Pulse Secure Desktop Client: Administration Guide
Supporting Pulse Secure Desktop Client 9.1R9
Pulse Secure, LLC
2700 Zanker Road,
Suite 200 San Jose
CA 95134

www.pulsesecure.net

© 2020 by Pulse Secure, LLC. All rights reserved.

Pulse Secure and the Pulse Secure logo are trademarks of Pulse Secure, LLC in the United States. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

Pulse Secure, LLC assumes no responsibility for any inaccuracies in this document. Pulse Secure, LLC reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

**END USER LICENSE AGREEMENT**

The Pulse Secure product that is the subject of this technical documentation consists of (or is intended for use with) Pulse Secure software. Use of such software is subject to the terms and conditions of the End User License Agreement (“EULA”) posted at http://www.pulsesecure.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.
# Contents

CONTENTS ................................................................. i

REVISION HISTORY ......................................................... 1

PREFACE ................................................................. 3
  DOCUMENT CONVENTIONS ................................................ 3
  TEXT FORMATTING CONVENTIONS .................................... 3
  COMMAND SYNTAX CONVENTIONS ..................................... 3
  NOTES AND WARNINGS .................................................. 4
REQUESTING TECHNICAL SUPPORT .................................... 4
  SELF-HELP ONLINE TOOLS AND RESOURCES .......................... 4
  OPENING A CASE WITH PSGSC ......................................... 5
  REPORTING DOCUMENTATION ISSUES ................................ 5

OVERVIEW OF UNIFIED PULSE SECURE CLIENT ......................... 6
  INTRODUCING UNIFIED PULSE SECURE CLIENT ..................... 6
  USING UNIFIED PULSE SECURE CLIENT INTERFACE .................. 7
    TO LAUNCH UNIFIED PULSE SECURE CLIENT FROM DESKTOP .... 7
    TO LAUNCH UNIFIED PULSE SECURE CLIENT FROM THE TERMINAL 7
  ADDING VPN CONNECTION .............................................. 7
  MODIFYING VPN CONNECTION ......................................... 8
  DELETING VPN CONNECTION ........................................... 9
  INITIATING VPN CONNECTION ......................................... 9
  TERMINATING VPN CONNECTION ..................................... 10
  ADVANCED CONNECTION DETAILS .................................... 11
  ABOUT PULSE LINUX CLIENT .......................................... 11

PULSE CLIENT FOR WINDOWS ........................................... 12
PULSE CLIENT FOR macOS ................................................ 19
PULSE CLIENT FOR LINUX ............................................... 20

USER EXPERIENCE ....................................................... 20
  L3 AND PULSE SAM COEXISTENCE .................................... 25
  HVCI COMPATIBILITY .................................................. 26
  PULSE SAM IPv6 SUPPORT ............................................... 26
  LOCATION AWARENESS .................................................. 28
  CENTRALIZED PULSE CLIENT CONFIGURATION MANAGEMENT ....... 28
  SESSION MIGRATION .................................................... 29
  SMART CONNECTIONS - LIST OF URLS ................................. 29
**LAUNCHING Unified Pulse Secure Client from the Pulse Secure server Web Portal**  
62

- Usage Notes. ................................................................. 62
- Launching Unified Pulse Secure Client using URL. .................. 63
- Benefits. ........................................................................... 67

**Installing Unified Pulse Secure Client on Windows Endpoints Using a Preconfiguration File** .................................................. 69

- Installing Unified Pulse Secure Client using Advanced Command-Line Options 71
- Examples. ........................................................................... 71
- Repairing a Unified Pulse Secure Client Installation on a Windows Endpoint 71

**Installing Unified Pulse Secure Client on OS X Endpoints Using a Preconfiguration File** 72

- Installing Unified Pulse Secure Client on OS X Endpoints Using Command-Line Options 73
- Installing Unified Pulse Secure Client on Linux Using Command-Line Options 73

**Unified Pulse Secure Client Command-line Launcher** .................. 74

- Examples. ........................................................................... 76
- Using JAMCOMMAND to Import Pulse Secure Connections. .... 77
- JAMCOMMAND Reference ................................................... 78
- Managing Server Certificate Authorities ............................... 79
- Chromium Embedded Framework (CEF) Support .................. 81
- CEF Installation on UI ....................................................... 81
- CEF Installation Using Scripts ........................................... 83

**Customizing Pulse Secure Desktop Client** .............................. 85

- Customizing Pulse Secure Desktop Client Overview. ............ 85
- BrandPackager Workflow ................................................... 86
- Setting Up the Pulse Client Customization Environment ....... 87
- Initializing the Pulse Client Customization Environment ..... 88
- Importing an Existing Customized Pulse Client Package .... 89
- Editing Pulse Client User Interface Labels. ......................... 89
- Editing Pulse Client Messages .......................................... 93
- Adding Custom Graphics to Pulse Client ......................... 94
- Customizing Pulse Client for Apple OS X Online Help. .... 95
- Validating Customizations to Pulse Client .......................... 96
- Building the New Pulse Client Package ............................ 96
- Testing the Pulse Client Package .................................... 96
- Installing or Upgrading Pulse Client for Windows with a Branding Package 97
- Installing or Upgrading Pulse Client for Apple OS X with a Branding Package 97
The following table lists the revision history for this document:

