ivanti

# Guide

ivanti

# CONTENTS


1. **Ivanti Tunnel 4.12.0 - 4.13.0 for macOS Guide** ..... **3**
2. **New features summary** ..... **4**
3. **About Ivanti Tunnel for macOS** ..... **5**
   - 3.1. Overview of Ivanti Tunnel for macOS ..... 6
   - 3.2. About Ivanti Tunnel configuration ..... 7
   - 3.3. Deployment use cases with Ivanti Tunnel for macOS ..... 8
4. **Setting up Ivanti Tunnel for macOS** ..... **9**
   - 4.1. Before you set up Ivanti Tunnel for macOS ..... 10
   - 4.2. Main tasks for configuring Ivanti Tunnel for macOS (Ivanti EPMM) ..... 12
   - 4.3. Main tasks for configuring Ivanti Tunnel for macOS (Ivanti Neurons for MDM) ..... 14
   - 4.4. Ivanti Tunnel VPN configuration field description ..... 16
5. **Additional configurations using key-value pairs for Ivanti Tunnel** ..... **22**
   - 5.1. Key-value pairs for Ivanti Tunnel for macOS ..... 23
6. **What users see in Ivanti Tunnel for macOS** ..... **26**
   - 6.1. Ivanti Tunnel installation on macOS ..... 27
   - 6.2. Ivanti Tunnel icon on macOS devices ..... 28
   - 6.3. Exiting Ivanti Tunnel on macOS devices ..... 29
   - 6.4. Activating Ivanti Tunnel on macOS devices ..... 31
   - 6.5. Emailing debug log information ..... 32
   - 6.6. Sharing debug logs ..... 33

# 1. Ivanti Tunnel 4.12.0 - 4.13.0 for macOS Guide

## July 2026

These are cumulative release notes. If a page does not list a release, then there is no new associated information for that release.

# 2. New features summary

These are cumulative release notes. If a release does not appear in this section, then there were no associated new features and enhancements.

# 3. About Ivanti Tunnel for macOS

The following provide an overview of Ivanti Tunnel for macOS devices:

# 3.1. Overview of Ivanti Tunnel for macOS

Ivanti Tunnel enables VPN capability on macOS devices. Ivanti Tunnel interacts with the Unified Endpoint Management (UEM) platform, Standalone Sentry, and Access to secure access to enterprise resources from outside the enterprise network. The enterprise resource can be on premise or in the cloud. The UEM platforms are: Ivanti EPMM and Ivanti Neurons for MDM.

# 3.2. About Ivanti Tunnel configuration

Configurations for Ivanti Tunnel are created in a Unified Endpoint Management (UEM) platform. Ivanti Tunnel receives the configuration from the UEM client. The Ivanti client for Ivanti EPMM is Mobile@Work, and the client for Ivanti Neurons for MDM is Go.

# 3.3. Deployment use cases with Ivanti Tunnel for macOS

Ivanti Tunnel enables native per app VPN on macOS devices. Ivanti Tunnel is part of the following Ivanti deployments for securing access to enterprise resources:

- UEM and Standalone Sentry
- UEM and 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.

# 4. Setting up Ivanti Tunnel for macOS

The following addresses the setup required for app VPN using Ivanti Tunnel for macOS and contains the following:

The bundle IDs for Ivanti Tunnel for iOS and for macOS are different. Therefore, create a separate Ivanti Tunnel configuration for macOS. DO NOT use the same configuration for iOS and macOS.

# 4.1. Before you set up Ivanti Tunnel for macOS

Before you set up Ivanti Tunnel for macOS, see the following:

- Required components for deploying Ivanti Tunnel for macOS
- Requirements for setting up Ivanti Tunnel for macOS
- Recommendations for setting up Ivanti Tunnel for macOS (Ivanti EPMM)
- Limitations for Ivanti Tunnel for macOS

## Required components for deploying Ivanti Tunnel for macOS

The following components are required for an Ivanti Tunnel for macOS deployment:

- Standalone Sentry with AppTunnel enabled or Access.
- Unified Endpoint Management (UEM) platform:
  - Ivanti EPMM

or

  - Ivanti Neurons for MDM
- macOS devices registered with a UEM.

Ivanti EPMM: For information about registering macOS devices, see "Registering iOS and macOS devices through the web" in the *Ivanti EPMM Device Management Guide* for iOS and macOS Devices.

