MobileIron Tunnel 4.1.0 for iOS Guide for Administrators

for MobileIron Core and MobileIron Cloud

Revised: February 22, 2021

For complete product documentation see:
Tunnel for iOS Product Documentation Home Page

Revision history

TABLE 1. REVISION HISTORY

Date	Revision
February 22, 2021	Updated the field description for User Name. Added the variable to use for Cloud: SSO with Kerberos configuration field description
December 21, 2020	Added AppStation support: • New features and enhancements • Required components for deploying MobileIron Tunnel for iOS

Contents

New features and enhancements

This guide documents the following new features and enhancements:

- **Send Tunnel Debug logs using Email+**: By default, Tunnel uses the native iOS email app to email debug logs. Tunnel can now also use MobileIron Email+ to send debug logs. To set up Tunnel to use Email+ to send debug logs, add the key-value pair, UseSecureEMail = true, in the Tunnel configuration.
  For more information, see Emailing debug log information.
- **Support for iOS native Mail, Calendar, and Contact domains**: Enter one or more domains that will trigger the configured per-app VPN connection in Mail, Contacts, and Calendar apps. The Tunnel configuration for iOS and macOS provides separate fields for entering the domain information.
  Requires MobileIron Core 10.6.0.0 or MobileIron Cloud 69.
  For more information seeTunnel for iOS configuration field description.
- **Report device ID to MobileIron Access**: For Access deployments, Tunnel reports the device ID to MobileIron Access if the key SendDeviceID is configured in the Tunnel VPN configuration with the value true. The device ID is reported on MobileIron Access in **Reports > Errors**. The key-value pair is useful in identifying devices that encounter connection errors when authenticating through Access.
  See Additional configurations using key-value pairs for MobileIron Tunnel.
- **Rebranding**: MobileIron has updated the Tunnel for iOS and macOS icons and user interface color scheme. See What users see in MobileIron Tunnel for iOS Additional information about MobileIron Tunnel rebranding is also provided in the Knowledge Base article Coming Soon - MobileIron UX changes MobileIron Tunnel Android and iOS App.
- **Support for MAM-only AppStation deployments**: MobileIron productivity apps, Email+, Web@Work, and Docs@Work, in a MAM-only AppStation deployment can use Tunnel to access enterprise resources. For more information about setting up a MAM-only AppStation deployment, see the *MobileIron AppStation for iOS Guide*. The support is provided with AppStation 1.3.0 for iOS through the latest version as supported by MobileIron.
  See also, Required components for deploying MobileIron Tunnel for iOS.

About MobileIron Tunnel for iOS

The following provide an overview of MobileIron Tunnel for iOS devices:

Overview of MobileIron Tunnel for iOS

MobileIron Tunnel enables VPN capability on iOS devices. MobileIron Tunnel interacts with the MobileIron enterprise mobility management (UEM) platform, MobileIron Standalone Sentry, and MobileIron Access to secure access to enterprise resources from outside the enterprise network. The enterprise resource can be on premise or in the cloud. The MobileIron UEM platforms are: MobileIron Core and MobileIron Cloud.

About MobileIron Tunnel configuration

Configurations for MobileIron Tunnel are created in a MobileIron unified endpoint management (UEM) platform. MobileIron Tunnel receives the configuration from the MobileIron UEM client. The MobileIron client for MobileIron Core is Mobile@Work, and the client for MobileIron Cloud is MobileIron Go.

Deployment use cases with MobileIron Tunnel for iOS

MobileIron Tunnel enables native per-app and device level VPN on iOS devices. MobileIron Tunnel is part of the following MobileIron deployments for securing access to enterprise resources:

- MobileIron UEM and Standalone Sentry.
- MobileIron UEM and MobileIron Access.

The following use cases are enabled with these deployments:

- access to internal corporate URLs from the Safari browser.
- per-app VPN for managed apps (managed apps do not need AppConnect wrapping or SDK).
- device-level VPN.
- single sign-on.

App proxy provider and packet tunnel provider

MobileIron Tunnel for iOS supports app proxy provider and packet tunnel provider VPN tunnels.

For apps that use a TCP connection, such as Office or GSuite apps, create an app proxy Tunnel VPN configuration. An app proxy Tunnel VPN configuration is applicable per app only. Previously, this was the only option available with MobileIron Tunnel.

For apps that use an IP connection (such as Skype for Business an Microsoft Teams), create a packet tunnel provider Tunnel VPN configuration. A packet tunnel provider Tunnel VPN configuration can be configured to be either per-app or device-level.

For additional information about use cases with specific types of apps, see the knowledge base article in the MobileIron Support Community: iOS and macOS - What are VPN Provider Types Packet-Tunnel and App-Proxy?

Note The Following:

- Standalone Sentry supports only limited types of UDP traffic, such as DNS traffic. Audio and video traffic through Standalone Sentry is not supported. Therefore, MobileIron recommends configuring SplitUDPPortList to manage UDP traffic. See UDP traffic.
- Split-tunneling for IP routes is supported only for device-level VPN. Configure the routes to through Tunnel in the **Included Routes (Added Routes)** field in the Tunnel VPN configuration. See Tunnel for iOS configuration field description.
- Multiple per-app VPN configurations are supported on a device. However, only one device-level VPN configuration is supported on a device.

Setting up MobileIron Tunnel

The following addresses the setup required for MobileIron Tunnel for iOS and contains the following:

In a MobileIron Tunnel for iOS deployment, AppConnect wrapping or SDK is not required and the client certificate is directly authenticated with the backend resource.

Before you set up MobileIron Tunnel

Before you set up MobileIron Tunnel for iOS devices, see the following:

Required components for deploying MobileIron Tunnel for iOS

The following components are required for a MobileIron Tunnel deployment:

- Standalone Sentry with AppTunnel enabled or MobileIron Access.
- MobileIron unified endpoint management (UEM) platform:
  - MobileIron Core

  OR
  - MobileIron Cloud
- iOS devices registered with a MobileIron UEM.
- MobileIron client for iOS:
  - Mobile@Work for MobileIron Core deployments

    OR
  - MobileIron Go for MobileIron Cloud deployments

OR

- MobileIron AppStation for MobileIron Cloud MAM-only deployments
  For information about deploying MobileIron AppStation for MAM-only, see *MobileIron AppStation for iOS Guide*.

