device Sent Notifications Log and see which notifications have been sent, which have failed, who received them, when they were sent, and more.
Configuring Notifications

You can use notifications in Device Manager to do the following:

- Communicate with select groups of users easily from the Device tab, such as all iOS device users, users whose devices are Out Of Compliance, all users with employee-owned devices, or all users with unmanaged devices, and so on.

- Enroll users and their devices into Device Manager.

- Automatically notify your users (through Automated Actions) when certain conditions are met, such as when a user’s device is about to be blocked from corporate access due to compliance policy violations, or when a user’s device has been jailbroken or rooted.

Notifications are used to send messages over three different channels: SMTP, SMS, and Agent Push (currently iOS only).

Before you can send notifications, you must configure a notifications server and a SMS gateway and carrier SMS gateway. Also, you must select a notification channel in the notification template.

**Note:** Port 25 must be opened from the Device Manager server located in your DMZ to point back to the SMTP server on your internal network for notifications to be sent successfully.
Sending Ad-Hoc Notifications

You can send a one time, ad-hoc notifications in Device Manager to single or multiple users directly to their devices using SMTP (email), SMS, or Agent Push.

1. On the Devices tab, select a single or select multiple devices. You can choose to filter the list of devices depending on your purpose. For example, you might want to send a message to all users who have jailbroken devices, or send a message to all users whose devices are listed as Out Of Compliance.

2. Select the devices to which you want to send notifications and then click Notification.

3. In the Notifications dialog box, enter the following information:
   a. From. Enter who you want to be shown as the sender of the notification (optional).
   b. To. The users associated with the devices you selected will automatically be added to the notification recipient list. If you want to add other users beyond the list of devices you selected, you can enter the user’s email address as known by Device Manager (case sensitive) and then click the plus icon to add the user.
   c. Template. You can choose a template to fit the purpose of your notification. For example, if you want to notify users whose devices have been jailbroken and are out of compliance, you can select a custom ad hoc notification template built for this purpose.
   d. Message. You can enter text, or if you choose a notification template, this field is populated with the text from the template.
   e. Channel. Select the communication channel you want to use to send the message, SMTP (email), SMS, or Agent Push (iOS only).

4. Before you send the notification, if you are sending the message via SMS, and you do not have a Nexmo subscription or SMS gateway server configured in Device Manager, click the Detailed Device List button to check if the recipients you have selected can be contacted through the Notification mechanism.

5. In the Detailed Device List dialog box, you can troubleshoot any of the devices that show red lights, which indicate channels of communication that are not currently working to send notification. The red lights indicate the recipients who may not receive the notification unless you add a carrier SMS gateway and address to use for sending the notification. The green lights in the SMTP column indicate that the SMTP server is functioning and will send the notification via email.

6. To manually enter an SMS carrier gateway and address, select the recipient and fill out the appropriate information.

7. When you are finished adding the SMS information, click Close.

8. Click Send to send the notification. Device Manager either delivers the message or queues it for sending. If the message is queued, the Sent Notification Log report indicates the results. Queuing occurs because either the system is busy (sending
Sending Ad-Hoc Notifications
To create a custom notification template in Device Manager

1. Click Options.

2. Click Notification Templates and then click New.

3. In the Create a Notification Template dialog box, on the Settings tab, enter the following information for your template:

   - Name. Enter a name for the template that indicates its use and purpose. For example, if this is a warning message regarding banned apps, you could name it Banned App Notification.

   - Description. Enter a brief description of this notification’s purpose.

   - Notification Type. Determines the Automated Action event type the template is used with.

   - Channel. Select the channels through which you want to send the notification. Agent push is currently for iOS only.

4. Click the SMTP tab and then enter the following information:

   - From. (Optional) Name used in the email From field. Only enter a value here if you do not want to use the default value from the Notification Server definition.

   - To. An email address, system macro, or list (delimited by semicolons). System macros are used when sending automated action notifications. The system macro ${user.email} is the default To field.

   - Subject. Enter a generic subject line for the message.

   - Message. Enter message text. If you want to use system macros in your custom notification template, open one of the predefined notification templates and borrow one of the commonly used macros, such as the macros used for users or devices.

5. Click the SMS tab and then enter the following information:

   - To. A system macro or mobile number. There are two system macros for use in enrollment templates and non-enrollment templates. For enrollment templates, use ${user.mobile}. For non-enrollment templates, use ${firstnotnull(device.TEL_NUMBER,user.mobile)}.

   - Message. Enter a message text that the user will see when the message is received.

6. Click the Agent tab and then enter the following information to be used for agent push notifications (iOS only):

   - To. Enter the following variable - ${device.TOKEN} - for the device’s token ID, which is used to identify and communicate with the device via agent push notification.
To create a custom notification template in Device Manager

- **Message.** Enter a message text that the user will see when the message is received.
- **Sound File.** Select a sound file to be played when the user receives the push notification on their device.

7. When you are finished, click Create.
Using Notification Templates

You can use notification templates in Device Manager when you do the following:

- Send enrollment invitations inviting users to enroll their devices.
- Send ad hoc notifications notifying users their devices are jailbroken or letting users know important IT information, without using a template.
- Configure Automated Actions to send notifications, such as an automatic notification when a user's device has a blacklisted app or has moved beyond an organization-defined geo-fencing policy.

Device Manager comes with a set of predefined templates that reflect the capabilities of the Automated Actions feature. Each template reflects a distinct type of event that Device Manager automatically responds to for each and every device in the system.

You can modify a pre-defined notification template, but you cannot delete one. Citrix recommends that you do not edit or modify the macros (for example, ${user.mail}) used inside of pre-defined templates, or they may not work.

The following table describes the predefined notification templates that come with Device Manager:

<table>
<thead>
<tr>
<th>Template name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android Download Link</td>
<td>Provides a download link Web address for users who are enrolling their Android devices into Device Manager.</td>
</tr>
<tr>
<td>Enrollment</td>
<td>Provides a Web address to the Device Manager server that allows users to enroll their devices.</td>
</tr>
<tr>
<td>Enrollment URL</td>
<td>Provides a special enrollment Web address that allows users to enroll their devices securely, combined with other forms of authentication, depending on the chosen enrollment mode.</td>
</tr>
<tr>
<td>Enrollment PIN</td>
<td>Provides a one-time generated PIN that is used in PIN based enrollment modes</td>
</tr>
<tr>
<td>iOS Download Link</td>
<td>Provides a download link Web address for users who are enrolling their iOS devices into Device Manager.</td>
</tr>
<tr>
<td>Jailbroken Device</td>
<td>Provides a message indicating that a specific device has been jailbroken.</td>
</tr>
<tr>
<td>Location Perimiter Breach</td>
<td>Provides a message informing a user that the device has gone outside of a predefined geo-fencing perimeter and thus could be blocked from corporate access.</td>
</tr>
<tr>
<td>Notification Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Location Services Disabled</td>
<td>Provides a message informing a user that the device has had its location services turned off and thus could be blocked from corporate access.</td>
</tr>
<tr>
<td>Non-Compliant Blacklist/Whitelist</td>
<td>Provides a message informing a user that their device has an app installed that violates a corporate blacklist or whitelist policy.</td>
</tr>
<tr>
<td>Revoked Device</td>
<td>Provides a message informing a user that the device has been revoked and that any further connection from the device to Device Manager is prohibited. The device is barred from reenrollment unless it is reauthorized by an administrator.</td>
</tr>
<tr>
<td>Roaming Domestic</td>
<td>Provides notification when device is roaming domestically across carrier network, indicating both device and user name associated with the device.</td>
</tr>
<tr>
<td>Roaming International</td>
<td>Provides notification when device is roaming internationally across carrier network, indicating both device and user name associated with the device.</td>
</tr>
<tr>
<td>SMG Blocked</td>
<td>Provides a message stating that a specific user’s device has been blocked because it has violated a specific compliance policy.</td>
</tr>
<tr>
<td>Unmanaged Device</td>
<td>Provides a message indicating that a specific user’s device has become unmanaged (possible due to un-installation of Device Manager agent or certificates) and must be reenrolled by a specific date or the device will no longer have access to corporate email.</td>
</tr>
</tbody>
</table>
System Macros in Device Manager Notification Templates

Notification templates in Device Manager use the following system macros when you use the Automated Actions feature for automated sending. Citrix recommends that you do not modify macros in templates or else the notifications may not work.

- Notifications are sent to the correct SMTP recipient address. For example, ${user.mail}.
- Enrollment invitation Web addresses use the proper syntax to ensure secure authentication. For example, ${enrollment.url}.
- Enrollment PINs can be generated. For example, ${enrollment.pin}.
- The correct Device Manager server host name is used. For example, http://${zdmserver.hostPath}/enroll.
- The correct user device (ID, name, and so on) is used when sending notifications. For example, ${firstnotnull(device.TEL_NUMBER, device.serialNumber)}.
- The cause of a automated notification is given to the user. For example, ${outofcompliance.reason(smg_block)}.
**Configuring High Availability on Device Manager**

You can deploy up to three instances of Device Manager to create a high availability pair, which is also called a *cluster*. You configure one Device Manager instance as the primary role in the cluster and the other Device Manager instances as the secondary role in the cluster. In this deployment, the primary Device Manager listens for requests, and serves user requests. The secondary Device Manager synchronizes its data with the data on the primary. The two instances of Device Manager work as an active-passive pair, in which only one instance of Device Manager is active at a time.

If the current primary Device Manager stops responding for any reason, the current secondary Device Manager takes over and becomes the primary. The new primary Device Manager begins to serve user requests.

Device Manager in a cluster configuration requires a network load balancer to create a high availability pair as well as to distribute the load between Device Manager servers.

You need to configure the following:

- **Windows Server 2008 R2.** Install each Device Manager instance on a separate Windows server.

- **Configure the Windows servers as a cluster.**

- **Virtual IP address or host name on the load balancer.** Device Manager uses this information to route user requests.

- **SSL session persistence for ports 443 and 8443 on the load balancer.**

- **SQL Server database accessible from the Device Manager node(s) and user credentials to connect to the database.** Each node connects to the same database.

- **Network Time Protocol (NTP) server to synchronize time for all nodes and SQL DB server.**

After you install Device Manager and configure the initial settings, there are some additional configuration steps. These include:

- **Editing an xml file to replicate session information on all cluster nodes in the Tomcat cluster.**

- **Enabling clustering on Device Manager.**

- **Configuring properties on the Tomcat server.**

- **Copy certificates from cluster node 1 to cluster node 2.**

- **Stopping and starting the Device Manager Windows service.**
You can also use the PostGRE SQL database for high availability. If you use this database, you need to run a utility to import database information to Device Manager.
To install Device Manager on cluster node 1

1. Clear the Database server check box if there is already a MS SQL server in your network.

2. On the Configure database connection screen, create a Device Manager database on your MS SQL server.

3. Install Device Manager on Cluster Node 1.

4. On the certificate creation screen, use the public virtual IP address or FDQN of the hostname configured in the F5 virtual server configuration.

After Device Manager is successfully installed, open a web browser from the same host; for example, Device Manager cluster node 1. Then, open http://localhost/zdm and verify that the Device Manager web console appears. Then, stop the Device Manager Windows service.
To install Device Manager on cluster node 2

1. Install Device Manager on Cluster Node 2 and clear database install. Remember to use the same database name as that of Cluster Node 1.

2. Copy the following files from Cluster Node 1 in <installation_dir>\tomcat\conf to the same place on Cluster Node 2.
   - https.p12
   - pki-ca-devices.p12
   - pki-ca-root.p12
   - pki-ca-servers.p12
   - pki-ca-root.crt.pem

3. Import the certificates; do not create new certificates. The installer prompts you to enter passwords with which certificates were created (during installation of cluster node 1). Only the Keystore password text box appears.

4. Enter the same keystore password which was used in 'cluster node 1' for the following screens.

After Device Manager is successfully installed, open a web browser from the same host - cluster node 2, go to http://localhost/zdm and then verify that the Device Manager web console appears. Stop the Device Manger Windows service.
To configure a Device Manager Tomcat Cluster

Tomcat clustering is used to replicate session information on all cluster nodes. In an event of a Tomcat server being unavailable on a cluster node, device connections can fail over to Tomcat servers on other cluster nodes because the state is being preserved across all nodes in the cluster.

**Note:** Make sure to update the configuration files/command to all cluster nodes.

1. Open file `<installation_dir>\tomcat\conf\server.xml` in wordpad and add a `<cluster>` section after the following element: `<Engine name="Catalina" defaultHost="localhost">:

   `<Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster">
   `<Manager className="org.apache.catalina.ha.session.DeltaManager"
      expireSessionsOnShutdown="false"
      notifyListenersOnReplication="true"/>
   `<Channel className="org.apache.catalina.tribes.group.GroupChannel">
      `<Membership className="org.apache.catalina.tribes.membership.McastService"
         address="228.0.0.8"
         port="45560"
         frequency="500"
         dropTime="3000"/>
      `<Receiver className="org.apache.catalina.tribes.transport.nio.NioReceiver"
         address="auto"
         port="4000"
         autoBind="100"
         selectorTimeout="5000"
         minThreads="3"
         maxThreads="6"/>
   `<Sender className="org.apache.catalina.tribes.transport.ReplicationTransmitter"/>
   `<Transport className="org.apache.catalina.tribes.transport.nio.PooledParallelSender"/>
   `</Sender>
   `<Interceptor className="org.apache.catalina.tribes.group.interceptors.TcpFailureDetector"/>
   `<Interceptor className="org.apache.catalina.tribes.group.interceptors.MessageDispatch15Interceptor"/>
   `</Channel>
   `</Cluster>`

   `<!--
   `<Deployer className="org.apache.catalina.ha.deploy.FarmWarDeployer"
      tempDir="${catalina.base}/war-temp"
      deployDir="${catalina.base}/war-deploy"
      watchDir="${catalina.base}/war-listen"
      watchEnabled="true"/>
   -->>
To configure a Device Manager Tomcat Cluster

</Cluster>

2. After copying the above contents, check for the following elements in server.xml:

   - Membership. Determines cluster membership. address: 228.0.0.8 (multicast address)

   - port. 45560 (multicast address and the port determine cluster membership)

   - frequency. 500 (broadcast ping send frequency. Must be smaller than timeToExpiration)

   - dropTime.3000

   - Receiver. Responsible for listening to session tomcat session replication messages

   - address. auto (listening address)

   - port. 4000 (port number used to listen for session replication messages)

   - autoBind. 100 (number of ports to try : 13000 to 13099)

   - selectorTimeout. 5000 (select operation selector timeout)

   - minThreads. 3 (work thread pool configuration)

   - maxThreads. 6 (work thread pool configuration)
To configure the Device Manager Server

1. Edit the ew-config.properties file
   (<installation_dir>/tomcat/webapps/zdm/WEB-INF/classes).

2. Change the following line from false to true:

   # CLUSTERING
   cluster.everywan.enabled=false

   To

   cluster.everywan.enabled=true

3. Add the following line: cluster.hibernate.cache-provider=com.opensymphony.oscache.hibernate.OSCacheProvider

   Your cluster configuration should look like the following example:

   # CLUSTERING
   cluster.everywan.enabled=true
   cluster.hibernate.cache-provider=com.opensymphony.oscache.hibernate.OSCacheProvider

4. For the DAO configuration, verify that the following properties exist. If not, add them.

   For MS SQL. dao.configLocation=classpath:\com\sparus\nps\dao/hibernate-native.cfg.xml

   For MySQL database. dao.configLocation=classpath:com/sparus/nps/dao/hibernate-mysql-hilo.cfg.xml

   For other databases:
   dao.configLocation=classpath:com/sparus/nps/dao/access/hibernate-hilo.cfg.xml

5. Please add the following properties in ew-config.properties:

   # Everywan cluster shared secret for application connection
everywan.secret=everywan

   # Everywan node name (used on load balancer front end)
cluster.everywan.nodeName=auto
To configure the Device Manager Server

# Everywan direct IP access (ex. used by remote support)

cluster.everywan.directAccess=auto

# Everywan broadcast

cluster.everywan.broadcast.address=228.0.0.8
cluster.everywan.broadcast.port=45561

Note: It is recommended that you change the
cluster.everywan.nodeName=auto to node1 and node2 rather than leave as auto, as follows:

The following parameters are used:

- cluster.everywan.nodeName. "node1" (or node2, node3, and so on).
- cluster.everywan.directAccess. "auto" (search for the first IP address of the first network interface). If you want to assign a specific IP address, use: "ip:192.168.1.251".
- cluster.everywan.broadcast.address. "228.0.0.8" (UDP broadcast address).
- cluster.everywan.broadcast.port. "45561" (UDP broadcast port).