Ivanti Neurons for MDM: For information about registering macOS devices on Ivanti Neurons for MDM, see the *Ivanti Neurons for MDM Guide* or the Help on Ivanti Neurons for MDM.

For supported versions see the *Ivanti Tunnel for macOS Release Notes*.

## Requirements for setting up Ivanti Tunnel for macOS

The following are requirements for setting up Ivanti Tunnel for macOS:

- 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 Standalone Sentry for Ivanti EPMM Guide for 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 Access, ensure that Access is set up. See the *Access Guide* for information on how to set up Access. For documentation, see Access product documentation.

- The appropriate ports are open.
  See the *Ivanti Tunnel for macOS Release Notes*.

## Recommendations for setting up Ivanti Tunnel for macOS (Ivanti EPMM)

Standalone Sentry: Ivanti 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 Ivanti EPMM:

- In the Services > Sentry page, for the Standalone Sentry, click the View Certificate link. This makes the Standalone Sentry's certificate known to Ivanti EPMM.
- Follow the instructions in the **Using a Self-signed certificate with Standalone Sentry and Ivanti Tunnel** knowledge base article.
  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, Ivanti recommends using a certificate from a trusted certificate authority for the Standalone Sentry.

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

## Limitations for Ivanti Tunnel for macOS

Ivanti Tunnel for macOS has the following limitations:

- Single sign on with Kerberos is not supported.

# 4.2. Main tasks for configuring Ivanti Tunnel for macOS (Ivanti EPMM)

Following are the main steps for configuring Ivanti Tunnel for macOS:

1. **Configuring Ivanti Tunnel VPN (Ivanti EPMM)**
2. **Applying the Ivanti Tunnel VPN setting to managed apps (Ivanti EPMM)**
3. **Distribute Ivanti Tunnel for macOS as a VPP app (Ivanti EPMM)**

## Configuring Ivanti Tunnel VPN (Ivanti EPMM)

Ivanti 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, macOS 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 Ivanti EPMM.
- If you are configuring Ivanti Tunnel for securing authentication traffic with Access see the *Access Guide*.

Ivanti 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 Ivanti Tunnel.
4. Add the necessary configurations.
5. Click Save.
6. Apply the configuration to a label containing the macOS devices.

Next steps

Go to **Applying the Ivanti Tunnel VPN setting to managed apps (Ivanti EPMM)**.

Related topics

- For a description of the configuration fields for Ivanti Tunnel VPN, see **Ivanti Tunnel VPN configuration field description**.
- For a description of the key-value pairs, see **Key-value pairs for Ivanti Tunnel for macOS**.

## Applying the Ivanti Tunnel VPN setting to managed apps (Ivanti EPMM)

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 Ivanti Tunnel VPN setting for macOS that you created.

Before you begin

Ensure that the apps to which you will apply the Ivanti Tunnel VPN setting are available in the App Catalog on Ivanti EPMM. See the *Ivanti EPMM Apps@Work Guide* for your release for more information.

> macOS apps can be deployed either as VPP apps or as in-house apps. Ensure that the VPP apps are assigned to a VPP label and a macOS label, and in-house apps are assigned to a macOS label.

**Procedure**

1. In the Admin Portal, go to Apps > App Catalog.
2. Select macOS from the Platform list.
3. Select an app and click the edit icon next to the app.
4. In the form, for Per App VPN Settings, select the Ivanti Tunnel (macOS) VPN you created.

Related topics

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

- "Populating the iOS and macOS App Catalogs."
- "Using the wizard to add an in-house iOS or macOS app to the App Catalog."
- "Using the Apple Volume Purchase Program (VPP)."
- **Ivanti EPMM product documentation**.

## Distribute Ivanti Tunnel for macOS as a VPP app (Ivanti EPMM)

Ivanti Tunnel is available in the Mac App Store. Device users can download the app directly from the Mac App Store.

Ivanti Tunnel can also be distributed as a Volume Purchase Program (VPP) app from Ivanti EPMM. Apple provides VPP to facilitate app purchase and distribution within an organization. The App Store Volume Purchase Program (VPP) allows participating organizations to purchase iOS and macOS apps in volume and distribute the apps to their users.

While Apple supports user-based licensing for macOS VPP apps, currently there is an Apple issue with the installation of user-based licensed VPP apps through MDM. As a result, Ivanti does not recommend applying user-based licenses to macOS VPP apps.