For supported versions see the *MobileIron Tunnel for iOS Release Notes*.

Requirements for configuring MobileIron Tunnel for iOS

Ensure the following before configuring MobileIron Tunnel for iOS:

- If your deployment uses Standalone Sentry:
  - You have installed Standalone Sentry. See the *Standalone Sentry Installation Guide*.
  - Standalone Sentry is set up for AppTunnel using identity certificates for device authentication.
    For information about setting up a Standalone Sentry for AppTunnel, see *MobileIron Sentry Guide* for your MobileIron unified endpoint management (UEM) platform.
  - The Standalone Sentry IP address is publicly accessible.
  - The Standalone Sentry name is registered in DNS.
  - To tunnel IP traffic, ensure that you have created an IP_ANY service.
  - For documentation, see Standalone Sentry product documentation.
- Standalone Sentry is required for packet tunnel provider with per-app VPN.
- If your deployment uses MobileIron Access, ensure that MobileIron Access is set up. See the *MobileIron Access Guide* for information on how to set up MobileIron Access. For documentation, see MobileIron Access product documentation.
- The appropriate ports are open.
  See the *MobileIron Tunnel for iOS Release Notes*.

Recommendations for setting up MobileIron Tunnel for iOS

Review the following recommendations for setting up MobileIron Tunnel for iOS.

- Standalone Sentry
- UDP traffic

Standalone Sentry

MobileIron recommends that Standalone Sentry use a trusted CA certificate. If Standalone Sentry uses a self-signed certificate, you must do the following additional setup in MobileIron Core:

- In the **Services > Sentry** page, for the Standalone Sentry, click the **View Certificate** link. This makes the Standalone Sentry's certificate known to MobileIron Core.

- Follow the instructions in the *Using a Self-signed certificate with Standalone Sentry and MobileIron Tunnel* knowledge base article in the MobileIron Support and Knowledge Base portal at https://help.mobileiron.com/customer/articles/MI_Article/Using-a-self-signed-certificate-with-Standalone-Sentry-and-MobileIron-Tunnel.
  If the self-signed certificate is changed at any time, you must push the changed certificate to the device, otherwise there may be a disruption in service. Therefore, MobileIron **recommends using a certificate from a trusted certificate authority for the Standalone Sentry**.

UDP traffic

Standalone Sentry supports only limited types of UDP traffic, such as DNS traffic. Audio and video traffic through Standalone Sentry is not supported.

To limit the UDP traffic through Standalone Sentry, gather a list of destination UDP ports that should be tunneled through Tunnel VPN. All other UDP traffic is, therefore, not tunneled. Configure the **SplitUDPPortList** key-value pair to limit the UDP traffic through Tunnel.

Main tasks for configuring MobileIron Tunnel for iOS (Core)

Following are the main steps for configuring MobileIron Tunnel for iOS. These configuration tasks are performed in the MobileIron Core Admin Portal.

1. Configuring MobileIron Tunnel VPN in MobileIron Core
2. Applying the MobileIron Tunnel VPN setting to managed apps in MobileIron Core

Configuring MobileIron Tunnel VPN in MobileIron Core

Tunnel supports per-app and device-level VPN. Choose the appropriate configuration depending on whether you are creating a per-app VPN or a device-level VPN.

You can create multiple Tunnel configurations to push to a device. The VPN profiles pushed to a device are listed in **Settings > General > VPN**, and in **Settings > General > Device Management**. Depending on the app in use, iOS automatically switches to use the VPN profile applied to the app.

You can apply both per-app VPN and device-level VPN to a device. However, per-app VPN takes priority over device-level VPN. The device-level VPN is used for apps that are not associated with a per-app VPN.

Before you begin

- If you are configuring app proxy VPN, ensure that you have created a TCP AppTunnel service in Standalone Sentry.
- If you are configuring packet tunnel provider type, ensure that you have created an IP AppTunnel service in Standalone Sentry.

- For information on setting up a TCP or IP AppTunnel service see "Working with Standalone Sentry for AppTunnel" in the *Standalone Sentry Guide* for MobileIron Core. See MobileIron Sentry product documentation.
- If you are configuring MobileIron Tunnel for securing authentication traffic with MobileIron Access see the *MobileIron Access Guide*. See MobileIron Access product documentation.

IMPORTANT: MobileIron strongly recommends creating separate Tunnel VPN configurations for iOS and macOS. Using the same Tunnel VPN configuration for iOS and macOS may cause issues with how Tunnel operates and how traffic through Tunnel is handled.

Procedure

1. In the Admin Portal, go to **Policies & Configs > Configurations**.
2. Click **Add New > VPN**.
3. For **Connection Type**, select **MobileIron Tunnel**.
4. Add the necessary configurations.
5. Click **Save**.
6. If you created a device-level VPN configuration, apply the configuration to a label that contains iOS devices.
   The configuration is distributed to the devices in the label.

Next steps

Go to Applying the MobileIron Tunnel VPN setting to managed apps in MobileIron Core.

Related topics

- For a description of the configuration fields for MobileIron Tunnel (iOS) VPN, see Tunnel for iOS configuration field description.
- For a description of the key-value pairs, see Additional configurations using key-value pairs for MobileIron Tunnel.

Applying the MobileIron Tunnel VPN setting to managed apps in MobileIron Core

When you Add or Edit an app in the App Catalog, you have the option to select the per-app VPN setting to apply to the app. Select the per-app MobileIron Tunnel (iOS) VPN setting you created. This procedure is not needed for a device-level VPN configuration.

Procedure

1. In the Admin Portal, go to **Apps > App Catalog**.
2. Select **iOS** from the **Platform** list.
3. Click the **Add+**, or select an app and click the edit icon next to the app.
4. In the form, for **Per App VPN Settings**, select the per-app MobileIron Tunnel (iOS) VPN configuration you created.