<table>
<thead>
<tr>
<th>Revision</th>
<th>Document Version</th>
<th>Date</th>
<th>Feature</th>
<th>Add/Update/Remove</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1R9</td>
<td>1.6</td>
<td>October 2020</td>
<td>Single Logout (SLO); PZTA with Pulse Client</td>
<td>Added new sections; PCS and PPS related configurations removed and published as new documents. Android and iOS related information removed and published as new documents.</td>
</tr>
<tr>
<td>9.1R8</td>
<td>1.5</td>
<td>July 2020</td>
<td></td>
<td>Cosmetic Changes</td>
</tr>
<tr>
<td>9.1R5</td>
<td>1.4</td>
<td>April 2020</td>
<td>PSAM Session tab Disabling JNPRNS driver Using Advanced Client Configuration</td>
<td>Updated Pulse Secure Client for Windows section. Included Disabling JNPRNS driver Using Advanced Client Configuration.</td>
</tr>
<tr>
<td>9.1R4</td>
<td>1.3.1</td>
<td>February 2020</td>
<td></td>
<td>Cosmetic Changes</td>
</tr>
<tr>
<td>9.1R4</td>
<td>1.3</td>
<td>January 2020</td>
<td></td>
<td>Updated Pulse Secure Client for Windows section. Updated Configuring Client Certificate Selection Option section.</td>
</tr>
<tr>
<td>1.2.1</td>
<td></td>
<td>October 2019</td>
<td></td>
<td>Cosmetic Changes</td>
</tr>
<tr>
<td>9.1R3</td>
<td>1.2</td>
<td>October 2019</td>
<td>Configuring Client Certificate Selection Option</td>
<td>Added &quot;Configuring Client Certificate Selection Option&quot; section.</td>
</tr>
<tr>
<td>9.1R2</td>
<td>1.1</td>
<td>July 2019</td>
<td>Managed Pulse Secure Client Versions</td>
<td>Updated &quot;Manage Pulse Secure Client Versions&quot; section.</td>
</tr>
<tr>
<td>9.1R1</td>
<td>1.0</td>
<td>May 2019</td>
<td>Launching Pulse Secure Desktop Client using a URL</td>
<td>Added &quot;Launching Pulse Secure Desktop Client using a URL&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Pulse SAM IPv6 Support</td>
<td>Added &quot;Pulse SAM IPv6 Support&quot;.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Pulse Secure Client and Software Defined Perimeter</td>
<td>Added &quot;Pulse Secure Desktop Client and Software Defined Perimeter (SDP)&quot;.</td>
</tr>
</tbody>
</table>
Preface

- Document conventions .............................................. 3
- Requesting Technical Support ...................................... 4
- Reporting Documentation Issues ................................... 5

Document conventions
The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in Pulse Secure technical documentation.

Text formatting conventions
Text formatting conventions such as boldface, italic, or Courier font may be used in the flow of the text to highlight specific words or phrases.

<table>
<thead>
<tr>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bold text</strong></td>
<td>Identifies command names</td>
</tr>
<tr>
<td></td>
<td>Identifies keywords and operands</td>
</tr>
<tr>
<td></td>
<td>Identifies the names of user-manipulated GUI elements</td>
</tr>
<tr>
<td></td>
<td>Identifies text to enter at the GUI</td>
</tr>
<tr>
<td><em>italic text</em></td>
<td>Identifies emphasis</td>
</tr>
<tr>
<td></td>
<td>Identifies variables</td>
</tr>
<tr>
<td></td>
<td>Identifies document titles</td>
</tr>
<tr>
<td><strong>Courier Font</strong></td>
<td>Identifies command output</td>
</tr>
<tr>
<td></td>
<td>Identifies command syntax examples</td>
</tr>
</tbody>
</table>

Command syntax conventions
Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bold text</strong></td>
<td>Identifies command names, keywords, and command options.</td>
</tr>
<tr>
<td><em>italic text</em></td>
<td>Identifies a variable.</td>
</tr>
<tr>
<td>[]</td>
<td>Syntax components displayed within square brackets are optional.</td>
</tr>
<tr>
<td></td>
<td>Default responses to system prompts are enclosed in square brackets.</td>
</tr>
</tbody>
</table>
Notes and Warnings

Note, Attention, and Caution statements might be used in this document.

**Note:** A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

**ATTENTION**

An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

**CAUTION**

A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

Requesting Technical Support

Technical product support is available through the Pulse Secure Global Support Center (PSGSC). If you have a support contract, file a ticket with PSGSC.

- Product warranties—For product warranty information, visit https://support.pulsesecure.net/product-service-policies/

Self-Help Online Tools and Resources

For quick and easy problem resolution, Pulse Secure provides an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: https://support.pulsesecure.net
- Search for known bugs: https://support.pulsesecure.net
- Find product documentation: https://www.pulsesecure.net/techpubs
- Download the latest versions of software and review release notes: https://support.pulsesecure.net
• Open a case online in the CSC Case Management tool: https://support.pulsesecure.net
• To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://support.pulsesecure.net

For important product notices, technical articles, and to ask advice:

• Search the Pulse Secure Knowledge Center for technical bulletins and security advisories: https://kb.pulsesecure.net
• Ask questions and find solutions at the Pulse Community online forum: https://community.pulsesecure.net

Opening a Case with PSGSC

You can open a case with PSGSC on the Web or by telephone.

• Use the Case Management tool in the PSGSC at https://support.pulsesecure.net.
• Call 1-844 751 7629 (Toll Free, US).

For international or direct-dial options in countries without toll-free numbers, see https://support.pulsesecure.net/support/support-contacts/

Reporting Documentation Issues

To report any errors or inaccuracies in Pulse Secure technical documentation, or to make suggestions for future improvement, contact Pulse Secure Technical Support (https://support.pulsesecure.net). Include a full description of your issue or suggestion and the document(s) to which it relates.
Overview of Unified Pulse Secure Client

- Introducing Unified Pulse Secure Client .................................................. 6
- Using Unified Pulse Secure Client Interface ........................................... 7
- Pulse Client for Windows ........................................................................ 12
- Pulse Client for macOS ......................................................................... 19
- Pulse Client for Linux ............................................................................ 20
- User Experience ...................................................................................... 20
- Pulse Client and Software Defined Perimeter (SDP) ............................... 33
- Pulse Client Configuration Overview ..................................................... 34
- Pulse Client Status Icons ...................................................................... 35
- Installation Requirements ...................................................................... 36
- Pulse Client Error Messages Overview ................................................ 36
- Accessing Pulse Client Error Messages on macOS Endpoints ................ 37
- Pulse Client Log Files ............................................................................ 37
- Deleting Pulse Client Log Files .............................................................. 41
- Uploading Pulse Client Log Files ............................................................ 41
- Migrating from Odyssey Access Client to Pulse Client ............................ 42
- Migrating from Network Connect to Pulse Client ................................... 44
- Predictable Pulse Secure Server Hostname Resolution with IPv6 ........... 45

Introducing Unified Pulse Secure Client

Unified Pulse Secure Client (Pulse Client) is an extensible multi-service network client that supports integrated connectivity and secure location-aware network access. Pulse Client simplifies the user experience by letting the network administrator configure, deploy, and control the Pulse Client software and the Pulse Client connection configurations that reside on the endpoint.