For information on how to distribute Ivanti Tunnel for macOS as a VPP app, see "Using the Apple Volume Purchase Program (VPP) in the *Ivanti EPMM Apps@Work Guide.*

> For countries for which a VPP program is not available, device users can download and install Ivanti Tunnel directly from the Mac App Store.

# 4.3. Main tasks for configuring Ivanti Tunnel for macOS (Ivanti Neurons for MDM)

Following are the main steps for configuring Ivanti Tunnel for macOS:

1. Adding Ivanti Tunnel configuration in Ivanti Neurons for MDM
2. Applying the Ivanti Tunnel VPN setting to managed apps (Ivanti Neurons for MDM)
3. Distribute Ivanti Tunnel for macOS as a VPP app (Ivanti Neurons for MDM)

## Adding Ivanti Tunnel configuration in Ivanti Neurons for MDM

You create the configuration for Tunnel in Configurations. Ivanti 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.

Before you begin

- If you are configuring per app VPN, create Ivanti Tunnel service for macOS in Standalone Sentry. For information on setting up Standalone Sentry with an Ivanti Tunnel service, see "Working with Standalone Sentry for AppTunnel" in the *Standalone Sentry Guide* for Ivanti Neurons for MDM.
- If you are configuring Ivanti Tunnel for securing authentication traffic with Access see the *Access Guide*.

Ivanti 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 Ivanti Neurons for MDM, go to Configurations > +Add.
2. Search for Ivanti Tunnel.
3. Click one of the following:
   - Ivanti Tunnel: Use this configuration to create a per-app VPN configuration for Ivanti Tunnel.
   - Ivanti Tunnel (On Demand): Use this configuration to create a per-device VPN configuration for Ivanti Tunnel.

The Ivanti Tunnel configuration page displays.

4. If you selected the Ivanti Tunnel configuration, click iOS/macOS.
   The configuration options display.
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 Ivanti Tunnel for macOS app.

Next steps

Go to Applying the Ivanti Tunnel VPN setting to managed apps (Ivanti Neurons for MDM).

Related topics

- For a description of the configuration fields for Ivanti Tunnel VPN, see Ivanti Tunnel VPN configuration field description.
- For a description of the key-value pairs, see Key-value pairs for Ivanti Tunnel for macOS.

## Applying the Ivanti Tunnel VPN setting to managed apps (Ivanti Neurons for MDM)

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 Ivanti Tunnel VPN setting you created for macOS. This procedure is not needed if you configured per-device VPN using Ivanti Tunnel (On Demand).

**Procedure**

1. In Apps > App Catalog, add or edit an app.
2. In App Configuration for the app, for Per App VPN click +.
3. Enter a name for the configuration.
4. Check Enable Per-App VPN for this app.
5. Select the Ivanti Tunnel for macOS 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 *Ivanti Neurons for MDM Guide*:

- "Adding an app from a public store."
- "Adding an In-house app."

## Distribute Ivanti Tunnel for macOS as a VPP app (Ivanti Neurons for MDM)

Ivanti Tunnel is available in the Mac App Store. Device users can download the app directly from the Mac App Store.

Ivanti Tunnel can also be distributed as a Volume Purchase Program (VPP) app from Ivanti Neurons for MDM. Apple provides VPP to facilitate app purchase and distribution within an organization. The App Store Volume Purchase Program (VPP) allows participating organizations to purchase iOS and macOS apps in volume and distribute the apps to their users.

While Apple supports user-based licensing for macOS VPP apps, currently there is an Apple issue with the installation of user-based licensed VPP apps through MDM. As a result, Ivanti does not recommend applying user-based licenses to macOS VPP apps.

For information on how to distribute Ivanti Tunnel for macOS as a VPP app, see "Licenses" in the *Ivanti Neurons for MDM Guide* or by clicking Help in Ivanti Neurons for MDM.

> For countries for which a VPP program is not available, device users can download and install Ivanti Tunnel directly from the Mac App Store.