Related topics

For more information about adding and editing apps for distribution, see the following sections in the *MobileIron Core Apps@Work Guide*:

- "Using the wizard to import iOS apps from the Apple App Store."
- "Using the wizard to add an in-house iOS or macOS app to the App Catalog."
- MobileIron Core product documentation.

MobileIron Tunnel for iOS distribution

Adding MobileIron Tunnel to the App Catalog makes the app available in the MobileIron app storefront.

MobileIron Tunnel is also available in the Apple App Store. The device user can download the app directly from the Apple App Store. Device users can download the app directly from the Apple AppStore at https://itunes.apple.com/us/app/mobileiron-tunnel/id1150035878?mt=8.

If you are a using a self-signed or an untrusted certificate for the Standalone Sentry, the certificate must be pushed to the device. The Standalone Sentry certificate is required on the device for Tunnel to authenticate the Standalone Sentry and establish a per-app VPN session. If the certificate is changed at any time, you must push the changed certificate to the device, otherwise there may be a disruption in service. Therefore, we **recommend using a certificate from a trusted certificate authority for the Standalone Sentry**.

If the certificate is changed at any time, you must push the changed certificate to the device, otherwise there may be a disruption in service. To push the Standalone Sentry certificate to the device, follow the instructions in the Using a Self-signed certificate with Standalone Sentry and MobileIron Tunnel knowledge base article on the MobileIron Community site.

Main tasks for configuring MobileIron Tunnel for iOS (Cloud)

You configure MobileIron Tunnel in MobileIron Cloud.

Before you begin

- If you are configuring app proxy VPN, ensure that you have created a **MobileIron Tunnel** service for **iOS / mac** with **Service Type TCP_ANY** in the Standalone Sentry profile.
- If you are configuring packet tunnel provider type VPN, ensure that you have created a **MobileIron Tunnel service** for **iOS / mac** with **Service Type IP_ANY** in the Standalone Sentry profile.
- For information on setting up Standalone Sentry with a MobileIron Tunnel service, see "Working with Standalone Sentry for AppTunnel" in the *Standalone Sentry Guide* for MobileIron Cloud.
- If you are configuring MobileIron Tunnel for securing authentication traffic with MobileIron Access see the *MobileIron Access Guide*.

Procedure: Overview of steps

1. Adding Tunnel for iOS to the app catalog in MobileIron Cloud
2. Adding a MobileIron Tunnel configuration in MobileIron Cloud

Adding Tunnel for iOS to the app catalog in MobileIron Cloud

Tunnel for iOS is available in the app catalog in MobileIron Cloud.

Procedure

1. In MobileIron Cloud, go to **Apps > App Catalog > +Add**.
2. In **Business Apps**, click **Tunnel (iOS 9+)**.
3. Make any updates as necessary and click **Next.**
   You can change the category and add a description.
4. Select an option for app delegation and click **Next**.
5. Choose a distribution option for the app and click **Next.**
6. Update the default **App Configurations** settings as necessary.
7. Click **Done**.

Next steps

Go to Adding a MobileIron Tunnel configuration in MobileIron Cloud.

Related topics

For more information about topics such as app delegation, see the *MobileIron Cloud Administrator Guide*.

Adding a MobileIron Tunnel configuration in MobileIron Cloud

You create the configuration for MobileIron Tunnel in **Configurations**. You can create multiple Tunnel configurations to push to a device. The VPN profiles pushed to a device are listed in **Settings > General > VPN**, and in **Settings > General > Device Management**. Depending on the app in use, iOS automatically switches to use the VPN profile applied to the app.

Tunnel supports per-app as well as device-level VPN. Choose the appropriate Tunnel configuration depending on whether you are creating a per-app VPN or a device-level VPN.

You can apply both per-app VPN and device-level VPN to a device. However, per-app VPN takes priority over device-level VPN. The device-level VPN is used for apps that are not associated with a per-app VPN.

Procedure

1. In MobileIron Cloud, go to **Configurations > +Add**.
2. Search for MobileIron Tunnel.

3. Click one of the following:
   - **MobileIron Tunnel**: Use this configuration to create a per-app VPN configuration for MobileIron Tunnel.
   - **MobileIron Tunnel (On Demand)**: Use this configuration to create a device-level VPN configuration for MobileIron Tunnel.

   The MobileIron Tunnel configuration page displays.
4. If you selected the **MobileIron Tunnel** configuration, click **iOS/macOS**.
   The configuration for Tunnel for iOS displays.
5. Add the necessary configurations and click **Next**.
6. Choose a distribution option for the configuration and click **Done.**
   The configuration is distributed to the subset of the devices to which the app is distributed. Select the same distribution option that you selected for the MobileIron Tunnel for iOS app.

Next steps

Go to Applying the MobileIron Tunnel VPN setting to managed apps in MobileIron Cloud.

Related topics

- For a description of the configuration fields for MobileIron Tunnel (iOS) VPN, see Tunnel for iOS configuration field description.
- For a description of the key-value pairs, see Additional configurations using key-value pairs for MobileIron Tunnel.

Applying the MobileIron Tunnel VPN setting to managed apps in MobileIron Cloud

When you Add or Edit an app in the App Catalog, you have the option to select the per-app VPN setting to apply to the app. For this workflow, select the MobileIron Tunnel (iOS) VPN setting you created. This procedure is not needed if you configured device-level VPN using **MobileIron Tunnel (On Demand)**.

Procedure

1. In **Apps > App Catalog**, add or edit an app.
2. In **App Configurations**, add the **Per App VPN** configuration.
3. Enter a name for the configuration.
4. Check **Enable Per-App VPN for this app.**
5. Select the Tunnel configuration to apply to the app.
6. Select a distribution option and click **Next**.
7. Click **Done**.

Related topics

For more information about adding and editing apps for distribution, see the following sections in the *MobileIron Cloud Administrator Guide*:

- “Adding an app from a public store.”
- “Adding an In-house app.”