Important: This broadcast address, "228.0.0.8:45561" must be different from the one used by Tomcat server in server.xml.

For cluster.everywan.directAccess, you can use the following parameters:

Important: In order for Remote Support to work if the node has two or more nics, you might need to put the node IP here.

- eth1. Use the first IP address of eth1 interface.
- ip:192.168.1.128. Use the specified IP address.
- lo. Use the first IP address of the lo interface (127.0.0.1).
To configure Tomcat oscache.properties

File oscache.properties is located under <installation_dir>\tomcat\webapps\zdm\WEB-INF\classes.

1. Use wordpad to open the file. At the end of the file, look for JGroups configuration. It looks like the following example:

    cache.cluster.properties=UDP(mcast_addr=228.0.0.8;mcast_port=45566;diagnostics_addr=228.0.0.8;diagnostics_port=45567;mcast ... ;down_thread=false;up_thread=false):pbcast.GMS(join_timeout=5000;join_retry_timeout=2000;shun=false;print_local_addr=true)
    cache.cluster.multicast.ip=228.0.0.8

2. Check the following parameters:
   · mcast_addr= 228.0.0.8
   · mcast_port=45566
   · diagnostics_addr= 228.0.0.8
   · diagnostics_port=45567
   · cache.cluster.multicast.ip= 228.0.0.8
      · mcast_addr and mcast_port, diagnostics_addr and diagnostics_port are used to check the Hibernate cache consistency among the cluster nodes. They must have the same values on all the cluster nodes.
      · cache.cluster.multicast.ip must have the same address as mcast_addr.
To configure the Tomcat applicationContext.xml file

1. Open applicationContext.xml file under
   <installation_dir>\tomcat\webapps\zdm\WEB-INF\ and verify the following values:

   <import resource="classpath:push_services.xml" />
   <import resource="classpath:ios_configuration.xml" />

   <import resource="classpath:cluster_configuration.xml" />

   <import resource="classpath:deploy-scheduler.xml" />
To run update-hilo-sql on all databases besides MS-SQL

Only for PostGres database, run the PostGres administrator utility pgadmin3.exe located under 〈installation directory〉\postgres\bin.

1. Open File > Add Server and then connect to the postgres database name/instance.

2. Open the query tool and then import update-hilo.sql located under 〈installationdir〉\tomcat\webapps\zdm\sql-scripts\sql_update\PostgreSQL and then execute the same.
To overwrite the .pem file

1. Back up the following files:
   - cacerts.pem
   - cacerts.pem.jks
   - certchain.pem
   - https.crt.pem
   - https.p12.pem
2. Copy and overwrite the files from 'Cluster node 1' <installation_dir>\tomcat\conf to 'Cluster Node 2' <installation_dir>\tomcat\conf.
To start the Device Manager windows service

1. Start Device Manager windows service on both nodes.

2. Verify that individual instances are working. (For example, open browser with URL http://127.0.0.1/zdm).

3. Create a test user on any Device Manager instance.
To test the cluster setup

1. The Virtual Server IP address (in this case, IP 172.30.1.221) should be reachable.

2. Verify that ports 80, 443, and 8443 are open on the virtual server IP address. You can telnet to the virtual server IP address and port 80, 443 and 8443 or a port scanner utility.

3. Open a browser and go to URL http://172.30.1.221/zdm. This should redirect to one of the cluster nodes and eventually open the Device Manager web console.

4. Open a browser and go to URL https://172.30.1.221/zdm. This should redirect to the one of the cluster nodes and eventually open the Device Manager web console.
Enrolling Users and Devices

In order to get users’ devices under management, you need to enroll the devices into Device Manager. You first install the Device Manager client software on the user device, authenticate the user’s identity, and then install Device Manager and user’s profile, so you can manage the device remotely and securely. After the devices are enrolled, you can perform device management tasks, such as applying policies, deploying applications, pushing data to the device, locking, wiping, locating lost or stolen devices, and more.

To enroll users, you must first add users to Device Manager if you have not yet established an Active Directory connection. The topics in this section describe the subsequent required steps for enrolling users:

- Configure enrollment modes - Default, SHP.
- Configure notification servers - SMTP and SMS.
- Configure the enrollment notification template.
- Send enrollment notification.

**Note:** Before you can enroll iOS device users, you need to request an APNs certificate. See [Requesting an APNs Certificate](#) for more information.
Requesting an APNs Certificate

In order to manage iOS devices, you need to set up and create an APNs (Apple Push Notification service) certificate from Apple for use with XenMobile Device Manager as described in the iOS Mobile Device Management (MDM) guidelines. You need to generate a certificate signing request (CSR) for submission to Apple on either a Windows server using Microsoft IIS or on a Mac computer (one or the other). Optionally, you can use OpenSSL to generate the signing request as well.

In order to enable this feature and allow for Device Manager to communicate with and manage iOS devices, you need to request an APNs certificate from Apple. Then, you need to import the certificate to Device Manager before you can enroll iOS devices into the Device Manager system.

**Note:** The APNs certificate from Apple enables mobile device management via the Apple Push Network. If you accidentally or intentionally revoked the certificate, you will lose the ability to manage your devices.

### APNs Request Process

<table>
<thead>
<tr>
<th>Steps</th>
<th>APNs Certificate Request Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Generate a Certificate Signing Request (CSR) with Microsoft Internet Information Server (IIS). On Windows 2008 R2 Server, you can generate a CSR from any IIS or on Mac OS X by using the Keychain Access application. (.txt)</td>
</tr>
<tr>
<td>2.</td>
<td>Submit CSR to Citrix for signing</td>
</tr>
<tr>
<td>3.</td>
<td>Citrix signs the CSR with its MDM signing certificate.</td>
</tr>
<tr>
<td>5.</td>
<td>Download the APNs certificate from Apple.</td>
</tr>
<tr>
<td>6.</td>
<td>Export the APNs certificate as a p12 certificate.</td>
</tr>
<tr>
<td>7.</td>
<td>Import the certificate into Device Manager.</td>
</tr>
</tbody>
</table>
Apple MDM Push Certificate Migration Information

MDM push certificates created in the iOS Developer Enterprise Program have been migrated to the Apple Push Certificates Portal. This will impact the creation of new MDM push certificates and the renewal, revocation and downloading of existing MDM push certificates. This does not impact other (non-MDM) APNs certificates.

If your MDM push certificate was created in the iOS Developer Enterprise Program:

- It has been migrated for you automatically.
- You can renew it in the Apple Push Certificates Portal without impacting your users.
- You need to use the iOS Developer Enterprise Program to revoke or download a preexisting cert

If none of your MDM push certificates are near expiration, no action is needed at this time. If you do have an MDM push certificate that is approaching expiration, contact your MDM solution provider, then have your iOS Developer Program Agent log in to the Apple Push Certificates Portal with their Apple ID.

All new MDM push certificates must be created in the Apple Push Certificates Portal.

To create a new MDM push certificate, contact your MDM solution provider, then visit Apple Push Certificates Portal and login with a valid Apple ID. The iOS Developer Enterprise Program will no longer allow the creation of an App ID with a Bundle Identifier (APNs topic) that contains com.apple.mgmt.
To create a Certificate Signing Request with Microsoft IIS

The first step for generating an APNS certificate request for iOS devices is to create a certificate signing request. On Windows 2008 R2 Server, a CSR can be generated from any Internet Information Server (IIS) using the following steps.

1. Open Microsoft IIS.

2. Double-click the Server Certificates icon for IIS.

3. In the Server Certificates window, click Create Certificate Request.

4. Type the appropriate Distinguished Name (DN) information, and then click Next.

5. Select Microsoft RSA SChannel Cryptographic Provider for the Cryptographic Service Provider and "2048" for bit length, and then click Next.

6. Enter a filename and specify a location to save the CSR, and then click Finish.
To create a Certificate Signing Request on Mac OS

1. On a Macintosh computer running Mac OS X start the Keychain Access application located under Applications > Utilities.

2. Open the Keychain Access menu and select Preferences.

3. Select the Certificates tab, and then change the options for OCSP and CRL to Off. Close the Preferences window.

4. From the Keychain Access menu select Certificate Assistant > Request a Certificate From a Certificate Authority.

5. The Certificate Assistant prompts you to enter information to start your CSR. Enter the following information:
   a. Email Address.
   b. Common Name. The email address and common name can be that of the individual or a role account responsible for the management of certificate.
   c. CA Email Address. Email address of the Certificate Authority.

6. Select the Saved to disk and Let me specify key pair information options.

7. Click Continue.

8. Enter a name for your certificate signing request (CSR) file and save it to a location that you can easily retrieve the certificate request file, and then click Save.

9. In the next window, specify the key pair information by selecting the Key Size of 2048 bits and the RSA algorithm, and then click Continue.

10. The generated and saved CSR file is now ready for upload when performing the next part of the Apple APNs certificate request process. Click Done when the assistant completes the CSR process.
To submit Certificate Signing Request to Citrix for signing

Before you can submit the certificate to Apple, you need to submit the newly created CSR to your Citrix sales representative so it can be signed for use with Device Manager. The signed file will be returned to you in the .plist format.

1. Email the CSR file to your Citrix account team.

2. Citrix will prepare an MDM signed CSR and return it to you as a .plist file.
To create a p12 APNs Certificate Using Microsoft IIS

To use the Apple issued MDM certificate with Device Manager, you need to complete the certificate request in Microsoft IIS, export the certificate as a .pfx (p12), and then import the APNs certificate into Device Manager.

1. Open Microsoft IIS.
2. Click the Server Certificates icon.
3. In the Server Certificates window, click on Complete Certificate Request.
4. Browse to the Certificate.pem file from Apple. Type a friendly name or the certificate, and then click OK.
5. Select the certificate that you just completed and then click Export.
6. Specify a location and filename for the .pfx (p12) certificate and a password, and then click OK. Note: You will need the password for the certificate during the installation of Device Manager.
7. Copy the .pfx certificate to the server that Device Manager will be installed on.
8. Log on to the Device Manager web console as an administrator, or as a user with access to the About tab.
9. Click the About tab and then click Update APNS Certificate.
10. In the Update APNs Certificate dialog box, browse to find the appropriate file path to the APNs certificate file (in the .pfx format) on your computer and then enter a new password.
11. Click Load APNs Certificate.
12. Click Update.
To import an APNs certificate into Device Manager

If you have requested and received a new Apple Push Notification service (APNs) certificate, you import the APNs certificate into Device Manager to either add the certificate for the first time or to replace an existing certificate.

1. Log on to the Device Manager web console as an administrator, or as a user with access to the About tab.

2. Click the About tab and then click Update APNS Certificate.

3. In the Update APNs Certificate dialog box, browse to find the appropriate file path to the APNs certificate file (in the .pfx format) on your computer and then enter a new password.

4. Click Load APNs Certificate.

5. Click Update.
To renew an APNs certificate

In order to renew an Apple Push Notification service (APNs) certificate, you need to perform the same steps you would if you were creating a new certificate. Then, you visit the Apple Push Certificates Portal and upload the new certificate. On the Certificates Portal, the only difference when renewing the certificate is that you click Renew. You must have a developer account with the Certificates Portal in order to access the site.

**Note:** To determine when your APNs certificate expires, in Device Manager, click the About tab and then look in the APNs certificate information section.

1. Generate a Certificate Signing Request (CRS) by using Microsoft IIS.

2. Submit the new CSR to Citrix for signing.

3. Submit the signed CSR to Apple at [Apple Push Certificate Portal](#). After logging on, you see your existing certificate or you may see a certificate that was imported from your previous Apple Developers account. Choose the Renew option for that certificate.

4. Generate a p12 APNs certificate by using Microsoft IIS.

5. Update the new APNs certificate to the Device Manager server. Log on to the Device Manager web console, click the About tab, and then click Update APNS Certificate.

6. In the Update the APNS Certificate dialog box, locate the APNS file, enter the certificate password and then click Update.
To create a Certificate Signing Request using OpenSSL

If you do not want to use Microsoft Internet Information Services (IIS) to create a Certificate Signing Request (CSR) to submit to Apple for your Apple Push Notification service (APNs) certificate, you can use OpenSSL instead.

**Note:** In order to use OpenSSL to create a CSR, you need to first download and install OpenSSL from the OpenSSL web site.

1. On the computer where you have installed OpenSSL, execute the following command from a command prompt or shell:

   ```bash
   openssl req -new -keyout Customer.key.pem -out CompanyAPNScertificate.csr -newkey rsa:2048
   ```

2. You will see the following message for certificate naming information. Enter the information as requested.

   You are about to be asked to enter information that will be incorporated into your certificate request.
   What you are about to enter is what is called a Distinguished Name or a DN.
   There are quite a few fields but you can leave some blank
   For some fields there will be a default value,
   If you enter ".", the field will be left blank.
   
   Country Name (2 letter code) [AU]:US
   State or Province Name (full name) [Some-State]:CA
   Locality Name (eg, city) []:RWC
   Organization Name (eg, company) [Internet Widgits Pty Ltd]:Customer
   Organizational Unit Name (eg, section) []:Marketing
   Common Name (eg, YOUR name) []:John Doe
   Email Address []:john.doe@customer.com

3. At the next message, enter a password for the CSR private key

   Please enter the following 'extra' attributes to be sent with your certificate request
   A challenge password []:
   An optional company name []:

4. Send the resulting CSR to Citrix.

   Citrix prepares an MDM signed CSR and returns it to you.
To generate a p12 APNs certificate using OpenSSL

After you use OpenSSL to create a Certificate Signing Request (CSR), you can also create a p12 Apple Push Notification service (APNs) certificate with OpenSSL.

1. At a command prompt or shell, execute the following command:

   ```
   openssl pkcs12 -export -in MDM_Zenprise_Certificate.pem -inkey Customer.key.pem -out apns_identity.p12
   ```

2. Enter a password for the p12 certificate file. Remember this password because you need to use it for the certificate when you upload it to Device Manager.