# 4.4. Ivanti Tunnel VPN configuration field description

The following table provides field descriptions for the Ivanti Tunnel configuration. There are some variations in field names between Ivanti EPMM and Ivanti Neurons for MDM.

| Item | Description |
| --- | --- |
| Name | Enter a name for the Ivanti Tunnel VPN profile. |
| Description | Enter a description for the profile. |
| Connection Type<br>(Ivanti EPMM) | Select Ivanti Tunnel.<br>Only fields relevant to Ivanti Tunnel are displayed. |
| Choose OS to create Ivanti Tunnel Configuration<br>(Ivanti Neurons for MDM) | Select iOS/macOS. |
| Profile selection mode to use for this configuration<br>(Ivanti Neurons for MDM) | Select one of the following:<br>• Sentry Profile Only: Select if Tunnel traffic goes only through Standalone Sentry only.<br>• Access Profile Only: Select if Tunnel traffic goes to Access only. Only authentication traffic is tunneled to Access. This option is available only if an Access deployment is set up.<br>If 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 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.<br>• 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 an Access as a service deployment is set up. |
| Legacy App Support (iOS only) | Not applicable for Ivanti Tunnel for macOS. |
| VPN Sub Type<br>(Ivanti Neurons for MDM) | (Optional) Overrides the bundle identifier for a customized Ivanti Tunnel app. |
| Enable Access<br>(Ivanti EPMM) | This option is not supported for macOS. |
| Provider Type | app-proxy: This is the default setting. Use this setting for TCP tunneling only.<br>packet-tunnel: Select to allow Ivanti Tunnel to also handle IP traffic.<br>Per-device VPN automatically uses the packet-tunnel provider type. |
| Per-app VPN<br>(Ivanti EPMM) | 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.<br>Yes: This is the default setting. Connectivity is established for an app, rather than the device. |

| Item | Description |
| --- | --- |
| | No: Select to establish connectivity for the device, rather than just an app. |
| Sentry (Profile) | **Ivanti EPMM:** Select the Standalone Sentry on which you created the TCP tunnel service.<br>Ivanti Neurons for MDM: If you are configuring per app VPN, select the Standalone Sentry profile. |
| Sentry Service | **Ivanti EPMM:** Select the TCP or IP service that the Safari browser or managed app will use.<br>Ivanti Neurons for MDM: Select the Ivanti Tunnel for macOS service that the Safari browser or managed app will use.<br>The field is not available if the profile mode is Access Profile Only.<br>Only TCP services are available for selection if the provider type is app-proxy.<br>Only IP services are available for selection for the following:<br>• The provider type is packet-tunnel.<br>• The VPN is per device. |
| SCEP Identity<br>(Ivanti Neurons for MDM) | Select the Identity Certificate configuration you created for Ivanti Tunnel.<br>The Identity Certificate is automatically selected if Sentry Profile Only or Sentry + Access Profile is enabled. |
| Debug Info Recipient<br>(Ivanti Neurons for MDM) | Enter an email address for emailing debug logs. |
| Identity Certificate<br>(Ivanti EPMM) | Select the certificate setting you created.<br>If you are using user-provided certificates, select the user provided certificate you created for Ivanti Tunnel. |
| On Demand Rules (iOS 9 and later; macOS 10.13 and later) (Ivanti EPMM)<br>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.<br>• A matching rule is not required. The Default Rule is applied if a matching rule is not defined.<br>• If you select Evaluate Connection, a matching rule is not required.<br>• You can create up to 10 On Demand matching rules.<br>• For each matching rule you can create up to 50 Type and Value pairs. | |

| Item | Description |
| --- | --- |
| An Ethernet on-demand rule is only applicable to macOS devices. If the rule is pushed to macOS device, the rule may cause issues with Tunnel behavior and how traffic through Tunnel is handled. Therefore, Ivanti recommends using separate Tunnel VPN configurations for iOS and macOS. | |
| Add + | Click to add a new On Demand matching rule. |
| On Demand Action | Select one of the following actions to apply to the matching rule:<br>• Connect<br>• Disconnect |
| Matching Rules<br>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.<br>A dialog box appears. |
| Type | Select the following key type:<br>• 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.<br>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<br>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<br>The device user can access servers ending with these domains in Safari.<br>Ivanti Tunnel configuration is only applied to a managed app. Therefore, a managed app with the Ivanti Tunnel configuration must be installed on the device for the device user to access the domains from Safari.<br>If the device resolves the destination domain, then Tunnel is not launched. | |
| Safari Domain | Enter a domain name. |