Tunnel for iOS configuration field description

The following table provides field descriptions for the Tunnel configuration. There are some variations in field names between MobileIron Core and MobileIron Cloud.

TABLE 2. TUNNEL CONFIGURATION FIELD DESCRIPTION

Item	Description
Name	Enter a name for the MobileIron Tunnel VPN profile.
Description	Enter a description for the profile.
Connection Type (MobileIron Core)	Select **MobileIron Tunnel**. Only fields relevant to MobileIron Tunnel are displayed.
Choose OS to create Tunnel Configuration (MobileIron Cloud. Per-app VPN)	Select iOS/macOS.
Profile selection mode to use for this configuration (MobileIron Cloud)	Select one of the following: • **Sentry Profile Only**: Select if Tunnel traffic goes only through Standalone Sentry only. • **MobileIron Access Profile Only**: Select if Tunnel traffic goes to Access only. Only authentication traffic is tunneled to Access. This option is available only if a MobileIron Access deployment is set up. NOTE: If **MobileIron Access Profile Only** is configured with per-app VPN packet tunnel provider type, only authentication traffic is tunneled to Access. All other traffic is dropped. If **MobileIron Access Profile Only** is configured with device-level VPN packet tunnel provider type, only authentication traffic is tunneled to Access. All other traffic goes directly to the destination. • **MobileIron Sentry + Access Profile**: Select if Tunnel VPN supports both traffic to Access for authentication to enterprise cloud resources and through Standalone Sentry to on-premise enterprise resources. This option is available only if a MobileIron Access as a service deployment is set up.
Legacy App Support (iOS only)	Select one of the following: • **Enabled**: Select to enable per-app VPN with the MobileIron Tunnel

TABLE 2. TUNNEL CONFIGURATION FIELD DESCRIPTION (CONT.)

Item	Description
	Legacy app (versions of MobileIron Tunnel prior to 2.0) on all versions of iOS. • **Enabled for iOS 7 and 8**: Select to enable per-app VPN using the MobileIron Tunnel Legacy app for devices running iOS 7 and 8 only. This option enables the per-app VPN feature with MobileIron Tunnel 2.0 on devices running iOS 9 through the most recently released version as supported by MobileIron. The per-app VPN feature with MobileIron Tunnel requires a separate license and Sentry 5.0 through the most recently released version as supported by MobileIron. Ensure your organization has purchased the necessary license before enabling this feature. MobileIron Tunnel 2.0 through the most recently released version as supported by MobileIron is required for devices running iOS 9 through the most recently released version as supported by MobileIron.
VPN Sub Type (MobileIron Cloud)	(Optional) Overrides the bundle identifier for a customized MobileIron Tunnel app.
Enable MobileIron Access (MobileIron Core)	Select to enable authentication traffic through MobileIron Access. The option is available only if Access as a service is set up with MobileIron Core. For information about how to set up Access as a service with MobileIron Core, see the *MobileIron Access Guide*.
Provider Type (In MobileIron Cloud, this field is available only in the MobileIron Tunnel configuration for per-app VPN.	**app-proxy**: This is the default setting. Use this setting for TCP tunneling only. **packet-tunnel**: Select to allow Tunnel to also handle IP traffic. NOTE: Device-level VPN automatically uses the packet tunnel provider type.
Per-app VPN (MobileIron Core)	The options are available if **Provider Type** is **packet-tunnel**. Otherwise, the options are grayed out. Device-level VPN is not available for app proxy tunnel. **Yes**: This is the default setting. Connectivity is established for an app, rather than the device. **No**: Select to establish connectivity for the device, rather than just an app.
Sentry (Profile)	**Core:** Select the Standalone Sentry on which you created the tunnel service. **Cloud:** Select the Standalone Sentry profile on which you created the Tunnel for iOS service. The field is not available if the profile mode is **MobileIron Access Profile Only.**

TABLE 2. TUNNEL CONFIGURATION FIELD DESCRIPTION (CONT.)

Item	Description
Sentry Service	**Core:** Select the TCP or IP service that the Safari domain or managed app will use. If you are configuring packet tunnel provider type, select the IP service you created for Tunnel. If you are configuring app proxy, select the TCP service you created for Tunnel. **Cloud:** Select the Tunnel for iOS service. The field is not available if the profile mode is **MobileIron Access Profile Only.** Only TCP services are available for selection if the provider type is app proxy. Only IP services are available for selection if the provider type is packet tunnel.
SCEP Identity (MobileIron Cloud)	Select the **Identity Certificate** configuration you created for Tunnel. The Identity Certificate is automatically selected if **Sentry Profile Only** or **MobileIron Sentry + Access Profile** is enabled.
Debug Info Recipient (MobileIron Cloud)	Enter an email address to forward the debug information.
Identity Certificate (MobileIron Core)	Select the certificate setting you created. If you are using user-provided certificates, select the user provided certificate you created for MobileIron Tunnel.
On Demand Rules (iOS 9 and later; macOS 10.13 and later) VPN on-demand rules are applied when the device's primary network interface changes, for example, when the device switches to a different Wi-Fi network. Devices will drop the Tunnel VPN connection if an enterprise Wi-Fi is detected. If the network is not a Wi-Fi network or if its SSID does not appear in the list, the device will continue to use Tunnel VPN. Note The Following: • A matching rule is not required. The Default Rule is applied if a matching rule is not defined. • If you select Evaluate Connection, a matching rule is not required. • You can create up to 10 On Demand matching rules. • For each matching rule you can create up to 50 Type and Value pairs. IMPORTANT: An Ethernet on-demand rule is only applicable to macOS devices. If the rule is pushed to iOS device, the rule may cause issues with Tunnel behavior and how traffic through Tunnel is handled. Therefore, MobileIron strongly recommends using separate Tunnel VPN configurations for iOS and macOS.	
Add +	Click to add a new On Demand matching rule.

TABLE 2. TUNNEL CONFIGURATION FIELD DESCRIPTION (CONT.)