3. Note the location for the p12 certificate file and then copy the file to the Device Manager server so you can use the Device Manager web console to upload the file.
Configuring Enrollment Modes

You can use the enrollment options in Device Manager to configure device enrollment modes. You can use the Enrollment feature to send enrollment notifications to groups of users. The notifications invite users to easily enroll their devices into Device Manager, use a download link for the Device Manager client software, and use a link to enroll by using their internal network credentials.

You can use the enrollment options to choose varying levels of security for Device Manager enrollment, such as a one-time PIN to ensure the identity of the user, or multi-factor authentication (user, password, and PIN).

When you send an enrollment invitation, Device Manager uses the default mode in the Options dialog box unless you first modify the enrollment mode before sending the notification. The following table lists the enrollment modes you can configure in Device Manager.

<table>
<thead>
<tr>
<th>Enrollment mode</th>
<th>Description</th>
</tr>
</thead>
</table>
| High Security           | This enrollment mode sends the user the following three emails:  
                           - An email with a download link that allows the user to download and install the Connect client app.  
                           - An email with an enrollment invitation web address, that allows the user to launch the client app and enroll the user's device.  
                           - An email with a one-time PIN that the user must enter when enrolling the device, along with the user's Active Directory (or local) user name and password.  
                           When using this method, the user can only enroll by using the web address in the notification. If the user loses the notification invitation, the user cannot enroll with the sent invitation. You can, however, send another invitation. |
| Invitation URL           | This enrollment mode sends a single notification to the user that contains a web address that, when clicked, opens the Connect client app. The Device Manager server name and an Enroll button appears. The user taps Enroll to enroll the user's device into Device Manager. |
| Invitation URL + Password| This enrollment mode sends a single notification to the user that contains a web address that, when clicked, opens the Connect client app. The Device Manager server name appears, along with a field where the user must enter a password. |
## Configuring Enrollment Modes

<table>
<thead>
<tr>
<th>Enrollment Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invitation URL + PIN</td>
<td>This enrollment mode sends the following emails:</td>
</tr>
<tr>
<td></td>
<td>• An email with an enrollment invitation web address that allows the user to download and open the client app, install the app and enroll the users' device in Device Manager</td>
</tr>
<tr>
<td></td>
<td>• An email with a one-time PIN that the user must enter when enrolling the device, along with the user's Active Directory (or local) password.</td>
</tr>
<tr>
<td></td>
<td>Using this method, the user can only enroll by using the web address in the notification. If the user loses the notification invitation, the user cannot enroll with the sent invitation. You can, however, send another invitation.</td>
</tr>
<tr>
<td>Two Factor</td>
<td>This enrollment mode sends a single notification to the user that contains a web address and a one-time PIN. When the user clicks the web address, the Connect client app opens. The Device Manager server name appears, along with two fields where the user must enter a password and the PIN number.</td>
</tr>
<tr>
<td>Username + Password</td>
<td>This enrollment mode sends a single notification to the user that contains a web address that, when clicked, opens the Connect client app. The user then enters a user name and password to enroll the user's device into Device Manager.</td>
</tr>
<tr>
<td>Username + PIN</td>
<td>This enrollment mode sends the following emails:</td>
</tr>
<tr>
<td></td>
<td>• An email with an enrollment invitation web address that allows the user to download and open the client app and then enter a user name and password to enroll the device into Device Manager.</td>
</tr>
<tr>
<td></td>
<td>• An email with a one-time PIN that the user must enter when enrolling the device, along with the user's Active Directory (or local) password.</td>
</tr>
<tr>
<td></td>
<td>If the user loses the notification invitation, the user cannot enroll with the sent invitation. You can, however, send another invitation.</td>
</tr>
</tbody>
</table>
To configure a Notifications SMTP Server

In order to send enrollment invitations to your users, you need to configure the Device Manager SMTP notifications server and SMS Gateway notification server to be used for user enrollment.

1. From inside the Device Manager web console, select the Options link.

2. In the Options dialog, select Notification Server on the left, and then click New -> SMTP Server.

3. In the general tab, configure the SMTP server you want to use for Device Manager notifications.
   a. Name: Name used to represent this SMTP server account.
   b. Description. Optional description of the server.
   c. SMTP Server. SMTP server hostname.
   d. SMTP Port. Port to be used by the SMTP server.
   e. Secure Channel Protocol. Select secure protocol used by your SMTP server, if configured.
   f. Username. SMTP server login username login.
   g. Password. SMTP Server login user password.
   h. Microsoft SPA. Select if your SMTP server is an Exchange server and is configured to use Microsoft Secure Password Authentication (SPA).
   i. From Name. Name shown in the From box when a client receives a notification email from this server. E.g., Corporate IT.
   j. From Email. Email to be used if an email recipient replies to the notification.
   k. Test Configuration. Click to send a test email notification.

4. In the Advanced tab, configure the following settings:
   a. Number of SMTP Retries. Number if times to retry a failed send.
   b. SMTP Timeout. Timeout duration to wait when sending an SMTP request. Increase this number if message sending is continually failing due to timeouts. Caution: Decreasing this number could increase the number of timed out and undelivered messages.

5. Max. Number of SMTP Recipients. Maximum number of recipients per email message.

6. Click Create.
To configure an SMS Notifications Gateway

Device Manager currently supports only Nexmo SMS gateway, which is a commercial (pay to use) 3rd party gateway. If you do not have a Nexmo subscription, or haven’t activated a Nexmo account, then Device Manager will attempt to send SMS messages using the recipient's SMS Carrier Gateway.

You must have Nexmo configured as your SMS Notifications Gateway in order to send SMS notifications, both manual and automated.

1. In the Options dialog under Notifications Server, select New -> SMS Gateway.

2. In the SMS Gateway dialog configure the following information:

   a. Name. Name to be used for the SMS Gateway configuration.

   b. Description. Optional description for the gateway.

   c. Gateway. Nexmo is the only SMS commercial (pay to use) 3rd party gateway that Device Manager currently support. If you do not have a Nexmo subscription, or haven't activated a Nexmo account, then Device Manager will attempt to send SMS messages using the recipient's SMS Carrier Gateway (if a phone is Verizon, for example, it will be sent to 1234567890@vtext.com.)

   d. Key. Key provided by Nexmo when activating your Nexmo account.

   e. Secret. Secret (or password) provided by Nexmo.

   f. Virtual Phone Number. When sending to North America phone numbers (with the +1 prefix), this must be a Nexmo virtual phone number; otherwise, it can be a meaningful label or name. Virtual numbers can be purchased on the Nexmo website.

   g. HTTPS. Select if you want to use HTTPS to transmit SMS requests to Nexmo.

   h. Default Recipient Country. Default SMS country prefix to be used for recipients in your organization. Always starts with a +. Click the link to look up codes for specific countries.

   i. Use Carrier Gateway. Select if you want to use the recipient's carrier gateway to send the SMS notifications. If carrier gateway is not available, then Nexmo will be used to send the message (if an account has been purchased and activated). Deselect this option if you only want to use Nexmo and never use the carrier gateway.

   j. Test Configuration. Click to send a test message using the current configuration. Any obvious errors, such as authentication or virtual phone number errors, will be detected immediately. Messages are received in the same timeframe as those sent between mobile phones.
To configure an SMS Notifications Gateway

3. Click Create.
To configure a notification template

Create and update notification templates used in Automated Actions, Enrollment, and standard Notification message sending. Templates are used to send messages over 3 different channels: SMTP, SMS, and Agent Push (currently iOS only). Notification templates that state ‘No’ in the Deletable column are default, predefined notification templates that are provided for the main features listed above, but these can be edited and customized to suit your needs.

1. In the Device Manager web console, click the Options link.

2. In the Options dialog, select Notifications Templates from the left side.

3. Click New to create a new template, or double-click an existing template to edit it.

4. In the Notification Template Settings tab, enter or modify the following information:
   - a. From. Enter a sender for the notification here, which can be either a name, an email address, or both.
   - b. To. This pre-built notification templates utilizes user macros to ensure that notifications are sent to the correct SMTP recipient address. We recommend that you do not modify macros in templates. For more information on Device Manager macros, see Using Device Manager Macros in Policies.
   - c. Subject. Subject for the notification.
   - d. Message. Pre-built notification templates utilize macros to ensure that notifications messages contain the proper information and links. You can edit this field, but we recommend that you do not modify the macro.

5. In the Notification SMS tab, enter or modify the following information:
   - a. To. This pre-built notification templates utilize user macros to ensure that notifications are sent to the correct SMS recipient number. We recommend that you do not modify macros in templates. For more information on Device Manager macros, see Using Device Manager Macros in Policies.
   - b. Message. Pre-built notification templates utilize macros to ensure that notifications messages contain the proper information and links. You can edit this field, but we recommend that you do not modify the macro.

6. In the Notification Agent tab, enter or modify the following information that will be sent to the user through agent push:
   - a. To. Pre-built notification template utilize user macros to ensure that notifications are sent to the correct recipient number. We recommend that you do not modify macros in templates. For more information on Device Manager macros, see Using Device Manager Macros in Policies.
   - b. Message. Pre-built notification templates utilize macros to ensure that notifications messages contain the proper information and links. You can edit this field, but we recommend that you do not modify the macro.
To configure a notification template

c. Sound File. You can choose a sound file to be played when the message is received by the recipient.

7. Click Create (or OK if you are editing the template).
Enrolling User Names with Special Characters

The following special characters *can* be used for user names of the devices you want to enroll using Device Manager:

` ` ~ | ! @ # $ % ^ & * ( ) _ -

The following special characters *cannot* be used in user names for device enrollment:

`< > / \ = + , "`
To configure Self Help Portal enrollment

You can configure Device Manager enrollment so users can use the Self Help Portal to enroll their devices.

If you have Self Help Portal privileges in Device Manager, you can select the type of enrollment mode with which you would like users to enroll. After you set the enrollment mode, users can log on to the Self Help Portal and generate enrollment links that allow them to download the Connect app and enroll their devices, or they can choose to send themselves an enrollment invitation. When they receive an enrollment invitation, they can download and install the Device Manager client software and enroll their devices.

1. In Device Manager, click Options in the upper-right side of the console window.

2. In the XenMobile Server Options dialog box, on the left side, click Enrollment Setting.

3. Select an enrollment mode, such as High Security or Two Factor and then click Edit.

4. In the Configure Enrollment Mode dialog box, enter the following information:

   **Note:** The options may differ depending on the enrollment mode you choose.

   a. In Expire After, set an expiration deadline to ensure that if a user does not enroll within a certain time period, the user cannot enroll through the Self Help Portal.

   b. In Maximum Attempts, enter the number of attempts to enroll a user can make before the user is locked out of the enrollment process.

   c. In Template For Enrollment URL, select a template to use for the Enrollment URL. For example, Enrollment Invitation generates or sends the user an email or SMS (depending on how you configured the template) that enables the user to enroll with Device Manager.

   d. In Template for Enrollment PIN, select a template used for the enrollment PIN (if you selected the PIN-based enrollment mode). This option generates and sends the user a one-time PIN to use for secure authentication during enrollment.

5. Click Update.

6. In the Configure Enrollment Mode dialog box, click SHP to set the selected mode as the default Self Help Portal enrollment mode. If you see an error message stating “Please go to Edit and make sure this mode is bound to all notification templates first,” make sure you have selected a template in Step 4.
To check sent notifications logs

If you want to check if an enrollment notification was sent, for example, if any of your users indicated they did not receive an enrollment invitation, you can check the Device Manager notification logs report to determine if the notifications were sent.

1. In the Device Manager web console, select the Reporting tab.

2. Scroll down and click the Sent Notifications Log link to generate the report. You see the report that shows all messages that were sent, the recipients of notifications, notification attempts (successes and failures), time the messages were sent, and so on.
To create custom terms and conditions for enrollment

Many companies have their own policies and legal review around using technology that connects to a corporate network. You can use your own, company-branded terms and conditions policy file to enforce your users to Accept or Decline the Terms of Use, for both iOS and Android devices. This file must be in the .pdf format to work properly.

1. First, convert your Terms and Conditions a .pdf file.
2. Name the file "enduserterms.pdf".
3. From the Device Manager web console, select the Files tab.
4. Click New File.
5. In the Import a file to the XenMobile MDM database dialog box, click Choose File to select your terms and conditions .pdf file, and then click Import. During the device enrollment process, your users will be presented with the custom terms and conditions and must Accept or Decline in order to continue the enrollment process. Declining the Terms will exit the enrollment process.
Enrolling Client Devices - By Platform

XenMobile MDM supports the following device platforms: iOS, Android, Windows Phone 8 and Windows Tablet, Windows Mobile, and Symbian. For a list of platforms versions and the features supported for each platform, see Feature Support by Device Platform. When you install Device Manager client apps, you enroll users and their devices into the Device Manager system.

Enrollment consists of the three following basic steps depending on the device type.

- Downloading client application software on the user device.

  **Note:** For iOS devices, users install the Citrix Enroll app to enroll. After you enroll on iOS and Android, the Worx app will be pushed to users' devices though the base deployment package.

- Authentication with Device Manager from the device.

- Installing security certificates on the device (iOS and Android). For Windows Phone 8, Device Manager will install Root and intermediate CA certificates for SSL authentication, and a client certificate used for further communication with the server.
To enroll your Android device in Device Manager

1. Download and install the Citrix Worx Home app on your Android phone from Google Play.
2. Tap the app and on pages that asks if you want to install Citrix Worx Home, tap Install.
3. When the installation finished, tap Open.
4. Enter your corporate email address and then tap Next.
5. Next, enter your username, password, and passcode, and then tap Sign On.
6. In the Activate device administrator screen, tap Activate.
7. Enter your Device Manager user password and tap Sign On.
8. Enter your device security token and then tap Sign On..
To unenroll your Android device from Device Manager

1. Open the Control app, tap the Menu key on your phone and then tap Settings.
2. Tap Personal, tap Security and then tap Device Administrators.
3. Clear the Connect check box.
4. In the Device Administrator dialog box, tap to select Deactivate and then tap OK.
5. In Settings, tap Application Manager.
6. Select the Worx Home, tap Uninstall and then tap OK.
To enroll an iOS device in Device Manager

To enroll your iOS device, go to the Apple iTunes App Store on your device and download the Citrix Enroll app.

1. From the iTunes app store, download and install the Citrix Enroll app.

2. On the iOS device Home screen, tap the Citrix Mobile Enroll app to start it.

3. On the Welcome screen, tap Enroll.

4. Enter your corporate email address and then tap the right arrow.

5. Enter the password and then tap Next.

6. When the device, user account, and Device Manager server is authenticated, a Safari browser starts with instructions for configuring the device. Tap 1 Install Company Profile.

7. On the Install Profile screen, tap Install.

8. If a warning page appears, tap Install.

9. Tap Done. The profile finishes installing.

10. On the Configure your device page, tap 2 Complete Enrollment.

11. When enrollment is finished, close the app.

Once you have completed the enrollment process, if you have added apps - such as Citrix Worx Home and Citrix Receives - to the iOS base package, then the user will see installation messages to install those apps. A Device Manager base package will automatically push policies, configurations, and apps to the user as soon as they enroll, and are useful for making sure first time users get the data they need on their device for device management. For more information on base package setup, see How Base Packages Work.
To unenroll an iOS device from Device Manager

1. On your iOS device, open the Citrix Worx Home app.
2. Tap and hold your finger on Configuration.
3. In the menu, tap Reset Application Data.
4. Tap and hold your finger on Configuration again, and in the menu, tap Unenroll.
5. On the iOS Home screen, hold your finger on the Citrix app and then tap the X icon to delete the app.
6. On the Home screen, tap the iOS Settings app.
7. In Settings, tap General.
8. Scroll down and then tap Profiles.
9. Select and remove all Citrix profiles.
To enroll Windows 8 devices in Device Manager

In order to enroll Windows 8 devices (Pro or Enterprise) you need a license for sideloading. For more information, see http://technet.microsoft.com/en-us/library/hh852635.aspx.