| Item | Description |
| --- | --- |
| | Only alphanumeric characters and periods (.) are supported. |
| 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)<br>A Tunnel VPN connection is automatically established for these domains.<br>Only available for per-app VPN. | |
| Calendar Domain | Enter a domain name.<br>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)<br>A Tunnel VPN connection is automatically established for these domains.<br>Only available for per-app VPN. | |
| Contact Domain | Enter a domain name.<br>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)<br>A Tunnel VPN connection is automatically established for these domains.<br>Only available for per-app VPN. | |
| Mail Domain | Enter a domain name.<br>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)<br>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.<br>Enter list of IPv4 addresses in CIDR format.<br>For multiple values, enter a semicolon separated list. | |
| DNS Resolver IPs<br>Only for packet-tunnel provider type.<br>Enter a domain name server (DNS) to resolve the IP address.<br>IPv4 only.<br>For multiple values, enter a semicolon separated list. Ensure that the DNS is routable if the default route in not used. | |

| Item | Description |
| --- | --- |
| If DNS is not configured, the Sentry DNS is used. | |
| DNS Search Domain List<br>Only for packet-tunnel provider type.<br>Enter DNS search domains for resolving the domain names. IPv4 only.<br>For multiple values, enter a semicolon separated list. | |
| Match Domain List<br>Only for packet-tunnel provider type.<br>Enter domains for which a proxy connection is used. IPv4 only.<br>For multiple values, enter a semicolon separated list.<br>Only for the domains configured here, Tunnel checks the DNS configured in DNS Resolver IPs to resolve the domains. DNS look up for all other domains goes directly to the device network's DNS server. If no domains are configured here, Tunnel checks the VPN's DNS to resolve all domains. | |
| Custom Data<br>Enter Key Value pair to configure the Ivanti Tunnel VPN disconnect, debug, and timeout behavior.<br>See **Key-value pairs for Ivanti Tunnel for macOS**. | |

# 5. Additional configurations using key-value pairs for Ivanti Tunnel

Key-value pairs are used to customize Ivanti Tunnel for macOS 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.

# 5.1. Key-value pairs for Ivanti Tunnel for macOS

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

| Key | Value |
| --- | --- |
| Manage Tunnel timeout | |
| disconnectTimeoutInSeconds (Ivanti EPMM) | Enter 0 or a number between 5 - 18000.<br>If the value is 0, then Ivanti Tunnel VPN never disconnects itself. You have to manually disconnect the VPN in the Ivanti Tunnel.<br>If the value is > 0, the Ivanti Tunnel VPN is disconnected after number entered.<br>Default value if the key-value pair is not configured: 60 seconds. |
| TcpIdleTmoMs | Enter an integer between 5000 - 1800000.<br>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.<br>The default idle timeout with Standalone Sentry for per app VPN if the key-value pair is not configured: 60 seconds. |
| Troubleshooting | |
| debugInfoRecipient (Available as field value in Ivanti Neurons for MDM) | Enter an email address to forward the debug information. |
| LogLevel | • error: Captures error logs if the Tunnel app errors out while performing an action.<br>Enter debug <Log Level><br>Use one of the following log level options. The options are listed from the least to the most verbose level.<br>• warning: Captures warning messages logged if there is missing or incorrect information that might cause an error. This log level is rarely used.<br>• info: Captures informational level details such as, log prints inputs, metadata, parameter values.<br>• debug: Captures debug level information such as, actions, operations, values of critical data, and information that is helpful in debugging.<br>• session: Captures everything that occurs during a tunnel session.<br>• packet: Captures packet level information, such as, length in bytes. Used for troubleshooting DNS queries and responses to and from Tunnel.<br>Default if the key-value pair is not configured: info |
| AnalyticsEnabled | **true**: Enables collection of analytics data in network extension.<br>**false**: Disables collection of analytics data in network extension.<br>Default value if the key-value pair is not configured: **true**. |
| DNS and network | |
| IPv6NetworkPrefix | IPv6 ULA network prefix to use for internal NAT table. |
| Certificates | |
| DisablePinning | false: Default, if the key-value pair is not configured. Certificate pinning is enabled.<br>true: Certificate pinning is disabled. Disabling certificate pinning is not recommended for security reasons. |