Item	Description
On Demand Action	Select one of the following actions to apply to the matching rule: • Connect • Disconnect
Matching Rules For each On Demand matching rule to which the action is applied enter the type and value pair.	
Add +	Click to add a new On Demand matching rule. A dialog box appears.
Type	Select the following key type: • **SSID**
Value	Enter a list of SSIDs to match the enterprise Wi-Fi. If the network is not a Wi-Fi network or if its SSID does not appear in the list, the match will fail. TIP: To add multiple SSIDs, create a separate SSID Type-Value pair for each SSID.
Description	Enter additional information about this matching rule.
OK	Click to add the On Demand Action and the associated Matching Rules.
Default Rule The default rule (action) is applied to a connection that does not match any of the matching rules.	
On Demand Action	From the drop down list, select **Connect**.
Safari Domains The device user can access servers ending with these domains in Safari. A MobileIron Tunnel configuration is only applied to a managed app. Therefore, a managed app with the MobileIron Tunnel configuration must be installed on the device for the device user to access the domains using per-app VPN. Note The Following: • If the device resolves the destination domain, then Tunnel is not launched. • If the Safari domains use Kerberos authentication, you must also do the setup described in Setting up single sign-on with Kerberos.	
Safari Domain	Enter a domain name. Only alphanumeric characters and periods (.) are supported.

TABLE 2. TUNNEL CONFIGURATION FIELD DESCRIPTION (CONT.)

Item	Description
Description	Enter a description for the domain.
Add New	Click to add a domain.
Calendar Domains (iOS 13 and later; macOS 10.15 and later) A Tunnel VPN connection is automatically established for these domains. Only available for per-app VPN.	
Calendar Domain	Enter a domain name. Only alphanumeric characters and periods (.) are supported.
Description	Enter a description for the domain.
Add New	Click to add a domain.
Contact Domains (iOS 13 and later; macOS 10.15 and later) A Tunnel VPN connection is automatically established for these domains. Only available for per-app VPN.	
Contact Domain	Enter a domain name. Only alphanumeric characters and periods (.) are supported.
Description	Enter a description for the domain.
Add New	Click to add a domain.
Mail Domains (iOS 13 and later; macOS 10.15 and later) A Tunnel VPN connection is automatically established for these domains. Only available for per-app VPN.	
Mail Domain	Enter a domain name. Only alphanumeric characters and periods (.) are supported.
Description	Enter a description for the domain.
Add New	Click to add a domain.
Included Routes (Added Routes) Only available for device-level VPN. Configured routes are set to the TUN interface. If routes are not configured, Tunnel uses 0.0.0.0/0. Enter list of IPv4 ranges in CIDR format. For multiple values, enter a semicolon separated list.	
DNS Resolver IPs	

TABLE 2. TUNNEL CONFIGURATION FIELD DESCRIPTION (CONT.)

Item	Description
	Only for packet tunnel provider type. Enter a domain name server (DNS) to resolve the IP address. IPv4 only. For multiple values, enter a semicolon separated list. Ensure that the DNS is routable if the default route in not used. If DNS is not configured, the Sentry DNS is used.
DNS Search Domain List	Only for packet tunnel provider type. Enter DNS search domains for resolving the domain names. For multiple values, enter a semicolon separated list.
Match Domain List	Only for packet tunnel provider type. Enter domains for the VPN DNS to resolve. For multiple values, enter a semicolon separated list.
Custom Data	Enter Key Value pair to configure the MobileIron Tunnel VPN disconnect, debug, and timeout behavior. See Additional configurations using key-value pairs for MobileIron Tunnel.

Setting up single sign-on with Kerberos

The following addresses how to set up single sign-on with Kerberos on iOS devices and contains the following:

About the setup for single sign-on with Kerberos

The setup described allows Safari and managed apps that support Kerberos to securely access an internal resource using SSO when the device is outside the corporate network. The Key Distribution Center (KDC) sits inside the corporate network. A major architectural change was introduced in Tunnel 2.0 allowing Tunnel to use Network Extension framework introduced in iOS 9.0. Due to the support for Network Extension framework, use of Standalone Sentry as a KKDCP is no longer required for SSO with Kerberos.

Related topics

- Main tasks for configuring single sign-on with Kerberos (Core or Cloud)
- SSO with Kerberos configuration field description

Authentication workflow for single sign-on with Kerberos

The following describes the authentication flow for single sign-on with Kerberos:

1. The managed app or Safari domain initiates a connection with the backend resource through the TCP tunnel configured on the Standalone Sentry. The managed app must support Kerberos.
2. The backend resource, via the Standalone Sentry, returns a request to authenticate and the KDC realm information to the device.
3. The device sends an SRV Kerberos DNS query to Tunnel. Tunnel matches the requested domains to the domains configured in the SRV key-value pair. The kerberos DNS query is resolved to the host name (target) configured in the SRV key-value pair.

   NOTE: SRV configuration is not required if packet tunnel provider is configured.
4. The device communicates with the KDC server (target) through Tunnel and a ticket is returned to the device. The ticket is stored on the device.

5. The device presents the ticket to the backend resource for authentication.
6. The device uses the ticket to authenticate to backend resources configured in the single sign-on setting.

Main tasks for configuring single sign-on with Kerberos (Core or Cloud)

Following are the main steps for configuring single sign-on with Kerberos:

1. Configuring SRV (Core or Cloud)
2. Configuring single sign-on (Core)
   OR
   Configuring single sign-on (Cloud)

Before you begin

- Set up per-app VPN with MobileIron Tunnel as described in Setting up MobileIron Tunnel. Apply the MobileIron Tunnel VPN setting to the managed apps that will use single sign-on with Kerberos authentication. The managed app must support Kerberos.
- If you want an app to use a certificate to authenticate the device user to a backend resource when the Kerberos ticket has expired, create a certificate enrollment setting. You will reference the certificate in the single sign-on setting.
- If you do not provide an identity certificate, the device user is prompted to enter a user ID and password when the Kerberos ticket has expired.
- Ensure that devices have access to a Kerberos Domain Controller (KDC) and the backend resources that you specify in the single sign-on setting.