1. On the Windows 8 Tablet, open Internet Explorer.
2. Enter the web address of the Device Manager Citrix Worx Home app download: https://<devicemanager_servername>/zdm/CitrixMobileConnectAgent.exe
3. On the Windows 8 Tablet desktop, tap the Worx Home app and then tap Install.
4. After the app installs, on Start, tap the app to open it.
5. Enter the Windows domain or User Principal Name (UPN) and then tap Lookup. XenMobile searches for the server address name.
6. Tap Save.
7. Browse to the Downloads folder in Windows Explorer and then tap Run as administrator.
8. In the Run the Install Shield dialog box, tap Yes.
9. Follow the directions in the wizard. Click Finish then the installation completes. The Worx Home app opens.
10. Enter the Windows user name and password and then tap OK.
11. On the logon page, enter the Device Manager name, password, and the Device Manager server name and then tap login. The console status page appears.
To enroll a Windows Phone 8 device

In order to enroll a Windows Phone 8 device in Device Manager, users need their Active Directory or internal network email address, password, and the server web address for the Device Manager server. Then, they follow this procedure on their devices to enroll.

Note: If you plan to deploy apps through the Windows Phone company store, before your users enroll, make sure that you have configured an Enterprise Hub policy (with a signed Citrix Worx Home Windows Phone 8 App `CitroxWorxHome`).

1. On the main screen of the Windows phone, tap the Settings icon.

2. Tap company apps.

3. On the company apps screen, tap add account.

4. On the next screen, enter an email address and password and then tap sign in.

5. On the next screen, enter the web address of the Device Manager server, such as: https://<xenmobile_devicemanager_server>:<portnumber>/<instancename>/wpe. For example: https://mycompany.mdm.com:8443/zdm/wpe.

   Note: The port number has to be adapted to your implementation, but should be the same port that you used for an iOS enrollment.

6. Enter the user name and domain if authentication is validated through a user name and domain.

7. Tap sign in.

8. If a screen appears noting a problem with the certificate, the error is due to the use of a self-signed certificate. If the server is trusted, tap continue. Otherwise, tap Cancel.

9. When the account is added, you have the option of selecting ‘Install company app or Hub. If your administrator has configured a Company App store, select this option and then click done. If you deselect this option, in order to receive the Company app store, you will need to re-enroll.


11. To force a connection to the server, tap the refresh icon. If the device does not manually connect to the server, Device Manager connects to the device every fifteen minutes.

12. To unenroll, tap the trash can icon.
To enroll a Windows Mobile device in Device Manager

1. Browse to the Device Manager web address for your organization. The web address is in the following format:
   https://<zdmServerName>.domain.com/<zdmInstanceName>/setup

   **Note:** You can use the HTTPS prefix only if you have a certificate issued by a trusted authority, such as Thawte or VeriSign.

2. Tap Open.

3. After the Device Manager client installs, when prompted to restart the device, tap ok.

4. Open the XenMobile app and then enter the user credentials.

5. Verify the connection. In the heading area, a solid green ball indicates a good connection.
To enroll a Symbian device in Device Manager

1. Browse to the Device Manager enrollment web address. The format is: https://<DmServerName>.domain.com/<DmInstanceName>/setup. The setup file downloads on the Symbian device and the installation starts.

2. On the Install screen, tap OK.

3. Tap Phone Memory as the location where XenMobile agent installs.

4. When the installation is complete, tap Yes to open XenMobile.

5. On the Security Details screen, tap OK to allow XenMobile to access the phone.

6. Enter the first four numbers of the server code as 2831 and then tap OK.

7. On the Control Request Accepted screen, tap OK.

8. Enter the user name and password, server name, port, and instance name for the XenMobile server and then tap OK. The connection information appears.

9. Tap Options to review server connection details and then tap Close to finish the setup.
After you install and configure Device Manager, Citrix recommends that you make a backup of your configuration. If anything happens to the Device Manager server, you can use the backup to restore your configuration.

When Device Manager updates become available, you can download and install the update on Device Manager.

This section discusses backing up and restoring Device Manager and installing updates.
Backing Up and Restoring Device Manager

Backing up your Device Manager server installation and core application file system directory is crucial to a good disaster recovery or business continuity plan. This section describes backing up and restoring Device Manager.

You can back up Device Manager by using the following methods:

- Stop all services and then make a copy of the entire application directory on the server.

- Copy the application directories required for restoration and also perform a native SQL database server backup by using the PostgreSQL utility called pgAdmin. You can also use Microsoft SQL Server Management Studio for your version of Microsoft SQL Server.

If you want to restore Device Manager, you also use pgAdmin or Microsoft SQL Server Management Studio.
To perform a full manual backup of Devive Manager server

A very simple method for backing up a default installation of the Device Manager server is to stop all services and make a copy of the entire application directory on the server.

1. From the Services utility on the Device Manager server, stop the XenMobile Device Manager and the XenMobile Device Manager Database - PostgreSQL 8.3 services. MS SQL database installations should follow the best practices used for the particular type of SQL server installation. Online and Offline backups are acceptable as long as the backup database and transaction logs are maintained together for restoration.

2. Back up the XenMobile Device Manager database and application environment. This is accomplished by making a full directory copy of the Device Manager application directory typically located at: C:\Program Files (x86)\Citrix\XenMobile Device Manager

3. Save the full directory copy to a safe external location such as tape backup or external media storage system. This full directory backup contains the Database, Application, PKI configuration and certificates, and all configuration and log files.
To perform a directory and native SQL backup of Device Manager server

Another method of backup for Device Manager server is to copy the application directories required for restoration and also perform a native SQL database server backup utilizing the default PostgreSQL utility pgAdmin. If utilizing a Microsoft SQL Server database installation the Microsoft SQL Server Management Studio utility is used. The following steps will guide you through the process using the default PostgreSQL pgAdmin III utility only.

1. From the Services utility on the Device Manager server, stop the XenMobile Device Manager service.

2. Start the pgAdmin III utility from Start > All Programs > PostgreSQL 8.3. Database backup is performed using the pgAdmin III utility if using the default PostgreSQL database. For a Microsoft SQL Server database installation use the Microsoft SQL Server Management Studio application and follow the instructions provided by Microsoft or your database administrator to back up your database according to your needs.

3. Enter the password for the default postgres administrator account for the database. This was recorded during installation.

4. Expand the Databases branch of the servers tree in the pgAdmin utility, right-click on the xdm database object, and then select Backup.

5. Enter a directory location and new filename for the backup file then click OK.

6. When completed the backup process will show the following message window. When finished, click Done. The resulting backup file will be saved off to your predetermined location for archival and retrieval when a database restore is necessary.

7. Next, while the Device Manager service is stopped, backup at least the following directories within the main application folder:
   - C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\conf
   - C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\webapps\zdm\WEB-INF

8. Verify the backed-up directory has a complete copy of the Tomcat configuration and PKI certificates. These files are located under the parent directory: C:\Program Files (x86)\Citrix\XenMobile Device Manager\tomcat\conf

9. Verify that the backup directory also contains the license file normally found at: C:\Program Files (x86)\Citrix\XenMobile Device Manager\webapps\zdm\WEB-INF

10. The Device Manager application and database environment is now fully backed up and can be restored to the same or different system host.
Installing Patches for Device Manager

If a patch has been issued to resolve a problem that applies to your situation and Device Manager implementation, you may download the appropriate patch(es) for your system.

Patches follow the naming convention of ‘a_patch_###_xxxx.jar’ where ### signs are the version release number for Device Manager and xxxx refers to the patch number.

To install the patch, copy the file ‘a_patch_###_xxxx.jar’ to the following directory %systemroot%\Program Files (x86)\Zenprise\ZenpriseDevice Manager\tomcat\webapps\zdm\WEB-INF\lib or the directory in which you installed Device Manager.

After you copy the file to the directory, restart the Device Manager service.
Device Manager allows you to view device, user, and system information. With Device Manager you can:

- View and manage user devices by using the Device Manager Dashboard.
- View Device Manager system properties.
- View session reports and other types of reports for administrator actions, groups, users, applications, and devices.
- Add user-defined reports by using Crystal Reports.
Managing Devices with the Dashboard

The Device Manager Dashboard provides an interactive, high-level view of devices. Each section of the Dashboard displays a unique view of the devices you manage. You can use the Dashboard to do the following:

- **Perform actions on devices.** When you enable actions in a Dashboard widget, you can perform actions on multiple devices represented in a chart; for example, sending notifications to a set of devices on a particular platform.

- **View charts.** Charts show representations of devices in your environment, enabling you to view different groupings of the devices. For example, the Devices by Platform chart shows all of your managed devices by platform type. When you click a section of the chart by platform, the chart changes to show different platform versions for each operating system. If you want to view the devices that are running Android version 3.2 or 3, for example, you can click the slice that displays those operating system versions, and then the Devices tab appears showing only those devices running that version. The chart types you can view are as follows:

  - **Devices by platform.** Displays managed and unmanaged device platforms.

  - **Managed devices by platform.** Displays device platforms for all devices that are managed by Device Manager.

  - Unmanaged devices by platform. Displays devices that are not currently managed by Device Manager. Unmanaged devices appear in this chart if, for example, you perform a revoke, wipe, or selective wipe on the device; the device has an agent installed on the device the device is not enrolled, or; the device has an agent installed on it but the user profile or corporate certificate has been removed.

  - **Device by Secure Mobile Gateway status.** Displays the number of devices by Secure Mobile Gateway status: Blocked, Allowed or Unknown. You can click the bars to break down the data by platform.

  - Devices blocked by reason. Displays the number of devices blocked by Secure Mobile Gateway, grouped by reason for the block, such as devices that have blacklisted apps installed, devices that have been rooted, and so on. You can click the bars to break down the data by platform.

  - **Device ownership.** Displays the number of devices according to ownership, such as, corporate-owned, employee-owned, or if the unknown ownership.

  - **Android Touchdown license status.** Displays the number of devices that possess a TouchDown license.

  - Failed package deployments. Displays the total number of failed deployments per package. Only packages with failed deployments appear.

  - **Devices by carrier.** Drilling down on the chart by clicking the bars provides a further breakdown by platform.
To change the Dashboard chart type, click the gear icon on the lower-left of the report widget and then click a view.

- **View Dashboard alerts.** Alerts show you updates about the following device statuses:
  - New enrollments
  - Non-compliant devices
  - Inactive devices
  - Secure Mobile Gateway blocked devices
  - Number of full and selecting device wipes in the last 24 hours
  - Pending wipes
- **Send notifications.** For example, you may want to notify specific device users about an Internet virus that could affect some Android users.

You can customize the Dashboard to show exactly the information that is most relevant to your needs. You can create up to four different Dashboard views. Each dashboard configuration is saved on a per-user basis. To view the Dashboard, click Dashboard on the Device Manager web console.

### To create custom Dashboards

You can create up to four custom dashboards, in addition to the default dashboard provided by Device Manager. You can choose specific layouts and select the types of widgets to display, depending on your needs.


2. To create a new custom Dashboard layout, select a Dashboard.

3. Select a Layout Style.

4. Click a widget from the list of available widgets and then drag the widget the left to add it the Dashboard. You can edit existing Dashboard configurations by using a drag-and-drop operation to move Dashboard widgets over existing ones, thereby replacing the existing widget. You cannot remove widgets from a Dashboard; you can only add widgets.

5. Click Save and Apply.
To send notifications from the Dashboard

1. On a Dashboard widget, click Enable Actions. The Actions menu appears.

2. Click one of the bars in the graph to select the devices represented in the display. For example, if you wanted to send notifications regarding a virus to all Android users, click the Android bar.

3. Click Send Notifications.

4. In the confirmation message, click OK.

5. In the notification dialog box, enter the message you want to send to users. To send quick notifications to select groups of users, you can use the Ad Hoc template.

6. Click Send.
xmob-dm-manage-add-device-tsk

Due to technical difficulties, we are unable to display this topic. Citrix is currently fixing this problem. In the meantime, you can view this topic online:

Viewing Device Management Status

For each device you manage, Device Manager provides a variety of management status and property information, such as device management status, whether or not the device has been jailbroken, device operating system and hardware information, serial number and IMEI/MEID number, user of the device, device phone number, and so on.

Three of the most commonly used and important statuses for your device indicate whether or not a device is managed or not: Jailbroken/Rooted, SMG Status, and Managed.

The following table describes the status information and icons that you see on the Devices tab in Device Manager:

<table>
<thead>
<tr>
<th>Status</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jailbroken/Rooted</td>
<td>A green light means that the device is NOT jailbroken (iOS) or rooted (Android).</td>
</tr>
<tr>
<td></td>
<td>A red light means that the device has been jailbroken or rooted.</td>
</tr>
<tr>
<td>Secure Mobile Gateway Status</td>
<td>A green light means that Secure Mobile Gateway recognizes the device as legitimate and will allow it to access your Exchange email infrastructure.</td>
</tr>
<tr>
<td></td>
<td>A red light means that Secure Mobile Gateway recognizes the device as a potential threat to your email Exchange email infrastructure and is blocking it.</td>
</tr>
<tr>
<td></td>
<td>A gray light means that your instance of Device Manager does not have Secure Mobile Gateway installed and configured.</td>
</tr>
</tbody>
</table>
### Device Managed

A green light means that the device is managed by Device Manager, which means that the device has the XenMobile agent installed on it and that it is enrolled (and can communicate) with the server running Device Manager.

**Note:** In some cases, a device will appear as “managed” even though it does not have the XenMobile agent installed. This means that the device has likely been recognized by Device Manager through an ActiveSync connection. For example, if you import users into Device Manager who own a Blackberry or Palm device, or if they connect to their email server through Active Sync, their devices will appear in Device Manager as “managed.” Even though these devices cannot have a Device Manager agent installed, their communication with Device Manager is limited, and they cannot have policies deployed to them, it is possible to issue an ActiveSync or Blackberry wipe to them.

A red light means that the device is not currently being managed by Device Manager for the following possible reasons:

- If you perform a revoke, wipe, or selective wipe on a device.
- If the device has an agent installed on it but it was never enrolled.
- If the device has an agent installed on it but it was never enrolled.

### Anonymous

Under the User column, a status of ‘Anonymous’ can occur if a user authentication fails (wrong credentials).

When this happens, Windows Mobile, Symbian and Windows 8 devices will switch to anonymous mode. It can also happen if the user can’t be used to authenticate from a device any more.

iOS and Android devices authenticate using a client certificate, so those devices will only become ‘Anonymous’ if the user is deleted or disabled in Active Directory.
Viewing the Device Properties

When you click a device name in Device Manager and click Edit, you can view device overview information for a device type. The tabs that appear may differ slightly depending on the device.