The Pulse Secure suite comprises client and server software. The client enables secure authenticated network connections to protected resources and services over local and wide area networks. The Pulse Client software can connect with Pulse Connect Secure to provide remote access to enterprise and service provider networks. Pulse Client also delivers secure, identity-enabled network access control (NAC) for LAN-based network and application access when it is deployed with Pulse Policy Secure. Pulse Client also integrates with Pulse Collaboration Suite for online meeting services.

Users of mobile devices (smart phones and tablets) can install the Pulse Secure Client for Mobile Devices (Pulse Mobile Client) app from the respective app stores for secure connectivity to Pulse Connect Secure. Windows 8.1 (Pro and RT) introduced a Pulse Secure VPN client as part of the operating system.
Using Unified Pulse Secure Client Interface

To launch Unified Pulse Secure Client from Desktop
Launch Unified Pulse Secure Client by searching for Pulse Secure Icon under Applications List.

Figure 1 Pulse Secure Client Application

To launch Unified Pulse Secure Client from the Terminal
Launch the UI by executing the below command

```
/opt/pulsesecure/bin/pulseUI
```

Adding VPN Connections

To create a Pulse Secure VPN connection on a device:

Figure 2 Adding VPN Connection

- Click Add icon on the top-right-hand corner of the main Unified Pulse Secure client screen.
1. In the **Name** field, specify the name for the Pulse Connect Secure gateway.

2. In the **Server URL** field, specify the URL for the Pulse Connect Secure gateway. You can identify the server using the server IP address, the hostname, or a URL that optionally specifies the port the connection uses and the specific sign-in page. To specify an URL, use the following format:

   `https://hostname[:port][/][sign-in page]`

   The brackets indicate options. If you specify a specific sign-in page, make sure that the name you specify matches what is defined on the Pulse Connect Secure gateway. (Authentication > Signing in > Sign-in pages.)

3. Click **Save**. The new VPN connection appears in the VPN list.

   Click **Connect** to initiate a VPN connection. The VPN connection state is indicated in the VPN dropdown menu on the VPN list.

Use “JamCommand” tool to import connections to client store using CLI.

   `/opt/pulsesecure/bin/jamCommand /ImportFile ~/Downloads/pulsepreconfig`

### Modifying VPN Connection

To modify a Pulse Secure VPN connection on a device:

**Figure 3  Figure 3 Modifying VPN Connection**

1. Select the VPN connection and click the edit icon on the top-right-hand corner of the main Unified Pulse Secure Client screen.

2. In the **Name** field, specify the name for the Pulse Connect Secure gateway.
3. In the **Server URL** field, specify the URL for the Pulse Connect Secure gateway. You can identify the server using the server IP address, the hostname, or a URL that optionally specifies the port the connection uses and the specific sign-in page. To specify an URL, use the following format: https://hostname[:port][/]sign-in page

The brackets indicate options. If you specify a specific sign-in page, make sure that the name you specify matches what is defined on the Pulse Connect Secure gateway. (Authentication > Signing in > Sign-in pages.)

4. Click **Save**, modified VPN connection appears in the VPN list.

   Tap **Connect** to initiate a VPN connection. The VPN connection state is indicated in the VPN dropdown menu on the VPN list.

### Deleting VPN Connection

To delete a Pulse Secure VPN connection on a device:

1. Select the VPN connection and click on the delete icon on the top-right-hand corner of the main Unified Pulse Secure Client screen.

2. VPN connection is removed from the VPN list after user click the **OK** button on the above screen.

### Initiating VPN Connection

To initiate a Pulse Secure VPN connection on a device:
Figure 5  Initiating VPN Connection

1. Select the VPN connection and click **Connect** on the main screen.
2. New window opens to continue authentication process based on the authentication method configured for the realm.

**Terminating VPN Connection**

To terminate a Pulse Secure VPN connection on a device:

Figure 6  Terminating VPN Connection

Select the VPN connection and click **Disconnect** on the main screen.

**Note:** Unified Pulse Secure Client automatically attempts to reconnect in case of an interrupted connection, such as temporarily losing the Wi-Fi link.
Advanced Connection Details

Advanced connection details page shows the status of the selected VPN connection from the list.

To view advanced connection details, navigate to File > Connections > Advanced Status Details.

Figure 7  Advanced Connection Details

Note: To upload logs to the VPN server, the user needs to be authenticated with an active session.

About Pulse Linux Client

To view Pulse Linux client details:

1. Click About button on main Unified Pulse Secure Client UI.

Figure 8  About Pulse Linux client
Pulse Client for Windows
The Pulse Client for Windows user interface (see Figure 9) lists the deployed Pulse Client connections. Each connection is a set of properties that enables network access through a specific Pulse Secure server. The user can expand a connection to see more details about the connection.

Note: From 5.3R2, Pulse Client connects to PSA device through proxy at the first attempt and then try connecting directly upon failure.

VPN Connection Details
To view the VPN connection details dialog:

1. Select the VPN connection from the list of connection items.
2. Click File --> Connections --> Advanced Connection Details, or Right click the selected connection to get the context menu, refer to the following figure.