Configuring SRV (Core or Cloud)

Configuring SRV is not required if you configure packet tunnel provider type in the Tunnel VPN configuration.

The SRV feature resolves Kerberos DNS requests from devices in environments with different internal and public Kerberos domain controller (KDC) DNS domains. In order to resolve the SRV query and determine the KDC that handles the authentication requests for the backend server, you configure key-value pairs in the Tunnel VPN configuration for iOS. This feature replaces the need to create an SRV record in your DNS.

NOTE: If you are configuring split tunneling in MobileIron Access, ensure that domain name in the split tunneling configuration matches exactly the SRV record in the MobileIron Tunnel for iOS configuration.

Procedure

1. In your MobileIron Enterprise Mobility Management (UEM) platform, select the MobileIron Tunnel configuration for iOS to edit.

- In the MobileIron Core Admin Portal, go to **Policies & Configs > Configurations**.
- In MobileIron Cloud, go to **Configurations**.

2. In **Custom Data**, add the following key-value pair:
   - Key: SRV_kerberos._tcp.*DnsDomainName*
   - Value: SRV *Priority Weight Port Target*

   To configure multiple values for the same domain name, add a new row. Enter the same key with a trailing *#n*, where *n* is an integer, to the key. Add a trailing *#n* to the first record as well. Ensure that there are no spaces between the key and *#n.* If there are multiple entries, a KDC is contacted based on priority and weight.

   Example

Key	Value
SRV_kerberos._tcp.example.com#1	SRV 0 100 88 kdc.example.com
SRV_kerberos._tcp.example.com#2	SRV 0 100 88 kdc2.example.com

3. In **Safari Domains**, add the root domain.
   Configuring the root domain allows all traffic, including Kerberos traffic, to go through Tunnel.

   Example : example.com

   Alternately, if you do not want to configure the root domain, add the following to Safari Domains:
   - the backend resource being accessed.
   - _kerberos._tcp.*DnsDomainName:* configured in **Custom Data**.
     The realm name in the Kerberos DNS query is case sensitive. Therefore, the *DnsDomainName* must be in upper case.
   - *Target:* configured in **Custom Data**.

   Configuring the domains ensures that traffic required to resolve the Kerberos DNS request goes through MobileIron Tunnel.

   Example
   - sharepoint.example.com
   - _kerberos._tcp.EXAMPLE.COM
   - kdc.example.com
4. Click **Save**.

Next steps

- Configuring single sign-on (Core)
  OR
- Configuring single sign-on (Cloud)

Related topics

- For more information about the values for the SRV key, see https://www.ietf.org/rfc/rfc2782.txt.
- For more information about the SRV key-value pair, see Additional configurations using key-value pairs for MobileIron Tunnel.

Configuring single sign-on (Core)

Specify the URLs or resources that the device user can access using single sign-on (SSO).

Note The Following:

- For Realm, enter $REALM$.
- Create a separate Single sign-on configuration for each realm.

Procedure

1. In the Admin Portal, go to **Policies & Configs > Configurations**.
2. From the **Add New** drop-down menu, go to **iOS and OS X > Single Sign-On Account**.
   The **New Single Sign-On (SSO) Configuration** screen displays.
3. Complete the form.
4. Click **Save**.
5. In the Configurations page, select the configuration.
6. Click **More Actions > Apply To Label**.
7. Select a label to apply, and click **Apply**.

Related topics

- For a description of the fields in **New Single Sign-On (SSO) Configuration**, see SSO with Kerberos configuration field description.

Configuring single sign-on (Cloud)

Specify the URLs or resources that the device user can access using SSO.

NOTE: Create a separate Single sign-on configuration for each realm.

Procedure

1. In MobileIron Cloud, go to **Configurations > +Add**.
2. Search for single sign-on.
3. Click the **Single Sign-On Account** configuration
   The **Create Single Sign-On Account Configuration** page displays.
4. Add the necessary configurations and click **Next**.

5. Choose a distribution option for the configuration and click **Done.**
   The configuration is distributed to the devices in distribution option. Select the same distribution option that you selected for the MobileIron Tunnel for iOS app.

Related topics

- For a description of the fields in **New Single Sign-On (SSO) Configuration**, see SSO with Kerberos configuration field description.

SSO with Kerberos configuration field description

The following table provides field descriptions for the single sign-on configuration. There are some variations in field names between MobileIron Core and MobileIron Cloud.

TABLE 3. TUNNEL CONFIGURATION FIELD DESCRIPTION

Field	Description
Name	Enter a name for this configuration.
Description	Enter additional information that describes this configuration.
User Name	(Required) Enter the Kerberos user name. **Core:** You can also specify the variable $USERID$. **Cloud:** You can also specify the variable ${samaccountname}
Realm	(Required) **Core:** The default is $Realm$. This is the only valid variable. $Realm$ is supported for LDAP users only. The realm is calculated by extracting the base DN (e.g. DC=auto, DC=MyCompany, DC=com) and converting to a domain. Example: AUTO.MYCOMPANY.COM. **Cloud:** Enter a domain name. Example: AUTO.MYCOMPANY.COM.
Identity Certificate (MobileIron Core)	(Optional) Select a certificate enrollment setting from the drop-down list to specify an identity certificate. An app uses this identity certificate to authenticate the device user to the KDC server. After the user is authenticated, the KDC server issues a ticket to the user. If the Kerberos ticket has expired, it is silently renewed after the user is authenticated. If you do not provide an identity certificate, the device user is prompted to enter a user ID and password when the Kerberos ticket has expired.
Certificate (MobileIron Cloud)	(Optional) Select the certificate to use. An app uses this identity certificate to authenticate the device user to the KDC

TABLE 3. TUNNEL CONFIGURATION FIELD DESCRIPTION (CONT.)