The main tabs that appear and the information they contain are as follows:

- **General.** On this tab, you can view device properties, such as the software inventory, the device serial number, IMEI, as well as the Strong ID if the Secure Device option is available in the license installed on the server. You can also display the status of the Device Lock and Device Wipe commands:
  - The statement No device lock/wipe, if no command was sent.
  - A description and the date and time at which the command was sent or carried out.

- **Properties.** The hardware inventory appears on this tab. The list is updated automatically each time the device connects to Device Manager. For devices that use the Secure Device Option, additional tabs appear, such as Certificates and Master Keys.

- **Software.** The software inventory appears on this tab. The list includes all applications and software packages installed on a device, such as package name, author, size, installation date, and version of the software. You must request an inventory if you want to display the applications deployed through Device Manager as well as user-installed apps. To request an inventory, you need to configure a deployment from the Deployment tab. Under Resources to be deployed, select Software Inventory.

  **Note:** For Windows Mobile devices exclusively, only software programs available in the Add/Delete program menu on the device appear on the Software tab.

- **iOS Profiles.** You can view the profiles for an iOS device on the iOS Profiles tab. Profiles may include web clips, mobile device management (MDM) configurations, access permissions, and more.

  **Note:** When working with iOS configuration profiles generated with Apple's iOS Configuration Utility (IPCU), such as profiles for Exchange ActiveSync, WiFi, and VPN access with a certificate, Device Manager cannot prompt the device unless you include the certificate password in the profile when you create the certificate. You must include the certificate password in the IPCU steps, and then use Device Manager to import the profiles with the certificates.

- **Deployment.** You can view a complete real-time view of package deployment, on a per-device basis, on the Deployment tab. You can view of all packages assigned to a device, and the status of the deployment.

  **Note:** The status of pending is the same as remaining. The status means that the package has not yet been deployed.

- **Connection.** The Connections tab displays the users who have authenticated against a device. It lists the user name, and last two authentication times.
· **MDM Status.** On this tab, you can review the mobile device management (MDM) status for iOS devices. The information that appears is as follows:

  **MDM status:**

  · **INACTIVE.** The server does not expect the device to connect to it any time soon, nor does it consider it necessary.

  · **ENQUEUED.** The server is attempting to communicate with the device, but a push notification has yet to be sent to the Apple Push Notification service (APNs).

  · **ACTIVE.** The server is either currently handling a device request, or it expects the device to reply to a previously sent command.

  · **PENDING.** The server is waiting for a connection from the device.

  **Last push initiation.** The time of the most recent push notification initiated by Device Manager.

  **Last notification completion.** The time of the most recent completed push notification to the device.

  **Note:** The message “Completion of a Push notification attempt” means the notification payload was successfully sent to the server running APNs and the server did not reply with an error (which would indicate syntax errors and so on).

  **Last reply device time.** The time that a device connected to Device Manager following a push notification.
Tracking Devices by Using the Out of Compliance Property

You can use the Out Of Compliance property in Device Manager to recognize the compliance status of a device. Compliance status helps you track devices to make sure the devices comply with an organization's IT policy.

The Compliance property of a device is defined as:

- Out of Compliance = True. If a device does not meet the compliance standards and policy definitions set by an IT department, the device is out of compliance.

- Out of Compliance = False. If a device does meet the compliance standards and policy definitions set by an IT department, the device is compliant.

You can set the Out of Compliance property manually, by editing the property directly on the device. Or, you can set the property automatically by using Automated Actions. Automated Actions enable you to change the status of a device from a state of compliance to a state of non-compliance based upon specific conditions. For example, you can set an Automated Action to change a device to a state of Out Of Compliance=True if the device has been jailbroken or rooted, if the user disabled location services on the device, if the user installs a blacklisted application, and so on.

You can also configure Citrix Secure Mobile Gateway to block or allow devices from receiving email that do not meet compliance standards. When the device property of Out of Compliance = True, Secure Mobile Gateway blocks the device.

You can view the compliance status on the Device Manager Dashboard. When a device has the property set to true, the device appears in Alerts as a Non-Compliant device. When you click the Dashboard Non-Compliant Devices link, the Devices tab opens and you can view the non-compliant devices in an Out of Compliance column in which the status appears as true.
To search for and edit device properties

From the Devices tab in the Device Manager web console, you can search for a device in the list. You can also edit the device properties to add additional properties.

To search for a device

The Search option under the Devices tab is a free-form search field, in which you can search for a device by typing in information you know about a device and you can narrow your search within certain criteria as well.

1. Click the search icon and then specify one or more of the following criteria:
   - The name of one of the device’s users
   - The device serial number
   - The device IMEI
   - The model of the device
   - Device platform
   - Operating system version

   **Note:** For each search criteria, you can enter the first letters or numbers of the item you are looking for.

2. To narrow the search to specific criteria, in the Search list, select one or more of the following check boxes:
   - IMEI/MEID
   - User
   - Model
   - Platform
   - OS version
   - Serial number

To restore the complete list of devices, click x next to the Search field.
To edit the device properties

After you have added one or more devices into the repository database, you can populate additional comprehensive device data into the repository database. This ability allows administrators to maintain a detailed hardware inventory of their field devices within Device Manager. This process mirrors that of adding additional user information, minimizing training requirements.

1. Click the Devices tab.

2. Highlight the device to which you want to add additional hardware information and then click Edit.

3. Click the Properties tab and then click New Property.

4. Select either one of the included fields, or select Other to create a custom data field. This field is free form, and can contain up to a maximum of 256 characters.
Viewing Device Manager System Properties

You can view overview information on the About tab in Device Manager, including application version, license information, the state of optional security parameters, and Device Manager server properties.

You can check the Tomcat server connection status by clicking the binocular icon. The Connections tab opens showing the current Tomcat server connections by protocol and port. This view provides real-time insight into devices, users, and associated statistics.

Note: Java Virtual Machine (JVM) is the application framework used by Device Manager.
Viewing Reports

The Device Manager server repository database keeps a log of connections and data exchanges between each mobile device and the Device Manager server (Logs). Device Manager reporting provides detailed information such as by tunnel or by user. Device Manager reports are available through an integrated set of reports.

The Display a Report collection provides the following reports that assist you understand and manage your mobile device asset base:

- Session report (connection logs)
- Administrator options
- Groups, users and roles summary
- Device Software Report
- Hardware inventory
- Deployment Rule Report per device
- Deployment Rule Report per package
- jailbroken or rooted devices
- Inactive devices
- Device enrollment
- Distribution of devices
- Blacklist / Whitelist application compliance report
- Device Events
- Terms and Conditions

You can also export reports to a Microsoft Excel CSV file and delete reports from Device Manager by using the Manage Historical Data Collection. When you export the report to a CSV file, Device Manager creates a text file containing all of the activity report data for the specified range of dates.

The Delete option removes data logged before a specified date from Device Manager. Use this option carefully; it cannot be undone.
Report Types

You can view the following report types in Device Manager.

Session Reports (Connection Logs)

This report is a summary of mobile device activity. It includes total usage per user and overall data compression ratio.

The connection logs can contain a large amount of data. The date is created over a period of time by the Device Manager server and it is stored in the Device Manage repository database. Citrix recommends that you limit the use of connection logs to processing of small datasets to avoid impacting the performance of the Device Manager server.

Content reports, which are part of session reports, provide a summary of total data usage for a specified period of time. You create this report by using a custom date range. The report includes:

- List of users connected to the Device Manager server. You can view details of a specific user in the list.
- Real volume passing through this Device Manager server.
- Data traffic optimized by the Device Manager server.
- Percentage of data compression achieved by the Device Manager server or Agent software for all data streams (incoming raw data as opposed to actual data transmitted over-the-air).

Other Reports

In addition to session reports, you can also view the following reports:

- Administrator operations. Summarizes administrator activity, including insertions, updates and deletions of any resources in database.

- Groups, users, and roles summary. Summarizes the list of groups, roles and users defined in the Device Manager server, and reports the modification and creation dates of these elements. This report provides an administrative overview of all users, roles and groups creation and modification data, and is meant to assist in the administrative side of your mobile IT infrastructure.

- Device software report. Provides a summary of the installed applications within the mobile device environment.

- Hardware inventory report. This report summarizes the mobile device asset base by hardware property - such as operating system, operating system version, platform, or device type.
Report Types

- Deployment rule report per device. This report summarizes package deployments for each device. **State** refers to the deployment state; specify All states, Pending, Successful, Failed, or Not applicable.

- Deployment rule report per package. This report summarizes package deployments for each package. **State** refers to the deployment state; specify All states, Pending, Successful, Failed, or Not applicable.

- Jailbroken or rooted devices. Lists jailbroken iOS devices and rooted Windows, Android, and Symbian devices.

- Inactive devices. Lists devices that are inactive.

- Device enrollment. Lists devices enrolled during a specified period of time.

- Device distribution. List of devices owned by employees or by your organization.

- Blacklist and whitelist application compliance report. This report provides three options for device compliance reporting:
  - Blacklisted apps shows devices with apps that are not allowed and are installed on the user device.
  - Non-whitelisted apps shows devices with apps installed on the device that are not on the whitelist.
  - Missing whitelisted apps shows devices that do not have all the whitelisted apps present.
Navigating Reports

Each report uses a navigation bar to aid in moving through the report and its sections. The navigation bar allows you to export the report data, print the report, hide/reveal subsections of the report, page throughout the report, search for a specific string, and set a zoom level for the rendered data.

Many Device Manager reports present data by using a summary page, followed by one or more subsections that provides additional detailed information. You can use the Group Tree icon in the report navigation bar to view the subsections and open that subsection’s page.

Additionally, you can use the drop-down list in the navigation bar to go to a particular subsection.

Some Device Manager reports require parameters to run; parameters are supplied to a report from pop-up windows.
Adding User-Defined Reports

Reports must be in Crystal Report report format (file with a .rdp extension).

When you configure the reports, the property fields can have the following values:

- **reportFilename.** “My_report.rpt” is the personalized report in the Crystal Report.

- **format name** is the text that appears on the Device Manager Administration Console tab when you click on the link Click here for my report.

- **linkLabel** is the hypertext used to generate the report.

- **description** is the help that appears below the hypertext.

The link of the new user-defined report appears under the User-defined reports section on the Reports tab in Device Manager web console.

**To add user-defined reports**

1. Stop the Device Manager Server service.

2. Open the WEB-INF/classes/external-reports.xml file with a text editor that can read and write UTF-8 files, such as Notepad.

3. In the file, locate the tags <list></list> and add the following parameters:

   `<bean class="com.sparus.nps.reports.EWCrystalReport">`  
   `<property name="reportFilename" value="my_report.rpt"/>`  
   `<property name="name" value="my_report"/>`  
   `<property name="linkLabel" value="Click here for my_report"/>`  
   `<property name="description" value="Description of my report"/>`  
   `</bean>`

4. Save the file in UTF-8 format.

5. Restart the Device Manager server service.
Multi-Tenant Console

The Citrix Multi-Tenant Console for XenMobile MDM is a web-based console that enables consolidated management of multiple XenMobile Device Manager servers. You can use Multi-Tenant Console to manage service providers and enterprises with multiple distributed entities and to administer several physical XenMobile servers from a single site. Each server can run multiple instances (also called tenants) of XenMobile MDM Device Manager. The servers are logically independent from each other. In such cases, Multi-Tenant Console provides a single integrated management environment across the Device Manager servers.
Multi-Tenant Console 8.5

XenMobile Multi-Tenant Console is a web console that enables service providers and organizations to administer several physical servers running XenMobile Device Manager from a single site. Each server can run multiple instances (also called tenants) of Device Manager. The servers are then logically independent from each other.

In This Section

This section of eDocs introduces Multi-Tenant Console 8.5 and discusses how to configure Multi-Tenant Console.

<table>
<thead>
<tr>
<th>System Requirements</th>
<th>Provides system requirements for the Multi-Tenant Console server and web console.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deploy</td>
<td>Provides deployment information for Multi-Tenant Console.</td>
</tr>
<tr>
<td>Install and Setup</td>
<td>Provides information about how to install Multi-Tenant Console on a Windows Server 2008, as well as how to install Java JDK, Grails, and Google Web Toolkit.</td>
</tr>
<tr>
<td>Manage</td>
<td>Discusses how to add tenant hosts to the console and configure the hosts, how to add a connector, a Tomcat server, and a database connection to Multi-Tenant Console. Also, provides information on setting up cross-tenant reporting with role constriction.</td>
</tr>
<tr>
<td>Maintain</td>
<td>Contains information on maintaining Multi-Tenant Console.</td>
</tr>
</tbody>
</table>
XenMobile Multi-Tenant Console is a web console that enables service providers and organizations to administer several physical servers running XenMobile Device Manager from a single site. Each server can run multiple instances (also called tenants) of Device Manager. The servers are then logically independent from each other.

In This Section

This section of eDocs introduces Multi-Tenant Console 8.5 and discusses how to configure Multi-Tenant Console.

<table>
<thead>
<tr>
<th><strong>System Requirements</strong></th>
<th>Provides system requirements for the Multi-Tenant Console server and web console.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Deploy</strong></td>
<td>Provides deployment information for Multi-Tenant Console.</td>
</tr>
<tr>
<td><strong>Install and Setup</strong></td>
<td>Provides information about how to install Multi-Tenant Console on a Windows Server 2008, as well as how to install Java JDK, Grails, and Google Web Toolkit.</td>
</tr>
<tr>
<td><strong>Manage</strong></td>
<td>Discusses how to add tenant hosts to the console and configure the hosts, how to add a connector, a Tomcat server, and a database connection to Multi-Tenant Console. Also, provides information on setting up cross-tenant reporting with role constriction.</td>
</tr>
<tr>
<td><strong>Maintain</strong></td>
<td>Contains information on maintaining Multi-Tenant Console.</td>
</tr>
</tbody>
</table>
Multi-Tenant Console System
Requirements

To install XenMobile Multi-Tenant Console, you need an installation package that includes following files:

- XenMobile Mobility Manager for Multi-Tenant-xxxxx.exe. The installer used to install PostgreSQL and Tomcat for the servers running XenMobile Device Manager. It can also be used to install the repository database for Multi-Tenant Console.

- Java Development Kit (JDK)

- Grails framework that is included in the Multi-Tenant Console package distribution. The file name is grails-1.3.7.zip.

- Google Web Tools that is included in the Multi-Tenant Console package distribution. The file name is gwt-windows-1.5.3.zip.

- XenMobile_MTC-8.0.1.xxxxx.zip that are the files for Multi-Tenant Console.

- PsTools from Microsoft that are command line utilities.

You also need the Device Manager distribution to get the zdm.war archive, and the appropriate license file.

Multi-Tenant Console requires the following minimum system configuration:

<table>
<thead>
<tr>
<th>Component</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Computer and processor</strong></td>
<td>Intel Xeon/Pentium 4-3 Ghz or AMD Opteron-1.8 Ghz server class</td>
</tr>
<tr>
<td><strong>Memory</strong></td>
<td>512 megabytes (MB) of RAM or more recommended</td>
</tr>
<tr>
<td><strong>Hard disk</strong></td>
<td>500 MB of available hard disk space</td>
</tr>
<tr>
<td><strong>Operating system</strong></td>
<td>Microsoft Windows 2008 Server 64-bit, Standard Edition or Enterprise Edition</td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows 2008 Server R2 SP1 64-bit, Standard Edition or Enterprise Edition</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> NTFS partitioning is required for Windows operating systems.</td>
</tr>
<tr>
<td><strong>Browser</strong></td>
<td>Internet Explorer 7 or later, Mozilla Firefox 3.0 or later</td>
</tr>
<tr>
<td><strong>Additional software</strong></td>
<td>Sun JDK 6 most recent update or 7</td>
</tr>
</tbody>
</table>
Installing the Prerequisites for Multi-Tenant Console

Before you install Multi-Tenant Console, you need to install the following programs and configure the associated environment variables:

- Java JDK
- Grails
- Google Web Toolkit

**Note:** Citrix recommends that you back up the Multi-Tenant Console database if you need revert to an earlier version or if you need to restore the Multi-Tenant Console.