| Key | Value |
| --- | --- |
| | The Standalone Sentry server certificate is automatically pushed to the device. |
| Packet-tunnel | |
| IPRoutes | IP routes of the iOS or macOS device VPN. Enter list separated by semicolon.<br>The default value if the key-value is not configured is 0.0.0.0/0<br>Example<br>10.0.0.0/8;172.16.0.0/16<br>Ivanti recommends configuring IP routes for better Tunnel performance. |
| ExcRoutes | IP routes that will be excluded from IPRoutes.<br>Example<br>10.10.10.10/32. |
| SplitUDPPortList | List the destination UDP ports of the UDP packets that want to be sent through VPN. All other UDP packets are sent directly to destination from Tunnel client.<br>If the key-value pair is not configured all UDP packets from the VPN interface go through VPN.<br>Example<br>53;161-162;200-1024 |
| MTU | Tunnel MTU.<br>The default value if the key-value is not configured is 1400. |
| TunIP | IP address of the VPN network interface.<br>Configure only if the customer network is in the same range.<br>Example<br>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.<br>The minimum idle time is based on the last inbound response received by 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.<br>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.<br>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.<br>Default value if the key-value pair is not configured: 5 |
| App proxy | |
| DirectLocalhost | Enter true.<br>Configure if using app proxy Tunnel. The key-value pair is required for Tunnel to handle app proxy localhost traffic from apps.<br>true: If an app uses localhost, ::1, or 127.0.0.1, the localhost app proxy (TCP) traffic is redirected to the device itself. |

# 6. What users see in Ivanti Tunnel for macOS

The following provide information about device user experience:

# 6.1. Ivanti Tunnel installation on macOS

Ivanti Tunnel is distributed to macOS devices from Ivanti EPMM using Apple Volume Purchase Plan (VPP). For information on distributing Ivanti Tunnel for macOS, see

- Ivanti EPMM deployments: **Distribute Ivanti Tunnel for macOS as a VPP app (Ivanti EPMM)**.
- Ivanti Neurons for MDM deployments: **Distribute Ivanti Tunnel for macOS as a VPP app (Ivanti Neurons for MDM)**

No user action is required for installing Ivanti Tunnel on macOS. The app is automatically installed on macOS devices when the administrator applies the VPP label to the app in the app catalog in Ivanti EPMM. Users see the Tunnel icon on the top menu bar. The Tunnel icon is also seen on the Launchpad. However, Tunnel cannot be launched from the Launchpad.

## Allow keychain

When users access enterprise resources from a Safari browser or a managed app to which the Tunnel VPN configuration is applied, Ivanti Tunnel is automatically used to secure access to the enterprise resource. The experience is seamless to the end user. However, when the Tunnel is first pushed to the device or when the Tunnel configuration is re-pushed to the device, macOS prompts users to authorize the key. When prompted to authorize, click Allow.

If users either click Deny or ignore the prompt, the dialog is presented the next time Tunnel tries to connect.

- If Tunnel is in use, the app cannot be removed from the device. Users will see the following message: The item can’t be moved to the Trash because some of its extensions are in use.
- Per app VPN connection status is not provided on macOS.
- For countries for which a VPP program is not available, device users can download and install Ivanti Tunnel directly from the Mac App Store.
- Users cannot launch Ivanti Tunnel from the Launchpad.

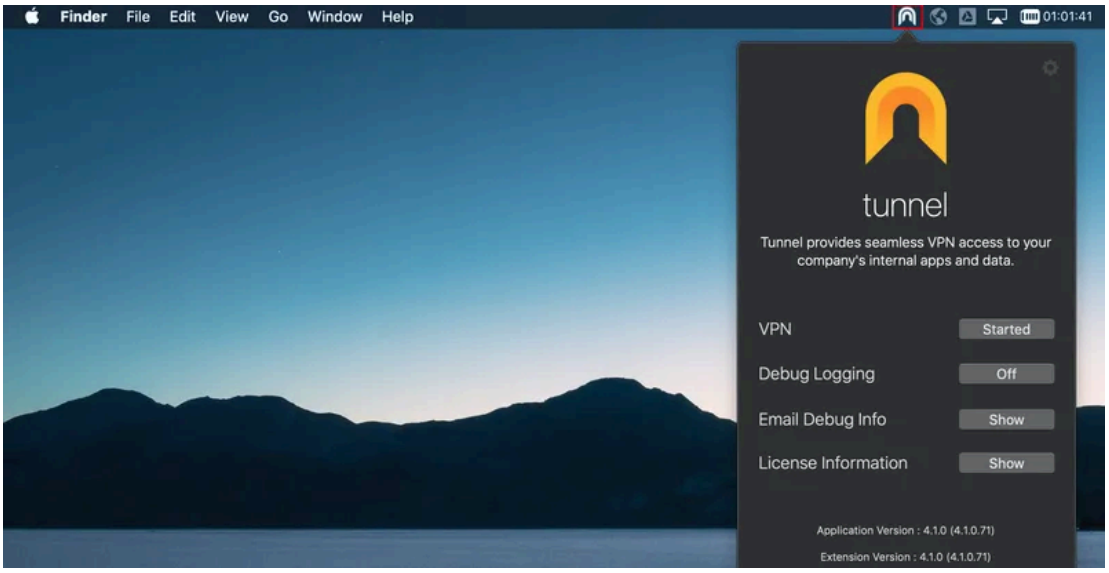
# 6.2. Ivanti Tunnel icon on macOS devices

If Tunnel is deployed to macOS devices, users see the Tunnel icon in the top menu bar.