Field	Description
	server. After the user is authenticated, the KDC server issues a ticket to the user. If the Kerberos ticket has expired, it is silently renewed after the user is authenticated. If you do not provide an identity certificate, the device user is prompted to enter a user ID and password when the Kerberos ticket has expired.
URL Prefix Matches (Required) Add the URLs or resources that the device user can access using SSO. At least one URL is required. If a bundle ID (application ID) is configured, SSO is enabled for the specified apps only when the apps access the URLs that match the configured URL prefixes. If a bundle ID (application ID) is not configured, SSO is applicable to all apps that support SSO when they access the URLs that match the configured URL prefixes.	
+	Click to add an URL.
URL	Enter the URL that the user can access using SSO. Note The Following: • The website or resource must support Kerberos based authentication. • Entries must begin with the URL scheme: `HTTP://` or `HTTPS://` • A simple string match is performed. For example, `http://www.example.com/` does not match `http://www.example.com:80/` • If an entry does not end with the character /, a / is appended to the entry. • For devices running iOS 9 through the most recently released version as supported by MobileIron, you can use a single wildcard * to specify all matching values. For example, `http://*.example.com` matches both `http://store.example.com/` and `http://www.example.com/` However, a wildcard at the end of the URL will not work. Example of incorrect url: http://www.example.com/* • The entries `http://.com` and `https://.com` match all HTTP and HTTPS URLs, respectively.
Description	Enter additional information describing this resource.
-	Click to delete the URL.
Application Identifier Matches (Optional) Add the apps that the device user can use to access the URLs or resources listed in URL Prefix Matches without having to enter their enterprise credentials.	

TABLE 3. TUNNEL CONFIGURATION FIELD DESCRIPTION (CONT.)

Field	Description
You can add up to twenty bundle IDs (application IDs) per configuration. If no apps are entered, the device user can access the URLs or resources from any app without having to enter their enterprise credentials.	
+	Click to add an app.
BundleID	Enter an exact or partial bundle ID (application ID) for the app. Use the following rules for formatting an entry: • The string you specify can be an exact match with a bundle ID. Example: `com.mycompany.myapp` • Partial matches are supported. • The string you specify can match a prefix of a bundle ID by using exactly one * wildcard character. The * appears after a period character, and at the end of the string. Example: `com.mycompany.*` matches any app for which the bundle ID begins with `com.mycompany.`
Description	Enter additional information describing the app.
_	Click to delete the entry.

Additional configurations using key-value pairs for MobileIron Tunnel

Key-value pairs are used to customize Tunnel for iOS app behavior. These key-value pairs define app behavior such as idle timeout, email address for sending debug information, and level of log detail that is collected.

The following table provides the key-value pairs for customizing Tunnel for iOS.

TABLE 4. KEY-VALUE PAIRS FOR TUNNEL FOR IOS

Key	Value
Manage Tunnel timeout	
disconnectTimeoutInSeconds (MobileIron Core)	Enter 0 or a number between 5 - 18000. If the value is 0, then MobileIron Tunnel VPN never disconnects itself. You have to manually disconnect the VPN in the MobileIron Tunnel. If the value is > 0, the MobileIron Tunnel VPN is disconnected after number entered. If this key-value pair is not configured, the default is 60 seconds.
TcpIdleTmoMs	Enter any integer between 5000 - 18000000. The timeout is measured in milliseconds. Configuring idle timeout allows you to control the idle session timeout for the TCP connection between the app and the backend server. You may want to configure idle timeout if the backend server takes more than 60 seconds to respond to a request. The default idle timeout with Standalone Sentry for per-app VPN if the key-value pair is not configured: 60 seconds. NOTE: For packet tunnel, MobileIron recommends setting the idle timeout equal to or larger than the idle timeout for the enterprise server being accessed. If you do not know the idle timeout for the server, set the value to 3600000.
Troubleshooting	
debugInfoRecipient (Available as field value in MobileIron Cloud)	Enter an email address to forward the debug information.
LogLevel	Enter debug <*Log Level*> Use one of the following log level options. The options are listed from the

TABLE 4. KEY-VALUE PAIRS FOR TUNNEL FOR IOS (CONT.)

Key	Value
	least to the most verbose level. • error: Captures error logs if the Tunnel app errors out while performing an action. • warning: Captures warning messages logged if there is missing or incorrect information that might cause an error. This log level is rarely used. • info: Captures informational level details such as, log prints inputs, metadata, parameter values. • debug: Captures debug level information such as, actions, operations, values of critical data, and information that is helpful in debugging. • session: Captures everything that occurs during a tunnel session. • packet: Captures packet level information, such as, length in bytes. Used for troubleshooting DNS queries and responses to and from Tunnel. Default if the key-value pair is not configured: info
UseSecureEMail	Enter `true`. Tunnel uses Email+ to send debug logs. If the key-value pair is not configured, Tunnel uses the native iOS email client to send debug logs.
SendDeviceID	Enter `true`. Tunnel provides the device ID to MobileIron Access. The device ID is reported on MobileIron Access in Reports > Errors. The key-value pair is useful in identifying devices that encounter connection errors when authenticating through Access. Default if the key-value pair is not configured: false.
DNS and network	
PublicDNS	Enter a space-separated list of DNS servers that are accessible from the device. Each DNS entry is -separated by a space. IPv4 and IPv6 addresses are supported.

TABLE 4. KEY-VALUE PAIRS FOR TUNNEL FOR IOS (CONT.)