After you complete the following procedures, make user you set the environment variables correctly. You can set environment variables from the command-line console, by running the command `C:\Windows\System32\set`.

In addition, you need to do the following:

- Install . For details, see [Installing PsTools from Microsoft for Multi-Tenant Console](#).
- Configure a relational Database Management System (R-DBMS).

**To install the Java JDK**

1. Install the Java JDK by browsing to the directory where you downloaded the Java JDK, double-click the jdk-6u25-windows-x64 executable and then follow the instructions.

2. Browse to the directory containing the java cryptography files and copy local_policy.jar and US_export_policy.jar to the following folders (overwrite existing files):
   - `C:\Program Files\Java\jdk1.6.0_xx\jre\lib\security`
   - `C:\Program Files\Java\jre6\lib\security`

3. Ensure that the JAVA_HOME and PATH Variables are set. To do so, do the following:
   a. From the Start menu, right-click Computer and then click Properties.
   b. Click the Advanced Systems Settings tab and then click Environment Variables.
Installing the Prerequisites for Multi-Tenant Console

To install Grails

1. In the Multi-Tenant Console distribution package, locate the Grails zip file, grails-1.3.7.zip.

2. Extract the files to: C:\Zenprise\.

3. Set the Grails environment variable by doing the following:
   a. From the Start menu, right-click Computer and then click Properties.
   b. Click the Advanced Systems Settings tab and then click Environment Variables.
   c. Create a new System Variable where the Variable Value equals the path where you installed Grails.
   d. In Variable value, add %GRAILS_HOME%\bin to the path.

To install the Google Web Toolkit

1. In the Multi-Tenant Console distribution package, locate the Google Web Toolkit zip file, gwt-windows-1.5.3.zip.

2. Extract the files to: c:\Zenprise\.

3. Set the Google Web Toolkit environment variable by doing the following:
   a. On the Start menu, right-click Computer and then click Properties.
   b. Click the Advanced Systems Settings tab and then click Environment Variables.
   c. Create a new System Variable where the Variable Value equals the path where you installed the Google Web Toolkit.
   d. In Variable value, add %GWT_HOME% to the path.
Deploying Multi-Tenant Console
Deploying a New Tenant from Multi-Tenant Console

A tenant is an instance of XenMobile Device Manager that is running on a server. To deploy a new tenant from Multi-Tenant Console, do the following:

- Check that the XenMobile release you want to use is already in the console by checking the XenMobile Application part.

- Check that a connection exists to the physical server to which you want to deploy the tenant. To do so, check the Tenant Hosts part. See To add a new tenant host for more information.

- Check that a tomcat server is installed on the physical server. See Installing Multi-Tenant Console for more information.

- Check that the Tomcat server definition is done in Multi-Tenant Console by checking the Tomcat Servers part of the console.

- Define a new secured connector for the tenant. The connector will be deployed on the Tomcat server as part as the tenant deployment. It will handle all requests for the tenant and hold its Public Key Infrastructure (PKI). You complete this step in the Connectors part of the console. See To add a connector to Multi-Tenant Console for more information.

- Check that the database definition used by the tenant is already complete. You complete this step on the Database Servers part of the server. See Declaring a Database Server in Multi-Tenant Console for more information.

- Declare the new tenant and link it, during definition, to the previously declared tenant host, Tomcat server, connector, XenMobile application release, and Database server. You also have to include a license file during the definition and optionally import an existing PKI if it already existed. You complete this step in the Tenants part of the console. See To add a new tenant host for more information.
Installing Multi-Tenant Console

The complete installation of the XenMobile Muti-Tenant Console requires additional installations of the following third party tools: Java JDK, Grails web application framework, Google Web Toolkit, and PsTools. All three of these application distributions are included in the Multi-Tenant distribution.
Installing PsTools from Microsoft for Multi-Tenant Console

The XenMobile Multi-Tenant Console is designed to run in a distributed server environment. If the Tomcat server you want to manage is a remote one and not located on the same physical server as Multi-Tenant Console, however, you must install PsTools utilities on the server.

PsTools is a set of command-line utilities that enable you to manage local and remote systems. All of the utilities in the PsTools suite work on Windows NT, Windows 2000, Windows XP, Windows 7, and Windows 8. Multi-Tenant Console uses two tools included in the PsTools suite:

- PsExec to execute processes remotely
- PsService to view and control services

You download and install the preceding tools from Microsoft Technet. When you finish installing the files, you can verify the commands and accept the terms and conditions of use by entering the following commands on the command-line console:

```
psexec \127.0.0.1 cmd
psservice \127.0.0.1 query
```

Note: If you are running Windows Vista, you should run the command-line console as an administrator.

In the case of a domain user accounts, such as an Active Directory user account, on Windows XP or Vista, it is recommended that you do the following:

- Add both machines to the same domain
- Add a domain user as local Administrator to the target-machine.
- Use this domain user account for connecting with PsExec
- If your local machine is a Windows Vista machine, run the command prompt for executing PsExec or PsService as administrator (right-click > Run as administrator), otherwise connection may fail (error 5, access denied).
- Use “domain\user” syntax for authentication on command-line, for example:

```
psec.exe \1target-machine -u domain\user -p password -s cmd
```

In the case of a local user accounts, such as a Security Account Manager user account, on Windows Vista, verify the minimum requirements posted in this Microsoft forum. You have to change a UAC flag in the registry that allows you to toggle this behavior for local accounts: HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\system\LocalAccountTokenFilterPolicy
0 - build filtered token (Remote UAC enabled)

1 - build elevated token (Remote UAC disabled)

If you set the DWORD entry to 1, you will be able to remotely connect to Vista with a local admin account from XP or Vista machine since the remote login is not filtered. It is recommended to reboot the machine to take account the modification of UAC flag.

**Important:** On the remote machine, it is very important that the RPC port (135) has the status "LISTENING" to remotely execute processes. Type "psexec \hostname -u username -p password cmd" to check the remote connection.
To install Multi-Tenant Console

1. To install XenMobile Multi-Tenant Console, unzip the zdm_mtc.zip (packaged with Multi-Tenant Console setup) to a zdm_mtc directory.

2. Double-click the Multi-Tenant-xxxxx.exe and then follow the steps in the wizard.

   **Note:** Within the wizard, on the Choose Components page, if you want to use SQL Server, clear the Database server check box.
Upgrading Device Manager MultiTenant Console to version 8.5

If you are upgrading the XenMobile Device Manager Multi-Tenant Console (MTC) to version 8.5, you will need to install Java JDK version 7, upgrade the version of Device Manager for Multi-Tenant on the server host (Tomcat version 7) and then you will need to upgrade the MTC to version 8.5. Once you have performed those tasks, then you can upgrade your tenants (a ‘tenant’ is an instance of Device Manager managed by MTC). For instructions on how to upgrade Device Manager hosts to version 8.5, see Upgrading Device Manager to 8.5.

Note: If you upgrade your Multi-Tenant Console to 8.5 (XenMobile_MTC-8.5.0.xxxxx.zip) and Multi-Tenant host to version 8.5 (XenMobile Device Manager for Multi-Tenant-8.5.0.xxxxx.exe), then any tenant also needs to be upgraded to version 8.5, or they will not be manageable with the Multi-Tenant Console version 8.5.

Before you upgrade, note the following:

- If you are running a single server configuration, for example, the MTC, Tomcat and Java all on one server, then upgrading these components will mean that ALL tenants will have to be upgraded to version 8.5. You cannot run lesser version tenants on Tomcat 7.

- Tenants that exist on a remote Tomcat server running Tomcat 6 and Java 6 are not affected - these can still be managed and redeployed with the MTC version 8.5.

To upgrade MTC to version 8.5

This task shows you how to upgrade a single server MTC deployment, where both the MTC host (where there Tomcat server resides) and the MTC console have been installed on the same server. If you have deployed the Tomcat server on a different computer than the MTC Console, then the first seven steps should be performed on the server where the Tomcat server resides.

1. On the server where you have installed the Tomcat server, from the Windows Control Panel stop the XenMobile Device Manager for Multi-Tenant service.

2. If Java 6 is installed, uninstall both components using the Windows Control Panel.


4. Replace the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 7. To install JCE files, first install Java SE 7. Then copy local_policy.jar and US_export_policy.jar from the JCE zip file downloaded, and overwrite the two files of the same name located in .../Java/jdk1.7.0_x/jre/lib/security and .../Java/jre/lib/security.
5. Edit the `JAVA_HOME` Environment variable and update the installation path, as shown below:

![Environment Variables](image)

6. Run the XenMobile Device Manager for Multi-Tenant-8.5.0.xxxxx.exe installation file and follow the instructions to upgrade.
7. Select Start > Edit Service and update the Java Virtual Machine path to reference the new installation path for the Java application:

8. Next, on the server where the MTC Console is installed, back up the following two files:
   a. \install_dir\zdm_mtc\grails-app\conf\BuildConfig.groovy
   b. \install_dir\zdm_mtc\grails-app\conf\DataSource.groovy
9. Back up the \install_dir\zdm_mtc folder.
10. Delete all the contents of the \install_dir\zdm_mtc folder, but leave the folder intact.
11. Remove the following folders:
    a. C:\Users\Administrator\.grails
    b. C:\Users\Administrator\.ivy2
12. Extract the provided XenMobile_MTC-8.5.0.xxxxx.zip file to the existing empty folder: \install_dir\zdm_mtc
13. \install_dir\zdm_mtc
14. Optional for larger deployments with multiple tenants. Browse to the \install_dir\zdm_mtc folder, right-click the 'zdm_mtc.bat' file and select Edit. Modify the values shown below:

```
@if "%DEBUG%" == "" @echo off
if "%JAVA_OPTS%" == "" set JAVA_OPTS=-Xms1024m -Xmx2048m -XX:MaxPermSize=512m
set JAVA_OPTS=%JAVA_OPTS% -Dgrails.env=prod
set CMD_LINE_ARGS=%
if "%CMD_LINE_ARGS%" == "" set CMD_LINE_ARGS=run-app
REM set HOT_DEPLOYMENT=1
REM Set GWT_HOME Below
REM set GWT_HOME=c:\dropbox\get\...
CALL %GRAILS_HOME%\bin\grails.bat "CMD_LINE_ARGS%"
```

15. Right-click the 'zdm_mtc.bat' file and select Start. Note that this will take longer than usual since it recreates and builds the files in Step 11.

16. Log in to the MTC Console and check the MTC version to ensure that it is version 8.5.

17. In the MTC Console, click Tomcat Servers > Edit on the existing Tomcat server.

18. Change the Service Name to ZDMTomcat7 and then click Update.

19. Finally, upload the v8.5 war file into XDM Applications and update all tenants to this version, noting the caveats at the start of this topic.
To reconfigure Java memory size for Multi-Tenant Console

In some cases, the default Java memory sizing may be too low for the Multi-Tenant Console installation. To reconfigure the Java memory size, do the following:

1. From the Start menu, click Programs > XenMobile > XenMobile Device Manager for Multi-Tenant and then click Edit Service.

2. Click the Java tab and the modify the following settings:

   - XX-MaxPermSize = 512m
   - Initial memory pool = 1187
   - Maximum memory pool = 1187
   
   These values can be increased according to available memory on the server.
To configure an R-DBMS for Multi-Tenant Console

The Multi-Tenant Console repository requires a relational Database Management System (R-DBMS). PostgreSQL is the default R-DBMS used by Multi-Tenant Console. Alternatively, Multi-Tenant Console also supports SQL Server 2005 or 2008.

Note: The server on which you install the database needs to have NTFS partitions.

To install the SQL database

1. Using Microsoft SQL Server Management Studio, create a new database named zdm_mtc: Use or create an user with administrative rights who has full access to the database in order to configure Multi-Tenant Console.

2. Using SQL Server Configuration Manager, under Protocols for MSSQL Server, enable the following parameters: TCP/IP and Named Pipes.

3. In the SQL Server Configuration Manager, under SQL Server Services, set the SQL Server Browser to Automatic. Also, make sure that port 1433 is open from the MTC server to the SQL server if there is a firewall in between.

To create the PostgresSQL Database

If necessary, you can install PostgreSQL from the Multi-Tenant Console installer (packaged with Multi-Tenant Console). To do so, select the PostgreSQL database installation during setup and clear the other options. To manually create the database "zdm_mtc" on Postgres, issue the following SQL command once logged in the database. On PostgreSQL, enter:

CREATE DATABASE zdm_mtc ENCODING='UTF8';
To set up Multi-Tenant Console

To configure Multi-Tenant Console database access

1. Edit the file <INSTALL_DIR_ZP_MTC>/grails-app/conf/DataSource.groovy and adapt the parameters of the data source.

2. Note the Dialect and DriverClassName. Comment out relevant lines for either SqlServer or Postgres.

To configure the Multi-Tenant Console server URL

1. At the bottom of the file is the production mode. Configure the hostname/IP address and Port for the server. You should edit all modes (development, test, production).

2. Note the URL name is postgresql or sqlserver. If you are using Postgres, you will need comment out the SqlServer line and un-comment Postgres.

To configure the tcp/ip port number

Multi-Tenant Console is a web interface that needs to be accessed on a specific port number. The default port used is 8080. You need to change the port to 9090.

1. To customize the web port used, edit the file <MTC Dir>/grails-app/BuildConfig.groovy and adapt the following line:

```
// Available build settings
grails.server.port.http=9090
//grails.server.port.https=
//grails.config.base.webXml=
tent
```
To open Multi-Tenant Console

1. Navigate to C:\Program Files (x86)\Zenprise\zdm_mtc-x.x.x and execute zdm_mtc.bat. If you completed everything correctly, you should see http://localhost:9090/zdm_mtc.

2. To logon to Device Manager, browse to the following URL:

Configuring Tenant Servers for the Multi-Tenant Console Deployment

A tenant is the name for an instance of XenMobile Device Manager that is running on a server. To deploy one or more tenants from Multi-Tenant Console, you need to prepare the servers that are running the tenants. On each server, do the following:

- Install Apache Tomcat (version 6.0.16) from the Multi-Tenant Console installer by running Device Manager for Multi-Tenant-x.y.z.exe.

- Run Tomcat as a Windows service.

- Increase the size of PermGen space for Tomcat (for example -XX:MaxPermSize=128m or 256m if -Xmx is equal to 512m). To do so, click Start, click Programs > XenMobile > XenMobile Device Manager for Multi-Tenant, click Edit service and then on the Java tab, adapt the MaxPermSize parameter.

  Note: Depending on the number of tenants on a running a Tomcat instance, you will have to adapt the MaxPermSize parameter. Add 50 MB to the preceding number each time a new tenant is added.