The icon is highlighted if Tunnel VPN is actively connected. The icon is grayed out if Tunnel is not in use. However, the Ivanti Tunnel application is always running in the background.

If users click on the Tunnel icon, the icon expands to display the Tunnel status and additional options.

Figure 1. Tunnel icon and expanded options




The following table describes the options in the Tunnel window.

| Option | Description |
|---|---|
| VPN | Click to change the status of the VPN connection. Displays one of the following status.<br>• Started: The VPN connection is active.<br>• Stopped: The VPN connection is inactive. |
| Debug Logging | Click to turn on debug logging.<br>The default is Off.<br>Enabling debug logging saves the Tunnel logs to a file so that it can be emailed. |
| Email Debug Info | Click to email the log file.<br>Log information is included only if there has been activity using the Tunnel app. |
| License Information | Click to view the license information. |

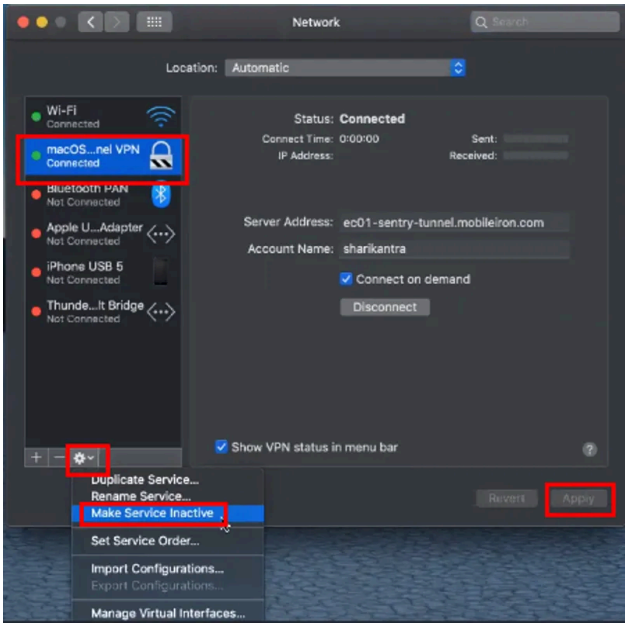
# 6.3. Exiting Ivanti Tunnel on macOS devices

Once Tunnel is deployed, the application is always running in the background. However, if needed, users have the option to exit the Tunnel application. To exit the application, users must first deactivate Tunnel.

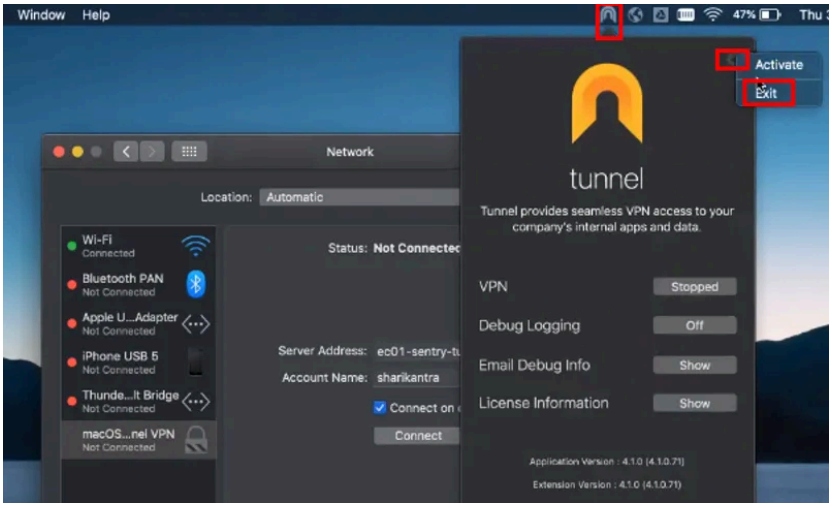
> If the macOS device is not registered or a Tunnel VPN setting has not been pushed to the device, Ivanti Tunnel will not be activated.