Figure 9  Pulse Secure Client for Windows - Sub Menu for VPN Options

The Advanced Connection Detail information window will not update automatically. For example, the session time remaining shows how much time remains when you open the dialog. To update advanced detail information, click Refresh or click the check box labeled automatically refresh.
The Advanced Connection Details window gives the following information:
### Table 1  Pulse Client for VPN Advanced Connection Details:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session time remaining</td>
<td>The duration that the current VPN session will remain active before credentials must be re-entered or the session manually extended.</td>
</tr>
<tr>
<td>Session Duration</td>
<td></td>
</tr>
<tr>
<td>Tunnel type</td>
<td>This describes that the connection is a VPN tunnel.</td>
</tr>
<tr>
<td>VPN type</td>
<td>The protocol used to create the tunnel (SSL or ESP).</td>
</tr>
<tr>
<td>Assigned IPv4</td>
<td>The IPv4 address assigned to the Pulse virtual adapter.</td>
</tr>
<tr>
<td>Bytes in</td>
<td>Number of bytes received through the tunnel.</td>
</tr>
<tr>
<td>Bytes out</td>
<td>Number of bytes sent through the tunnel.</td>
</tr>
<tr>
<td>Connection Source</td>
<td>This describes how the Pulse client received the connection entry:</td>
</tr>
<tr>
<td></td>
<td>If the value is Preconfigured, then the connection entry came from a Connection Set that was downloaded from a gateway.</td>
</tr>
<tr>
<td></td>
<td>And if the value is Dynamic, then it means that the connection entry was resulted from launching the Pulse client by connecting a web browser to a Pulse Secure gateway and pressing the &quot;Start&quot; button on the web page.</td>
</tr>
</tbody>
</table>

### PSAM Connection Details

To view the PSAM Advanced Connection Details dialog:

1. Select the PSAM connection from the list of connection items.
2. Click **File --> Connections --> Advanced Connection Details**, or right click the selected connection to get the context menu, refer to the following figure.

Figure 11  Pulse Secure Client for Windows - Sub Menu for SAM Options
The Advanced Connection Detail information will not update automatically. For example, the session time remaining shows how much time remains when you open the dialog. To update advanced detail information, click **Refresh** or click the check box labeled automatically refresh.

The Advanced Connection Details window gives the following information:

Table 2  Pulse Client for PSAM Advanced Connection Details

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session time remaining</td>
<td>The duration that the current VPN session will remain active before credentials must be re-entered or the session manually extended.</td>
</tr>
<tr>
<td>Session Duration</td>
<td></td>
</tr>
<tr>
<td>Tunnel type</td>
<td>This describes that the connection is a port/application mapping through SAM (Secure Access Manager).</td>
</tr>
<tr>
<td>VPN type</td>
<td>The protocol used to create the tunnel (SSL or ESP).</td>
</tr>
<tr>
<td>Bytes in</td>
<td>Number of bytes received through the tunnel.</td>
</tr>
<tr>
<td>Bytes out</td>
<td>Number of bytes sent through the tunnel.</td>
</tr>
</tbody>
</table>

**PSAM Application Details**

1. Select the PSAM connection from the list of connection items.
2. Click **File --> Connections --> Application Details**, or right click on the selected connection to get the context menu, refer to the following figure.
PSAM Session Details
To view the PSAM Session Details dialog:

1. Select the PSAM connection from the list of connection items.
2. Click File --> Connections --> SAM Session Details, or right click the selected connection to get the context menu, refer to the following figure.
The PSAM Session Details window gives the following information:

Table 3  Pulse Client for PSAM Session Details

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>The name of the application running in the active session.</td>
</tr>
<tr>
<td>Status</td>
<td>The status of the application in the session.</td>
</tr>
<tr>
<td>Protocol</td>
<td>The protocol used for the session.</td>
</tr>
<tr>
<td>Sent</td>
<td>Number of bytes sent in the established session.</td>
</tr>
<tr>
<td>Received</td>
<td>Number of bytes received in the established session.</td>
</tr>
<tr>
<td>Destination</td>
<td>The target used by the application.</td>
</tr>
</tbody>
</table>

Pulse Client also displays a system tray icon that provides connection status, and can allow the user to connect and disconnect and enables quick access to the program interface. One tray icon provides status for all active connections.

Typically, the network administrator defines and deploys the Pulse Client connections but you can also enable users to define, edit, and remove their own connections.

Table 4  Pulse Client for Windows Connection Status

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connected.</td>
<td></td>
</tr>
</tbody>
</table>

Figure 14  Session Details - PSAM
Pulse Client supports the Federal Information Processing Standard (FIPS), which defines secure communication practices for the U.S. government. If FIPS is enabled on the endpoint, "FIPS On" appears near the bottom the Pulse Client window.

A single system tray icon indicates the status of all active Pulse Client connections. You can right-click the system tray icon to control Pulse Client connections, to access Pulse Collaboration Suite meeting functions, to open the Pulse Client interface, or to exit from Pulse Client. The following table shows the connection status indicated by the system tray icon.

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Connecting" /></td>
<td>Connecting.</td>
</tr>
<tr>
<td><img src="image2" alt="Connected with limitations" /></td>
<td>Connected with limitations.</td>
</tr>
<tr>
<td><img src="image3" alt="Connection attempt failed" /></td>
<td>Connection attempt failed.</td>
</tr>
<tr>
<td><img src="image4" alt="Connection suspended" /></td>
<td>Connection suspended.</td>
</tr>
<tr>
<td><img src="image5" alt="Connected to the local network but no Internet access available" /></td>
<td>Connected to the local network but no Internet access available. Public WiFi locations often deploy a captive portal that requires the user to enter authentication information or to accept terms of service before network access is granted. Pulse Client detects the presence of captive portals and does not initiate a connection to a Pulse Secure server until Internet access is granted.</td>
</tr>
</tbody>
</table>
Table 5  Connection Status in the System Tray Icon.