- If the server is a remote one, i.e. not the same physical server as the one running Multi-Tenant Console, you also need to do the following:

  - Install psTools. For details, see Installing PsTools from Microsoft for Multi-Tenant Console.

  - Declare a Windows shared folder that will be used to upload the files. During the installation process, you can also install a local PostgreSQL database. For details, see To configure an R-DBMS for Multi-Tenant Console.

If you choose PostgreSQL, MySQL or a SQL Server database, Multi-Tenant Console will create the database for each tenant and the required tables. It will create also a database user for each database with all the necessary privileges (CREATE TABLE, INSERT, DELETE, UPDATE...).
To perform a Multi-Tenant Console Hot Deployment

The Multi-Tenant Console Hot Deployment allows you to upgrade a Device Manager tenant without affecting the service of other tenants that may reside on the same tenant host. This topic discusses how to set up your Tomcat environment so you can upgrade each tenant without affecting or shutting down the other hosted Device Manager tenants.

Before you follow the procedure, keep in mind the following considerations:

- If you do not have PsTools installed on your tenant host server, make sure you create a folder named C:\Pstools and that you add the folder path to the %PATH% environment variable. For details, see Installing PsTools from Microsoft for Multi-Tenant Console.

- A Multi-Tenant Console Hot Deployment only works with Tomcat version 6.0.33. If you are upgrading a tenant that precedes Version 6.5 of Device Manager or was upgraded to Device Manager Version 6.5, you will need to upgrade the Tomcat software to version 6.0.33. To upgrade Tomcat to version 6.0.33, uninstall the old version completely and then reinstall the 6.0.33 version before you perform a Multi-Tenant Console Hot Deployment.

  **Note:** If you are upgrading from a new installation of Device Manager 6.5, your version of Tomcat is correct and you can skip the Tomcat upgrade task. Tomcat version 6.0.33 is available on a new installation of Device Manager for Multi-Tenant-6.5 and later.

1. On the server that is running the Tomcat server, stop the Tomcat service.

2. From the command prompt, go to the following lib folder in the Tomcat installation directory on the tenant host: C:\Program Files\.....\zenprise\...tomcat\lib

3. From the tenant host where you are running Multi-Tenant Console, copy the catalina.jar from the ZDM MTC 6.6 distribution (zdm_mtc/web-app/WEB-INF/ew/tomcat-patch/v6_0_33) and paste the .jar to the directory on the tenant host server.

4. On the MTC console server, from the zdm_mtc/HotDeployment directory copy the following two files and paste the files in C:\Pstools folder on the tenant host:

   - DynamicConnector.bat
   - DynamicConnector.jar

5. On the tenant host where you have Tomcat installed, open the ‘tomcat\conf\server.xml’ file and make sure following key pairs are all set to true (if these key pairs are do not exist, then add them to the file):

   - unpackWARs="true"
   - autoDeploy="true"
To perform a Multi-Tenant Console Hot Deployment

- `liveDeploy="true"

6. Open the tenant server Tomcat bin folder (C:\Program Files (x86)\Zenprise\Zenprise Device Manager for Multi-Tenant\tomcat\bin) and run following command to enable Java Management Extensions (JMX):
   ```bash
   ```

7. On the server that is running your Tomcat server, start the Tomcat service.

8. On the server that is running Multi-Tenant Console, open the zdm_mtc.bat file in a text editor and set the following variable to 1: set HOT_DEPLOYMENT=1
Managing the XenMobile Multi-Tenant Console requires adding the XenMobile Device Manager application package to the Multi-Tenant Console, adding tenant hosts to the console (XenMobile Device Manager instances), adding a connector to the Device Manager server to the console, adding a Tomcat server to the console, configuring a database to be used with the console, and configuring reports.
To import and manage XenMobile application packages

XenMobile Applications are XenMobile Web application ARchive (WAR) files used by Tomcat. Those files contain the Zenprise Mobility Manager server software and will be used to deploy the tenants.

To import an application package

1. Click XenMobile Applications to view a list of the WAR files in Multi-Tenant Console.
2. To add a new application WAR file, click New Zenprise Application.
3. Click Browse to locate the .war file you want to import, add a comment (optional) and then click Import.

To edit or delete an application package

1. In the Application List, click the name of a .war file. A summary of the package appears.
2. Click Delete to delete the file or Edit to edit the file.
   
   Note: If the WAR file is linked to a tenant, a message appears stating that deleting the file will all tenants that use the file.
3. To update the WAR file to a new release, click Browse to locate the .war file you want to import, add a comment (optional) and then click Import.
4. To add a patch to the release, click New Patch. A patch may be one of the following:
   
   - A JAR file containing a Device Manager patch.
   - A CAB file containing a new Windows Mobile client.
   - An APK file containing a new Android client.
To add a new tenant host

1. To add a new tenant host, click New Tenant Host.

2. In Connection Type, select the type of connection to be used from Multi-Tenant Console to the tenant server.

3. If the server is local; for example, Multi-Tenant Console and Device Manager are running on the same physical server, in Host, enter a name for the server.

4. If you select Windows (Remote), enter the following details:
   - Host. Indicate the name or IP address of your server
   - Login / Password. Enter the credentials to be used on the remote server.
   - Shared drive path. Indicate a shared network path on the remote server used to upload all the files that will be used, such as \192.168.0.1\Temp
   - Local shared path. Indicate the same path as for the shared drive path as seen locally on the server, such as . D:\Temp

To edit or delete the tenant host

1. To edit a tenant host, click the tenant name. The tenant host summary appears.

2. If the server is a remote, click Check Connection to verify that Multi-Tenant Console can connect to the remote host.

3. Click Edit to edit the tenant host details or click Delete to delete the tenant host.
To add a connector to Multi-Tenant Console

A connector is a description of the TCP port that will be used on a Tomcat server. The Connectors tab displays the list of all the connectors for all the Tomcat servers handled by Multi-Tenant Console. For each Tomcat server and for each tenant, an HTTP connector is required. For the tenant, the connector is required to separate the PKI used by each tenant. Another HTTPS connector is also needed for iOS enrollment.

1. Click Connectors to display the list of managed connectors.

2. Click New Connector to add a connector to the list.

3. Fill in the required fields as follows:
   
   · Name. Provide a unique name for the connector.
   
   · Port. Enter the TCP port number to be used.
   
   · Max Threads. Enter number of threads on that Tomcat connector. Entering 20 is enough for the HTTP connector, but you should increase the value for an HTTPS connector because this is usually the port used by the devices to connect to the server. If your devices are in always-on mode, be careful to declare enough threads in this field.
   
   · Secure. Select this check box to declare a HTTPS connector. Leave it cleared for an HTTP connector.
   
   · Specify external port. Check the Specify external port box if the port used externally is not the same as the internal port (in the case in which Port Address Translation is used in your network).

4. Click Create.
Adding a Tomcat Server in Multi-Tenant Console

Tomcat is used as the application server by XenMobile Device Manager. To view the Tomcat instances handled by Multi-Tenant Console, click Tomcat Servers in Multi-Tenant Console.

The following information appears for each Tomcat server instance definition:

- **Service Status.** The colored bullet has the following meaning:
  - Green. Working server.
  - Red. Stopped or Paused server. Error while requesting status.
  - White. Unknown.
  - Orange. Starting or Stopping.
- **Name.** The given name of that Tomcat server.
- **Full Version.** Release number of this tomcat instance.
- **HTTP Connector.** HTTP Connector linked to the server.
- **Tenant Host.** Tenant host on which the Tomcat server is deployed.
- **Tenants.** Tenants that are using the Tomcat server.
To add a new Tomcat server definition

1. To add a new Tomcat server to the list, click New Tomcat Server.

2. Complete the following fields:
   - Name. Enter a name for the Tomcat server.
   - Full Version. Enter the release number of the installed Tomcat. The default is 6.0.16.
   - Directory paths. Enter the full local path to the Tomcat directories.
   - HTTP Connector. Pick up an available HTTP connector that will be used as the default connector for running the server. The connector has to be first declared in the Connectors part of Multi-Tenant Console.
   - Tenant Host. Pick up one of the tenant host declared on the server.
   - Service Name. Enter the Windows service name used on the host. Default is ZDMTomcat6.
   - Check the Enabled Cluster box to enable clustering of the Tomcat server. You can then specify all the cluster mode communication parameters. Default parameters are usually correct. Adapt the Tenants Direct Access, NIO Receiver Address, and Tenants P2P Address IP address if your server has more than one IP address.

   Note: Unless already running, declaring a new Tomcat server does not start the server. See Step 4 in the following procedure.

3. Click Create.

To edit or delete a Tomcat server definition

1. Click the name of the Tomcat server you want to edit.

   The Tomcat server summary appears.

2. Click Edit to edit the Tomcat server parameters.

3. Click Refresh to refresh the status of the Tomcat server.

4. Click Start to remotely start the Tomcat service. Click Stop to stop the service.

5. Click Uninstall/Delete to uninstall and delete the Tomcat server configuration.

   Note: Deleting the server removes all tenants running on the server.
XenMobile Device Manager stores its information in a repository database. You must declare all the database servers used by the tenants in Multi-Tenant Console. After you declare a database server, in the list of servers, you can click Check Connection to check the validity of the declared database server.

**To declare a new database server**

1. Click the Database Server tab.

2. Click New Database Server to declare a new server.

3. Complete the following fields:
   - Name. Enter a name for the server.
   - Driver Class Name. Enter the database server type.
   - Host. Provide the IP address of the server.
   - Port. Provide the TCP communication port of the database server.
   - Admin Name and Admin password. Provide the administrator credentials to be able to create the required usernames, tables, and so on, to be used by the tenants.
   - Driver Jar File. If the database is of type MySQL, Oracle, or DB2, enter the JDBC driver jar file.
   - SQL Server Instance Name. If the database is MS SQL, enter a name in this field.
   - SQL Server Domain Name. If the database is MS SQL, enter domain name.

4. Click Create.

**To edit or delete a database server**

1. Click the name of the database server.

2. Click Edit to edit the database server's parameters or click Delete to delete the database server.
To add a new tenant on Multi-Tenant Console

A tenant is an instance of XenMobile Device Manager. The tenant is running on a specific tenant host, using a specific connector on a specific Tomcat server and storing the data on a specific database on a database server.


2. Complete the following fields:

   a. Name. Enter a name for the tenant.

   b. Description. Enter a description of the tenant.

   c. XDM Application. Select the XenMobile application release to be used by the tenant.

   d. Tomcat Servers. Select one or more servers on which the tenant should run.

   e. HTTPS Connector. Select the HTTPS connector to be used.

   f. External DNS Name. Specify the name of the server as seen from the devices.

   g. Generate or import SSL certificates. Select whether you want to generate a new certificate chain for the tenant or import an existing certificate. If you choose to import the certificates, you are prompted for the path to all the certificates.

   h. Check Enable iOS. Select this option to enable iOS devices management and to allow the import of your APNS certificate. In this case, select a Web HTTPS Connector for iOS enrollment, your APNS certificate file, and specify your APNS Certificate private key password.

   i. Instance Name. Enter the name of the XenMobile instance. You should append this value to the server Web address to reach the tenant; for example, http://my_server/my_instance

   j. XenMobile Device Manager admin name. Enter the XenMobile administrator user name and password. These are the credentials to be used on the console.

   k. Remote support secret authentication code. Enter the authentication code used as a shared secret during the remote support sessions. The default is Zenprise.

   l. Crystal Report key code. If you want to be able to generate and use the reports in the XenMobile Administration Console, enter a Crystal Report license key.

   m. Database server. Select the database server to use to store Device Manager data.

   n. Database Name. Enter the name of the database to be created on the database server.
To add a new tenant on Multi-Tenant Console

o. Database user. Enter the credentials of the user to be created on the database to handle the tenant’s data.

p. NAT Device Access. Leave a default value to reuse the Tomcat server configuration and remote support tunnel definition.

q. Enable Cluster. Select the check box if the tenant is to be deployed on clustered Tomcat servers. If you select this option, the following advanced parameters appear:

   Broadcast UDP Port, P2P TCP Port, and OSCache Multicast UDP Port. Edit each port so that the ports do not overlap with another tenant.

To view the parameters for each tenant you have added to Multi-Tenant Console, click the Tenants tab. The following information appears for each declared tenant:

- Deploy Status. The colored bullet has the following meaning:
  - Green. Deployed.
  - Orange. To be redeployed or not-deployed.
  - Red. Does not belong to a Tomcat server but is not un-deployed.
  - White. Not associated with a Tomcat server.
- Name. Name of the tenant
- XenMobile Application. XenMobile application used by the tenant.
- Last Modified Date. Last modification date of the tenant.
- Tomcat Servers. Tomcat server running the tenant.
- HTTPS connector. HTTPS connector used by the tenant.
- Instance Name. Name of the XenMobile instance. You should append this value to the server Web address to reach the tenant; for example, http://my_server/my_instance
- Database Server. Database server used to store the tenant data.

To edit or delete a tenant

1. To edit the tenant, click Edit.
2. After you make changes to the tenant parameters or to redeploy the tenant, click Apply.
3. To apply a patch from a XenMobile Application, click Patch. You will install a patch without the need to redeploy the tenant.
4. To remove a tenant, click Uninstall/Delete. The tenant is undeployed from the Tomcat server and the declaration is deleted.
Configuring Cross-Tenant Reporting with Role Restrictions

You can view summarized data in Multi-Tenant Console individual tenants through a cross-tenant reporting capability. You can group individual tenants into different views and configure restricted access to the tenant information by using role-based access control (RBAC). You can also establish implementations that define the data that is available to specific Multi-Tenant Console administrators.

You can configure the following two scenarios for partner implementations within Multi-Tenant Console:

- New Multi-Tenant Console installation with initial partner implementation.
- Upgrade of existing deployment with new partner capability.

Most of the data for these cross-tenant reports is collected from the XenMobile Device Manager or from the Multi-Tenant Console tenants. Otherwise, you can populate the data manually. For Multi-Tenant Console deployments on the organizations' premises, the following fields are populated from an external database source:

**Note:** For XenMobile cloud deployment hosted customers, these fields are populated from the cloud deployment provisioning database.

- Hosting Region (a combination of Region + Country, in case of organizations outside of the United States)

  In the reports, for Hosting Region values, the “region” column has these values:

  - US East = United States East
  - USWEST = United States West
  All others appear as region(country) in the reports.

- Type (for example, Trial or Subscribed)

  In the reports, for Type, the “buy_status” column has these values:

  - 0 = Trial
  - 1 = Subscribed

**Note:** The values for U.S customers in the “cloud_user” table “region” should be match exactly to the values mentioned in this topic; for example, US East and USWEST.
To provide data manually for cross-tenant reports

1. Open Multi-Tenant Console database server page and create a new record.

2. Run the following SQL statements from the Multi-Tenant Console database.

   ```sql
   UPDATE DB_SERVER SET CLOUD_DB_CHECK = 'Y' WHERE DB_SERVER_ID = ?;
   COMMIT;
   ```

   **Important:** Change the ? for `DB_SERVER_ID` to the newly created Database Server record also make sure there is only one row in the `DB_SERVER` table with `CLOUD_DB_CHECK` field set to 'Y' per partner.