**Procedure**

1. On the macOS device, go to System Preferences > Network Preferences.
2. Select the Ivanti VPN from the list of connected networks.
   Figure 1. Make Service Inactive




1. Click the gear icon > Make Service Inactive.
2. Click Apply.
3. Click the Tunnel icon in the menu bar.
   Figure 2. Exit application



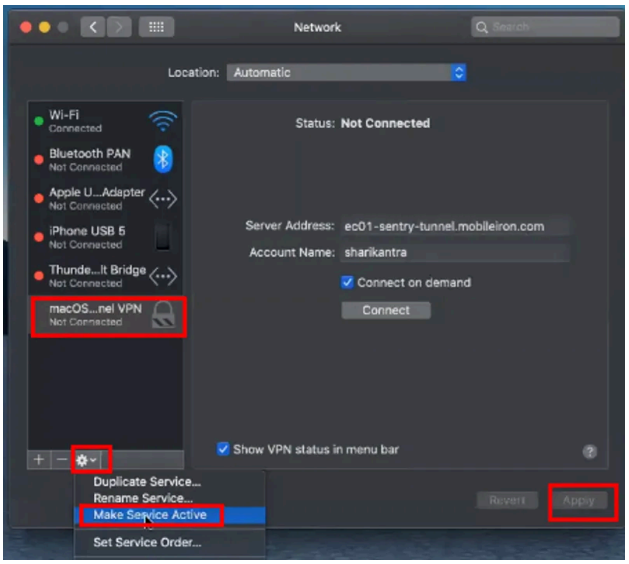

1. On the top right corner, click the gear icon > Exit.
   The Tunnel icon is no longer seen on the menu bar.

# 6.4. Activating Ivanti Tunnel on macOS devices

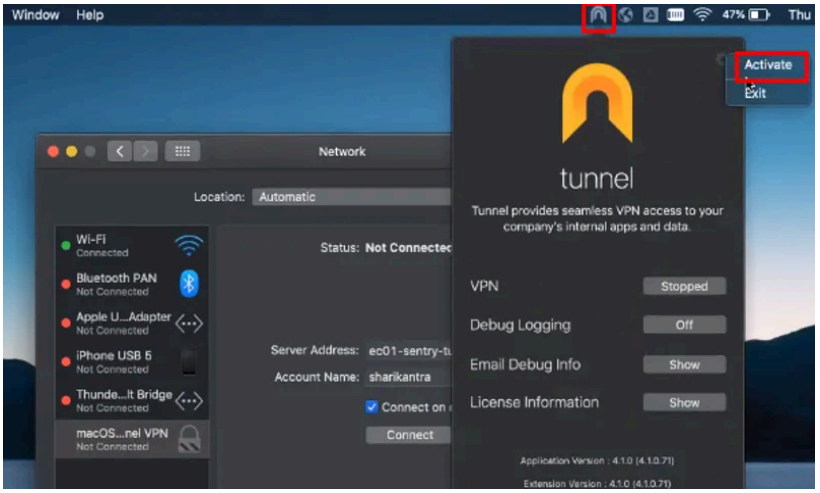
To active Tunnel from the Tunnel menu, users must first activate the service in System Preference > Network Preferences.

**Procedure**

1. On the macOS device, go to System Preferences > Network Preferences.
2. Select the Ivanti Tunnel VPN from the list of networks.
   Figure 1. Make Service Inactive




1. Click the gear icon > Make Service Active.
2. Click Apply.
3. Click the Tunnel icon in the menu bar.
   Figure 2. Exit application




1. On the top right corner, click the gear icon > Activate.

# 6.5. Emailing debug log information

IT administrators may require Tunnel debug and log data for troubleshooting purposes. Users can email the Tunnel debug and log file to the IT administrator.

**Procedure**

1. In the Tunnel app, turn on Debug Logging.
   Enabling debug logging saves the Tunnel logs to a file so that it can be emailed.
2. Redo the steps that resulted in the error so that the logs can be captured.
3. Click Email Debug Info.
   Log information is included only if there has been activity using the Tunnel app.
4. Enter the email address provided by your administrator.
5. Click Send.

# 6.6. Sharing debug logs

Ivanti Tunnel for macOS logs are now saved locally and you can email the debug info.