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="null" alt="Image" /></td>
<td>No connection</td>
</tr>
<tr>
<td><img src="null" alt="Image" /></td>
<td>Connecting. A connection stays in this state until it fails or succeeds.</td>
</tr>
<tr>
<td><img src="null" alt="Image" /></td>
<td>Suspended</td>
</tr>
<tr>
<td><img src="null" alt="Image" /></td>
<td>Connected with issues</td>
</tr>
<tr>
<td><img src="null" alt="Image" /></td>
<td>Connection failed</td>
</tr>
<tr>
<td><img src="null" alt="Image" /></td>
<td>Connected</td>
</tr>
<tr>
<td><img src="null" alt="Image" /></td>
<td>Connected to the local network but no Internet access available. Public WiFi locations often deploy a captive portal that requires the user to enter authentication information or to accept terms of service before network access is granted. Pulse Client detects the presence of captive portals and does not initiate a connection to a Pulse Secure server until Internet access is granted.</td>
</tr>
</tbody>
</table>

**Pulse Client for macOS**

Pulse Client supports Apple computers running macOS. You deploy Pulse Client to Mac endpoints the same way you deploy the Windows client.

Pulse Client for Mac endpoints supports the following features:

- Connections to Pulse Policy Secure
- Connections to Pulse Connect Secure

Pulse Clients connect to the Pulse Connect Secure in SSL fallback mode.

- Connections to Juniper Networks SRX Series gateways.
  - macOS endpoints can connect to SRX Branch series SRX100-SRX650 gateways that are running a Junos OS release between v10.2 and v12.3, and that have dynamic VPN access enabled and configured. SRX gateways do not support deployment of Pulse Client.
  - Requires Pulse Client for Mac 5.0R3 or later and OS X 10.8 or later.
  - Pulse Client for Mac connect to the gateway as an IPsec IKEv1 VPN connection.
  - Pulse Dynamic VPN functionality is compatible with SRX-Branch (SRX100-SRX650) devices only. SRX Data Center (SRX1400-SRX5800 - also called SRX HE or High End) devices do not support Pulse Dynamic VPN from either Windows or Mac clients.
  - On Pulse Client for macOS, IPsec connections to SRX are unable to use the DNS IP address supplied by the SRX.
- Host Checker
Host Checker for macOS supports the following rules and remediation actions:
- Port
- Process
- File
- Custom IMC
- Enable Custom Instructions
- Kill Processes
- Delete Files
- Send reason strings

**Pulse Client for Linux**

Unified Pulse Secure Client for Linux provides secure connectivity between a device running Linux and Pulse Connect Secure. After installing the Pulse Secure client VPN package on a Linux device, the user can configure a connection and establish Layer 3 VPN communications.

The following features are supported by the Pulse Secure Client for Linux:

- Pulse Linux Client Usability Improvements
- VPN (SSL) connections to a Pulse Secure client SSL/VPN server.
- IPV6 mixed modes like IPv4 connections in IPV6 tunnels and vice versa
- Source IP enforcement through Pulse Policy Secure
- SAML and Custom Sign-in support for Linux
- 64-bit Operating Systems Support
- Multi-Factor Authentication (MFA) Support
- Host Checker
- Command Line Support
- RPM/DEB Package Support
- Pre- and post-authentication sign-in notification messages
- Client Certificate Authentication Support
- VPN tunneling connections for IPv4 and IPv6 resource access
- IP based split tunneling and route monitoring

**User Experience**

From the user perspective, Pulse Client presents a clean, uncomplicated interface. The user can enter credentials, select a realm, save settings, and accept or reject the server certificate. When you configure the client, you can specify whether to permit end users to modify settings, such as by adding connections.
Security Assertion Markup Language (SAML) Authentication

Pulse Client facilitates SAML authentication for Single Sign-on (SSO) in the following two ways:

- The Pulse Client user sees an embedded browser (see Figure 15) - if Enable embedded browser for authentication is enabled in “Pulse Client Connection Set Options” on page 136.

  Pulse Client will close the embedded browser, once the SAML authentication is done.

  **Note:** If user resizes the Embedded browser window, size will remain same even if user reconnects to Pulse Client. Embedded browser window size will remain as pre-selected size which was set by the user for the first time, until user resizes it again.

  ![Figure 15 SAML Authentication with Embedded browser](image)

- The Pulse Client user sees an external browser (see Figure 16), if Enable embedded browser for authentication is disabled in “Pulse Client Connection Set Options” on page 136.

  ![Figure 16 SAML Authentication (External Browser)](image)
Single Logout

Single logout is a mechanism provided by SAML for logging out a particular user from all the sessions created by the identity provider.

Select this option if the system must receive and send a single logout request for the peer SAML identity provider. If you use the metadata option, the Single Logout Service URL setting can be completed by selecting the SLO service URL from the list. The list is populated by the identity provider entities defined in metadata files added to the System > Configuration > SAML page. The system sends Single Logout requests to this URL. In addition, if you use the metadata option, the Single Logout Response URL setting is completed based on your selection for Single Logout Service URL.

If the identity provider has left this setting empty in its metadata file, the system sends the Single Logout response to the SLO service URL. If you complete these settings manually, ask the SAML identity provider administrator for guidance. The Support Single Logout service for the identity provider must present a valid certificate.

Custom Sign-in Page in Embedded browser

To upload a custom sign-In page in Pulse Client, admin needs to perform the following steps:

1. Log into Pulse Connect Secure/Pulse Policy Secure as admin.
2. Go to Authentication > Signing-In > Sign-In Pages > Upload Custom Sign-In Pages.
3. Select the option Use Custom Page for the Pulse Desktop Client Logon.
4. Click **Browse** and select the custom sign-in page file and click **Upload Custom Pages**.
5. Go to **Signing In > Sign-In Policies > New Sign-In Policy** to create the new Sign-In policy.

6. Under Sign-in page, select the uploaded custom page from the drop-down box to associate custom Sign-In page with the Sign-In Policy.

Figure 19  Associating a Custom Sign-in Page with a Sign-in Policy - Pulse Policy Secure

Pulse Client can open a custom sign-In page in the following two ways:

- A Pulse Client user sees an embedded browser (see Figure 20) if **Enable embedded browser for authentication** is enabled.