Key	Value
	Since (managed) apps have access to the DNS servers configured on the device, this KVP is needed only in rare cases. **Example** 8.8.8.8 8.8.8.1
IPv6NetworkPrefix	IPv6 ULA network prefix to use for internal NAT table.
DNS query for SRV record (for SSO with Kerberos)	
SRV_kerberos._ tcp.*DnsDomainName* Where *DnsDomainName* is the internal domain name of the KDC server. Example: SRV_kerberos._tcp.example.com	Enter SRV *Priority Weight Port Target* Where: • Priority is the priority of the server. • Weight is the load-balancing mechanism that is used when selecting a target • Port is the port number the server is listening. • Target is the fully qualified domain name (FQDN) of the KDC server. **Example** SRV 0 100 88 kdc.example.com SRV record derived from the key-value pair: _kerberos._tcp.example.com. SRV 0 100 88 kdc.example.com. IMPORTANT: Ensure that the domain configured for *DnsDomainName* and for *Target* is also configured in **Safari Domains** in the Tunnel VPN configuration. Configuring the domains in **Safari Domains** ensures that the traffic goes through Tunnel.
Certificates	
DisablePinning	**false**: Default, if the key-value pair is not configured. Certificate pinning is enabled. **true**: Certificate pinning is disabled. Disabling certificate pinning is not recommended for security reasons. NOTE: The Standalone Sentry server certificate is automatically pushed to the device.
Packet-tunnel	

TABLE 4. KEY-VALUE PAIRS FOR TUNNEL FOR IOS (CONT.)

Key	Value
IPRoutes	IP routes of the iOS or macOS device VPN. Enter list separated by semicolon. The default value if the key-value is not configured is 0.0.0.0/0 **Example** 10.0.0.0/8;172.16.0.0/16
ExcRoutes	IP routes that will be excluded from **IPRoutes.** **Example** 10.10.10.10/32.
SplitUDPPortList	Enter list of UDP ports to send through Tunnel VPN. All other UDP packets are sent directly to destination. If the KVP is not configured, all UDP packets are sent through Tunnel VPN. **Example** 53;161-162;200-1024 NOTE: Standalone Sentry supports only limited types of UDP traffic, such as DNS traffic. Audio and video traffic through Standalone Sentry is not supported. Therefore, MobileIron recommends configuring SplitUDPPortList to manage UDP traffic.
MTU	Tunnel MTU. The default value if the key-value is not configured is 1400.
TunIP	IP address of the VPN network interface. Configure only if customer network is in the same range. **Example** 192.168.13.10
AtpProbeIdleSec	Sets the minimum idle time, in seconds, after which probe packets are sent out with outbound Tunnel traffic. If Tunnel does not receive a response for at least one of the probes sent, the existing connection is dropped and a new connection is established with the server. The minimum idle time is based on the last inbound response received by

TABLE 4. KEY-VALUE PAIRS FOR TUNNEL FOR IOS (CONT.)

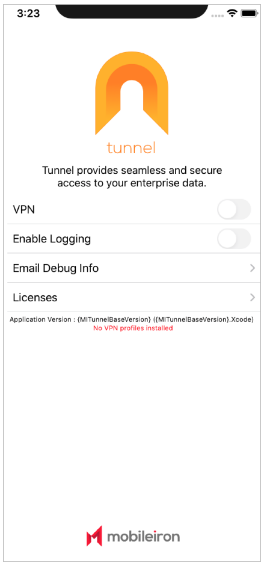
Key	Value
	Tunnel. For example, if the value is 60 seconds, if Tunnel does not receive any inbound traffic for 60 seconds, probe packets are sent with the next outbound Tunnel traffic. Default value if the key-value pair is not configured: 60 seconds
AtpProbeIntervalSec	Sets the interval, in seconds, between probe packets sent after the minimum idle time specified in AtpProbeIdleSec. Default value if the key-value pair is not configured: 1 second
AtpProbeCount	Sets the total count of the probe packets sent after the minimum idle time specified in AtpProbeIdleSec. Default value if the key-value pair is not configured: 5
App proxy	
DirectLocalhost	Enter `true`. Configure if using app proxy Tunnel. The key-value pair is required for Tunnel to handle app proxy localhost traffic from apps. true: If an app uses localhost, ::1, or 127.0.0.1, the localhost app proxy (TCP) traffic is redirected to the device itself.

What users see in MobileIron Tunnel for iOS

MobileIron Tunnel installation

MobileIron Tunnel is available in the Apple AppStore. You can install the app directly from the Apple AppStore.

FIGURE 1. MOBILEIRON TUNNEL



Note The Following:

- If your device does not have the Tunnel VPN profile, you will see the 'No VPN profiles installed' message. After the Tunnel VPN profile is pushed to the device, the message goes away.
- If the Tunnel VPN profile is installed on your device, when you tap a supported managed app and the app attempts to connect to a backend resource, the VPN connection is automatically turned on, and the app can securely connect to the enterprise resource. In some cases, if the VPN connection is not turned on, you can manually turn on VPN in the Tunnel app. Your IT administrator will tell you if you need to turn on VPN in the Tunnel app. You have to turn on VPN only once for the device.

Emailing debug log information

IT administrators may require Tunnel debug and log data for troubleshooting purposes. Tunnel provides device users the option to email the Tunnel debug and log file to the IT administrator. Depending on the Tunnel setup, users can either send debug logs using the native iOS email app or using MobileIron Email+.

By default Tunnel uses the native iOS email app.

Before you begin

Do one of the following:

- Ensure you have an email account set up in the native iOS email app your device.
  OR
- If your administrator has configured Tunnel to use Email+, ensure that Email+ is deployed on your device.
  See the description for `UseSecureEMail` in the table in Additional configurations using key-value pairs for MobileIron Tunnel

Procedure

1. In the Tunnel app, turn on **Enable Logging**.
2. Tap **Email Debug Info**.
   NOTE: Log information is included only if there has been activity using the Tunnel app.
3. Enter the email address provided by your administrator.
4. Tap **Send**.

Log information continues to be collected till **Enable Logging** is turned off. Therefore, after collecting the logs for debugging, turn off **Enable Logging** to stop collecting app logs. If the UseSecureEmail key is configured as true in the Tunnel VPN configuration, then Email+ must be set up on the device. If UseSecureEmail is not configured, then the iOS native email app must be set up. Otherwise, users see an error message and logs cannot be emailed..

If the device has multiple Tunnel VPN profiles,

- the email body includes the number of profiles and lists each profile with its app list and Safari domains.
- if Tunnel is set up to use Email+, and any one of the profiles is set up to use Email+, then all profiles automatically use Email+.