3. Connect to the newly created Database Server and execute the following SQL script.

   ```sql
   CREATE DATABASE zen_front;
   USE zen_front;
   CREATE TABLE cloud_user (id BIGINT, preferred_domain_word VARCHAR(255), region VARCHAR(255), country VARCHAR(255), buy_status VARCHAR(255));
   ```

4. Populate the `cloud_user` table with customer data. The `preferred_domain_word` column is used to map the tenants so you must give the column the same name as the name of the tenants in the Multi-Tenant Console tenant table. Also the `preferred_domain_word` column name should be unique.

   a. Create insert scripts and execute the scripts. For example: `INSERT INTO cloud_user VALUES(1,'foo','EU','France','0');` `INSERT INTO cloud_user VALUES(2,'bar','EU','France','1');`

   b. Create CSV files and populate the `cloud_user` table by using the CSV files.
To set up an initial Multi-Tenant Console partner implementation

- Ant is setup on the build server (in case a Web application ARchive (WAR) file is not available).

- The upgraded WAR file or the upgraded Multi-Tenant Console source stack is available.

1. Stop the Multi-Tenant Console application Tomcat server.
   
   **Note:** If an upgraded Multi-Tenant Console WAR file is available, you can skip Steps 2 and 3.

2. Run Ant sp.build from trunk. This will set up all the required libraries (.jar files).

3. Run Ant sp.build from the service_provider dir. This will build the Multi-Tenant Console WAR file.

   **Important:** There are now two build.xml files. The build.xml original file is for Windows distribution and the build-linux.xml file is for Linux. Please use the correct file to build the WAR file.

   **Note:** If the application is running with grails run-app, you can skip Step 4. You should, in this case, download gwt-window-1.5.3 from the Google Web Toolkit and set the `GWT_HOME` environment variable. If you are using zdm_mtc.bat or zdm_mtc.sh, make sure it has the `GWT_HOME` defined.

4. Place the newly created WAR file in web-app tomcat folder.

5. Start the Tomcat service.
Maintaining Multi-Tenant Console

You can maintain various functions in the Multi-Tenant Console.

You can back up the Multi-Tenant Console by using the backup and restore functions in your SQL Server or PostgreSQL database.

You can create administrative users who can configure all or some of the Multi-Tenant Console. You can add, edit, or delete administrative users.

You can restrict access to tenants by creating partners and then assigning the partner to the tenant.

You can create users in the Multi-Tenant Console and assign users to roles. When you assign users to the role, it restricts user access to specific functions in the Multi-Tenant Console. After you create a role, you can assign the user to the role by editing the membership of the role. For example, if you create the role “abcadmin” and you assign the privileges to the role, and then you add the ABC Partner, the role “abcadmin” can view the reports of ABC Partner.

When you configure partner association in Multi-Tenant Console, you allow the Multi-Tenant Console administrator to restrict access to specified tenants. When the administrators for the tenants log on to Multi-Tenant Console, on the home page, they see only the content they have privileges to view. For example, you may have configured privileges for XenMobile applications, tenant hosts, connectors, Tomcat servers, database servers, and tenants. These privileges correspond to tabs in Multi-Tenant Console. This topic describes a scenario in which the administrators have privileges to view reports. For example, they see only a Reports tab on the home page of Multi-Tenant Console where they can access a Customer Inventory or a Customer Usage report. The reports appear as a table with multiple columns of data listing information, such as:

- ID
- Hosting Region
- Service Activation
- Licensed Devices
- Software Version
- Service URL
- Number of Users Provisioned
To back up your Multi-Tenant Console installation

Because all of the Multi-Tenant Console data is stored in the database, the main backup and restore procedure is a classic database procedure. A simple copy of the files or possibly the whole Zenprise_mtc directory, is enough to ensure a reliable backup of the console.

Two other files might have been modified during installation. They are located in <installation folder>\grails-app\conf:

- BuildConfig.groovy. You can adjust the TCP port used by the interface in this file.
- DataSource.groovy. You can set the database parameters used to store all Multi-Tenant Console data in this file.
To add or change administrators on Multi-Tenant Console

If you want to change the administration console access on Multi-Tenant Console, on the Home tab, click Users and then do the following:

- To add an administrator, click New User, fill in the fields and then click Create.
- To edit an existing administrator, in the User List, click the users’ name and then click Edit.
- To delete an administrator, in the User List, click the users’ name and then click Delete.
To create a partner on Multi-Tenant Console

You should create a partner when you want to restrict customer tenant records to specific Multi-Tenant Console administrators. By associating tenants with a specified partner name, you can restrict certain users to access only the tenants that have a designated partner name.

1. Log on to Multi-Tenant Console and then click Partner. A list of partners appears.

2. Click New Partner.

3. In Create Partner, complete the following fields:
   - Name. Enter a name for the partner.
   - Email. Enter the partner’s email address.
   - Logo URL. Enter a Web address for the partner.
   - Support Email. Provide an email address for the support contact.
   - Product name.
   - Support Number. Enter the phone number for the support contact.

4. Click Create. The partner appears on the Show Partner screen.

Next, you create a user on Multi-Tenant Console.
To create a user on Multi-Tenant Console

To access the Multi-Tenant Console and customer tenant data, you must create a user.


2. Click New User.

3. Complete the following information:
   - Full name. The name of the user.
   - Login. The name the user enters to log on to Multi-Tenant Console.
   - Password. The user password.
   - Confirm password. The user password again.
   - Description. (Optional). Information about the user.
   - Email. The user’s email address.
   - Enabled.
   - Partner.

4. Click Create.

   **Important:** You cannot edit the partner information after you create the partner.

The user appears in the User List.

Next, you must assign the user to a specific role in Multi-Tenant Console.
To create a role on Multi-Tenant Console

In order to restrict a user's access to specific functions within Multi-Tenant Console; for example, allowing only reporting capabilities, you must create a role in Multi-Tenant Console with a designated list of privileges. After you create a role, you can assign the user to the role by editing the membership of the role. For example, if you create the role "abcadmin" and you assign the Reports privileges to the role, and then you add the ABC Partner, the role "abcadmin" can view the reports of ABC Partner.

1. On the Home page, click Roles.
2. Click New Role. A list of roles appears.
3. Click Create.
4. On the Privileges tab, in Authority, enter a role in the format ROLE_<your-defined-role> and in Description, enter a description of the privilege.
5. To assign privileges to the role you create, click Privileges and then select one or more privileges that you want to assign to a particular role.
6. To add a partner to a particular role, click Partner.
7. Click Create.

Next, you assign a user to a role.
To assign a user to a role on Multi-Tenant Console

After you create a role with the appropriate level of privileges and designated partner names, you can assign users to the role by editing the membership for the designated role.

1. In the Role List, click a role.
2. Click Edit Membership.
3. To add a user to a particular role, in Select User, enter the name of the user or select from the list of users that exist in the database and then click the user icon.
4. In the confirmation message that appears, click OK.
To assign a partner to a tenant on Multi-Tenant Console

1. On the Multi-Tenant Console home page, click the Tenants tab.
2. Click the tenant to which you want to assign a partner and then click Edit.
3. In Partner, click the partner name and then click Update.
To view a customer inventory report

1. Click Reports and then click Customer Inventory Report. A default customer inventory report appears.

2. Click Search or Advanced Search to identify or filter on specific tenants. Search enables you to find a specific tenant based on an alphanumeric string identifier. Advanced Search allows for a finer level of filtering based on specific column values. For example, users can filter for Licensed Devices with a value greater than 2.

To view a customer usage report
Remote Support

Remote Support is a software program installed on a Windows-based computer that allows support personnel to take remote control of the Windows Mobile devices. With Remote Support, you can:

- Display a list of all connected devices within one or more Device Manager servers.
- Display system information including device model, operating system level, International Mobile Station Equipment Identity (IMEI) and serial number, memory and battery status, and connectivity.
- Run the device task manager where you can display and end active processes and restart the mobile device.
- Run the remote file transfer that includes bidirectional file transfer between mobile devices and a central file server.
- Download and install software programs as a batch to one or more mobile devices.
- Configure remote registry key settings on the device.
- Optimize response time over low bandwidth cellular networks by using real-time device screen remote control.
- Display device skin with support of most of the mobile device brands and models and a skin editor to add new device models with mapping of physical keys.
- Enable device screen capture, record and replay with the ability to capture a sequence of interactions on the device that creates a video AVI file.
- Conduct live meetings by using a shared whiteboard, VoIP-based voice communications and chat between mobile users and support personnel.
Remote Support System Requirements

You can use the following system requirements for installing XenMobile Remote Support:

**Supported Platforms**

- Intel Xeon/Pentium 4 - 1 GHz minimum Workstation class
- 512 Mb RAM minimum
- 100 Mb free disk space minimum

**Supported Operating Systems**

- Microsoft Windows 2003 Server Standard Edition or Enterprise Edition SP1 or later
- Microsoft Windows 2000 Professional SP4
- Microsoft Windows XP SP2 or later
- Microsoft Windows Vista SP1 or later
- Microsoft Windows 7
Remote Support System Requirements

You can use the following system requirements for installing XenMobile Remote Support:

**Supported Platforms**

- Intel Xeon/Pentium 4 - 1 GHz minimum Workstation class
- 512 Mb RAM minimum
- 100 Mb free disk space minimum

**Supported Operating Systems**

- Microsoft Windows 2003 Server Standard Edition or Enterprise Edition SP1 or later
- Microsoft Windows 2000 Professional SP4
- Microsoft Windows XP SP2 or later
- Microsoft Windows Vista SP1 or later
- Microsoft Windows 7
Installing Remote Support

With Device Manager, your help desk representatives can remotely support smartphones used by a company’s mobile workforce to ensure that users have consistent, reliable, and secure access to enterprise data and applications. Device Manager has the following features:

- Remotely log onto a user’s smartphone that is located anywhere and control the device’s screen. You can see and control the user’s screen just as if they had the device physically with them. Once in control, users can watch you navigate the screen, which can be helpful in training users on how to best use the device.

- Remote Control allows you to remotely navigate and repair a device in real-time. By remotely navigating to a smartphone’s screen, you can change configurations, troubleshoot operating system issues, and disable or end problematic applications or processes.

- S Virus and malware threats are becoming increasingly pervasive on smartphones. Your security personnel can use Remote Control to isolate and contain threats before they spread to other smartphones by remotely disabling network access, stopping rogue processes, removing applications, and removing malware.

- Remote Control allows you to remotely enable the device’s ringer and call the phone, allowing the employee find and retrieve the device on their own. Smartphones that users cannot find or are stolen can also be located and wiped to ensure your sensitive data is not compromised.
To install the Remote Support Client

You can install the Remote Support Client by clicking a link in Device Manager.

When you install the Remote Support Client, you can enable users to have an icon on the device desktop and allow users to save their Device Manager credentials.

If you are deploying the Remote Support Client to a wide audience, the installer program supports the following variables:

- /S: to silently install the Remote Support Client with the default parameters.
- /D=dir: to specify a custom installation directory.

For example, run the following command: `<RemoteSupportClient>.exe /S` where `RemoteSupportClient` is the name of the installation program.

2. Follow the instructions in the wizard.
To add a remote support device manager connection

You can add a remote support connection to Device Manager, which is necessary to establish remote support connections to managed devices.

1. In the Remote Support application, to add a new connection entry, click New.

2. In the Connection Configuration dialog box, on the ZDM Server tab, enter the following values:

   a. Next to Configuration name, enter a name for the configuration entry.

   b. Next to Server IP address or name, type the IP address or the DNS name of the Device Manager host.

   c. Next to Port, type a TCP port number, as defined in the Device Manager server configuration.

   d. Next to Instance name, if Device Manager is part of a Multi-Tenant deployment, type an instance name.

   e. Next to Tunnel, type the name of the Remote Support tunnel configuration entry as defined in the Device Manager server configuration.

   f. Next to Secure connection, select the check box to indicate that the connection to the Device Manager host should be made using SSL.

   g. Next to At Remote Support startup, select the check box to connect to the Device Manager host described in this entry each time the Remote Support application is started.

   **Note:** The Connection Manager remains available to the Remote Support application via the latter’s menu options.

3. Select the Proxy tab.

4. Select Use a http proxy server and then enter the following information:

   a. Next to Proxy IP Address, type the IP address of the proxy server.

   b. Next to Port, type a TCP port number used by the proxy.

   c. Next to My proxy requires authentication, select the check box if the proxy server requires authentication before conveying traffic.

   d. Next to Username, type the user name to be authenticated on the proxy server.

   e. Next to Password, type the password to be authenticated on the proxy server.

5. Click the User Authentication tab and then select the Remember my password option.
To add a remote support device manager connection

6. Select the Remember my login and password check box to open the field elements.

7. Complete the fields on the User Authentication tab to associate a user name and password with the connection entry.

8. Click OK.

To connect to Device Manager, double-click the connection you created and then enter the user name and password you configured for the connection.
Configuring Remote Support App Tunnels

Device Manager application tunnels (app tunnels) are designed to increase service continuity and data transfer reliability for your mobile applications. App tunnels are used to define “proxy” parameters between the client component of any mobile device applications and the application server component. App tunnels can also be used to create remote support tunnels to a device for management support by clicking the Create button.

Device Manager tunneling acts as a stream buffer to overcome inherent network issues, such as irregular latency or network hopping. They also provide checkpoint restart capabilities, which is critical when bouncing between cellular data points. Furthermore, Device Manager will automatically apply on the fly data compression and AES encryption to all data traffic transiting within each tunnel.

On the Tunnels tab in the Remote Support application, you can assign a tunnel “channel" dedicated to each mobile application and monitor those applications. For each application defined on this page, Device Manager transmits and monitors the data streams in a separate tunnel.
Managing Remote Support

Remote support sessions launch a new application window and display the Remote Control window with a rendering of the controlled device. Remote control sessions can be started on all Windows and Android Mobile devices. Remote Support is available on all Windows mobile devices, but remote of devices is only available on Android Samsung SAFE devices.

Note: Remote control of iOS devices is not supported in this release.

You can view devices, users, and groups in the Remote Support application.
To start a remote support session

2. Select a device by clicking its row in the main workspace.
3. Click Control Device.
To inspect devices, users, and groups

The left-side of the Remote Support application window presents Device Manager user groups as defined in the Device Manager administrative console. By default, only groups containing users that are currently connected are shown. You can see the device for each user next to the user entry.

1. To see all users, expand each group from the left column. Those users currently connected to the Device Manager server are indicated with a green icon.

2. To display all users - including those not currently connected - click View and select Non-connected devices; non-connected users appear without the small green icon. Devices connected to the Device Manager server but not assigned to a user appear in Anonymous mode. (The string “Anonymous” appears in the list.) These devices can be controlled just like the device of a logged-in user.
Remote Support Session Options

During a remote control session, an icon appears on the handheld device to inform the user that a remote control session has been established. You can choose from an array of interactions with the controlled device including:

- Remote control over the device’s screen, including control with colors, in either the main window, or in a separate, floating window.

- Establishment of a Voice-over-IP session (VoIP) between the helpdesk and the user, and configuration of settings.

- Establishment of a chat session between the helpdesk and the user.

- Access to the task manager of the device, including memory and CPU usage, and running applications.

- Remote exploration of the mobile device’s local directories and file transfer.

- Remote editing of the Windows Mobile device’s registry.

- Display of the device’s system information.

- Display of all the device’s installed software.

- Update of the mobile device’s connection status with the Device Manager server.