  Pulse Client closes the embedded browser once the authentication is done.

  Whenever user logs into the custom sign-in URL from Pulse Client, embedded browser will be launched with custom sign-in pages uploaded into it.
L3 and Pulse SAM Coexistence

L3 and Pulse SAM coexistence (supported on Windows only) enables the user to establish Layer 3 connection to Pulse Connect Secure and Pulse SAM connection simultaneously (refer to Figure 21). This feature is available from 9.0R3 onwards.

To achieve, L3 and PSAM coexistence, Pulse Client should have minimum two Pulse Connect Secure connections, each for L3 and PSAM. Also, maximum three active user connections are allowed at once.

Limitation for L3 and Pulse SAM coexistence:

- At any given point, for any user only one L3 and one L4 is supported.
With L3 and PSAM coexistence, the way the packet is tunneled, depends on how the L3 and PSAM tunnel are configured. It can be done in following two ways:

Following are the 2 scenarios, where L3 and PSAM coexistence is supported.

**Scenario-1: PSAM is behind L3**

PCS1 has L3 tunnel configuration and PCS2 is behind PCS1.

If specific set of resources is not accessible on PCS1 server and needs to access from PCS2 server, which is accessible through PCS1 server, then additional authentication is needed to access PCS2 server. As access to PCS2 server is possible only after making connection to PCS1 server, it is the case of PSAM tunnel inside L3 tunnel.

**Scenario-2: L3 and PSAM are independent**

PCS1 has L3 tunnel configuration and PCS2 has Pulse SAM configuration.

L3 Connection for Pulse Connect Secure is established, split tunneling should be enabled and exclude the PCS2 IP from the split tunneling networks.

If single user needs to access two different set of resources available on PCS1 and PCS2, then one specific set of resources is under PCS1 and another set of resources is under PCS2.

As PCS1 and PCS2 are at different locations and user can not establish two L3 connections to access both set of resources on PCS1 and PCS2, so PSAM can provide the secure access to set of resources on PCS2.

**Note:** L3 based FQDN Split Tunneling feature with PSAM coexistence is not supported.

**HVCI Compatibility**

Pulse Client for Windows is compatible with Microsoft Windows 10 HVCI settings. Windows 10 HVCI settings are part of Windows Device Guard security features for mitigating cybersecurity threats. When HVCI is enabled, Windows OS performs code integrity checks and allows only secured applications. Pulse Client for Windows is compatible with these settings which would help customers adopt the latest security features of Windows.

**Pulse SAM IPv6 Support**

Pulse SAM IPv6 support is available for Windows 8.1 and later.

Internet Protocol Version 6 (IPv6) is the protocol designed to succeed Internet Protocol Version 4 (IPv4). From 9.1R1 release onwards, Pulse SAM (PSAM) will support IPv6 Pulse SAM tunneling along with IPv4 Pulse SAM tunneling with the help of new option for internet traffic filtering, Windows Filtering Platform (WFP) driver.

WFP driver supports both IPv6 and IPv4, however TDI driver supports only IPv4. WFP driver allows the user to provide a deeper inspection and control of packets by modifying or examining TCP/IP traffic at any TCP/IP stack layer.

Administrator can switch from WFP driver (supporting both IPv6 and IPv4) to classic TDI driver (supporting IPv4 only) with fallback mechanism, in case of any issue due to WFP driver installation.

Following are the steps to switch from WFP to TDI:
1. Go to **Users > User Role**.

2. Select the role.

3. Go to **SAM > Options**. The screen in Figure 22 appears.

4. Select **Enable fail-over to TDI for Pulse SAM connection**.

**Benefits**

Following are the benefits of this feature:

- PSAM will be able to filter the traffic from Windows 10 and Windows 8.1 Metro Mode Applications.
- PSAM will be able to filter the traffic from Internet Explorer 11 with Enhanced Protected mode.
- PSAM will support Dual Stack (both IPv6 and IPv4).

**Deployment Scenarios**

The following table summarizes the IPv6 in IPv6, IPv4 in IPv6 and IPv6 in IPv4 scenarios:
Table 6  Deployment Scenarios

<table>
<thead>
<tr>
<th>PDC</th>
<th>Endpoint</th>
<th>PCS External Interface</th>
<th>PCS Internal Interface</th>
<th>Tunnel</th>
<th>Description of the Connection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dual Stack or IPv6 only</td>
<td>Dual Stack (IPv6 and IPv4) or IPv6 only</td>
<td>IPv6</td>
<td>Dual Stack or IPv6 only</td>
<td>IPv6-in-IPv6</td>
<td>IPv6 resource on IPv6 PSAM session.</td>
</tr>
<tr>
<td>Dual Stack or IPv6 only</td>
<td>Dual Stack (IPv6 and IPv4) or IPv6 only</td>
<td>IPv6</td>
<td>IPv4</td>
<td>IPv4-in-IPv6</td>
<td>IPv4 resource on IPv6 PSAM session.</td>
</tr>
<tr>
<td>Dual Stack or IPv4 only</td>
<td>Dual Stack (IPv6 and IPv4) or IPv6 only</td>
<td>IPv4</td>
<td>Dual Stack or IPv4</td>
<td>IPv6-in-IPv4</td>
<td>IPv6 resource on IPv4 PSAM session.</td>
</tr>
</tbody>
</table>

**Note:** Pulse Client 9.0R1 Pulse SAM connection fails with Pulse Connect Secure 9.1R1 version. For more details, refer to 9.1R1 Unified Pulse Secure Client Release Notes document on the Pulse Secure website (www.pulsesecure.net).

**Location Awareness**

The location awareness feature enables you to define connections that are activated automatically based on the location of the endpoint. Pulse Client determines the location of the endpoint by evaluating rules that you define. For example, you can define rules to enable Pulse Client to automatically establish a secure tunnel to the corporate network through Pulse Connect Secure when the user is at home, and to establish a Pulse Policy Secure connection when the user is in the office and connected to the corporate network over the LAN. Pulse Client does not re-establish a VPN tunnel when the endpoint re-enters the trusted/corporate network. Location awareness rules are based on the client’s IP address and network interface information.

**Centralized Pulse Client Configuration Management**

Centralized configuration management is a key feature of Pulse Client. Pulse Client connection sets (the configurations that define how and when a Pulse Client connects), are bound to a particular Pulse Secure server. The binding server is the one that provides the initial configuration to the Pulse Client. For example, if you create a Pulse Client connection set on Server A, and then distribute those connections to endpoints, those clients are bound to Server A.

A bound client is managed by its particular Pulse Secure server. The Pulse Secure administrator defines Pulse Client connections and software components that are installed on the endpoint. When Pulse Client connects to the Pulse Secure server that is managing it, the server automatically provisions configuration and software component updates. The administrator can permit the user to add, remove, and modify connections. The administrator can also allow dynamic connections (connections that are added by Pulse Secure servers when the user logs into the server using a browser). A dynamic connection enables a bound client to add connections from Pulse Secure servers other than the one the client is bound to. Dynamic connections are created as manual rather than automatic connections, which means that they are run only when the user initiates the connection or the user browses to a Pulse Secure server and launches Pulse Client from the server's Web interface. Dynamic connections create the connection with the minimum configuration required to make the connection, which means that the URL used to install or launch Pulse Client from the Pulse Secure server’s Web interface is used as the Connection URL and connection name. Binding Pulse Clients to a
particular server ensures that the client does not receive different configurations when it accesses other Pulse Secure servers. A bound endpoint receives connection set options and connections from its binding server, but it can have its Pulse Client software upgraded from any Pulse Secure server that has the automatic upgrade option enabled. (SRX gateways do not support Pulse Client software updates.)

**Note:** Pulse Client can be bound to only one Pulse Secure server connection set at a time. Pulse Client can receive updates and changes to that bound connection set from other Pulse Secure servers only if the connection set is exported from the Pulse Secure server and then imported to another Pulse Secure server.

Pulse Client does not need to be bound to a Pulse Secure server. An unbound client is managed by its user. If Pulse Client software is installed without any connections, the user must add connections manually. Dynamic connections can be added by visiting the Web portals of Pulse Secure servers. An unbound client does not accept configuration updates from any Pulse Secure server.

**Session Migration**

If you configure your access environment to support the Pulse Client session migration feature, users can log in once through a Pulse Secure server on the network, and then securely access additional Pulse Secure servers without needing re-authentication. For example, a user can connect from home through Pulse Connect Secure, and then arrive at work and connect through Pulse Policy Secure without having to log in again. Session migration also enables users to access different resources within the network without repeatedly providing credentials. IF-MAP Federation is required to enable session migration for users.

**Smart Connections - List of URLs**

Each Pulse Client connection that connects to Pulse Policy Secure or Pulse Connect Secure can be configured with a list of Pulse Secure servers. Pulse Client attempts to connect to each of the servers in the URL list until it succeeds. You can choose different modes to control the behavior of a Pulse Client connection that is starting from a disconnected state, start at the top of the list, start with the most recently connected URL, or choose randomly. The random option helps distribute the connection load across different Pulse Secure servers. If a Pulse Client connection that is already established gets disconnected, for example, the wireless connection is interrupted, Pulse Client always tries to connect to the most recently connected URL. If that connection fails, Pulse Client uses the server list. The Pulse Client user can also choose a connection from the list as shown in Figure 23.
Security Certificates

Users cannot add CA servers or manage the server list. Pulse Client handles certificates in the same way that a browser handles certificates. If the Pulse Client dynamic certificate trust option is enabled for a connection, the user can accept or reject the certificate that is presented if it is not from a CA that is defined in the endpoint’s certificate store.

Compliance and Remediation

Pulse Client supports the Host Checker application to assess endpoint health and update critical software. Host Checker is a client-side agent that is based on Trusted Network Connect standards. You configure rules in Host Checker policies for Pulse Connect Secure and Pulse Policy Secure to specify the minimum criteria for the security compliance of endpoints that are allowed to enter the network. Endpoints that fail can be connected through a remediation role that provides limited access.

Host Checker can be deployed from a Pulse Secure server to Pulse Clients on Windows and macOS endpoints. It will be downloaded and run when a browser is used on a Windows or macOS endpoint to connect to the Pulse Secure server Web portal. You can use Host Checker policies at the realm or role level.

Note: Host Checker is not supported in the use case where the user employs a browser on the mobile device to connect to the Pulse Secure server Web portal.

For Windows and OS X clients, you can use Host Checker to perform the following:

- Virus signature monitoring
  
  You can configure Host Checker to monitor and verify that the virus signatures, operating systems, and software versions installed on client computers are up to date. You can configure automatic remediation for those endpoints that do not meet the specified criteria.
• Patch management information monitoring and patch deployment
  You can configure Host Checker policies that check for Windows endpoints’ operating system service
  pack, software version, or desktop application patch version compliance.

• Patch verification remediation options
  Pulse Client and Host Checker support endpoint remediation through Microsoft System Management
  Server or Microsoft System Center Configuration Manager (SMS/SCCM). With SMS/SCCM, Pulse Client
  triggers a preinstalled SMS/SCCM client to get patches from a pre-configured server.

• Endpoint configuration
  You can configure custom rules to allow Host Checker to check for third-party applications, files,
  process, ports, registry keys, and custom DLLs.

Pulse Mobile Client supports a set of Host Checker functions that vary from one OS to the next. For complete
information on Host Checker for mobile clients, see “Implementing Host Checker Policies for Pulse Mobile
Client for iOS Devices” on page 254, “Implementing Host Checker Policies for Pulse Mobile Client for
Android” on page 269, and “Host Checker for Pulse Mobile Client for Windows Phone” on page 293.

Two Factor Authentication

Pulse Client supports RSA SecurID authentication through soft token, hard token, and smart card
authenticators. The SecurID software (RSA client 4.1 and later) must already be installed on the client machine.

Captive Portal Detection

Public WiFi locations often deploy a captive portal that requires the user to enter authentication information
or to accept terms of service before network access is granted. Pulse Client detects the presence of captive
portals and does not initiate a connection to a Pulse Connect Secure or Policy Secure server until internet
access is granted. Pulse Client displays appropriate status information to enable the user to establish the
portal and network connections.

Captive portal detection notes:

• Captive portal detection is supported on Pulse Client for both Windows and Mac. Captive portal
detection is not supported on Windows In-Box Pulse Client or Pulse Secure Client for Mobile Devices.

• If Pulse Client connects through a proxy in Captive Portal scenario, the captive portal detection
algorithm is disabled and Pulse Client tries connecting directly to PCS.

• SRX connections do not support captive portal detection.

Pulse Collaboration Suite Integration

Pulse Collaboration Suite is accessible through the Pulse Client interface on Windows, macOS, and Linux
clients. Pulse Collaboration Suite enables users to schedule and attend secure online meetings. In meetings,
users can share their desktops and applications with one another over a secure connection. Meeting
attendees can collaborate by enabling remote control of their desktops and through text chatting.
Sign In Notifications
The notifications feature on Pulse Connect Secure and Pulse Policy Secure allows the network administrator to display notifications to Pulse Client users prior to the user logging in and after the user has already logged in. For example, you could display a legal statement or a message stating who is allowed to connect to the server before you display the Pulse Client credentials dialog. After the user has connected, you could display a message that notifies the user of scheduled network or server maintenance or of an upcoming company meeting.

Automatic Software Updates
After you deploy Pulse Client software to endpoints, software updates occur automatically. If you upgrade the Pulse Client configuration on the server, updated software components are pushed to a client the next time it connects. You can disable this automatic upgrade feature.

**Note:** The automatic update feature is supported on Pulse Connect Secure and Pulse Policy Secure servers only. SRX gateways do not support automatic Pulse Client software updates.

**Note:** If you configure Pulse Client to make 802.1X-based connections, a reboot might be required on Windows endpoints.

Pulse Client Customization and Rebranding
The Pulse Client customization tool (BrandPackager) enables you to customize the appearance of Pulse Client for Windows and Pulse Client for Apple OS X. You can add your own identity graphic to the Pulse Client splash screen, to the program interface, and to Windows credential provider tiles. Figure 24 shows graphic customizations applied to the Pulse Client for Windows. You can also customize error and informational message text, the text that appears in dialog boxes and on buttons, and make limited changes to Pulse Client online Help. For example, you might want to add your help desk phone number to Pulse Client error messages and the Pulse Client online Help.

BrandPackager is available for download from the Pulse Secure website (www.pulsesecure.net).
Pulse Client and Software Defined Perimeter (SDP)

Traditional network-based security (Network Defined Perimeter) architectures use firewalls on the network perimeter to limit access to public IP addresses. This exposes the network to a variety of network-based attacks.

Connectivity in a Software Defined Perimeter (SDP) system is based on a need-to-know model, in which mobile devices are verified and authorized before access to application infrastructure is granted. Application infrastructure cannot be detected remotely and has no visible DNS information or exposed IP addresses. This protects networked resources from many common network-based attacks.

Pulse Secure SDP uses PCS appliances which individually act as either an SDP controller or an SDP gateway. Mobile users of Pulse Client perform authentication on an SDP controller which runs an Authentication, Authorization and Accounting (AAA) Service. The SDP controller then enables direct communication between the user and the SDP gateways that protect the user's authorized resources and enables requested encryption. This does not require the general exposure of public IP addresses. It also separates the control plane and the data plane.

Pulse Secure SDP supports a number of network topologies, and can include both cloud-based and data center-based resources. For example:
Pulse Client Configuration Overview

You configure Pulse Client settings on the Pulse Secure server so that when users request authentication, they are assigned a role based on the role mappings and optional security profile that you create. Access to specific resources is permitted only for users and devices that provide the proper credentials for the realm, that are associated with the appropriate roles, and whose endpoints meet security restrictions. If a user attempts to connect to the network from an endpoint that does not comply with the security restrictions you have defined, the user cannot access the realm or role.

As you plan your Pulse Client configuration, be sure you know how you want to deploy Pulse Client. You can use one or more of the following Pulse Client deployment options:

- Use the defaults or make changes to the Pulse Client default component set and default connection set, and then download and distribute Pulse Client by having users log in to the gateway’s user Web portal and be assigned to a role. After the installation is complete, users have all the connections they need to access network resources.

- Create connections that an endpoint needs for connectivity and services, download the Pulse Client settings file (.pulsepreconfig), download default Pulse Client .msi installation program, and then run the .msi installation program by using an msiexec command with the settings file as an option. You can use the msiexec command to deploy Pulse Client using a standard software distribution process, such as SMS/SCCM.
• Distribute Pulse Client with no preconfiguration. You can download the default Pulse Client installation file (Mac or Win) from the device, and then distribute the file to endpoints using your organization’s standard software distribution methods. Because the installer does not contain preconfigured connections, users must define network connections manually. Or you can create dynamic connections on each access gateway. These connections are automatically downloaded to the installed Pulse Client when users provide their login credentials to the gateway’s user Web portal.

The following tasks summarize how to configure Pulse Client on the device:

• Create and assign user roles to control who can access different resources and applications on the network. If you are converting your access environment from agentless or a VPN Tunneling environment, you should create new roles that are specific for Pulse Client.

• Define security restrictions for endpoints with Host Checker policies.

• Define user realms to establish authentication domains. If you are converting your access environment from agentless or a NC environment, typically you can use your existing realms.

• Associate the roles with appropriate realms to define your access control hierarchy using role mapping.

• Define Pulse Client component sets, connection sets, and connections.

• Deploy Pulse Client to endpoints.

**Pulse Client Status Icons**

The Pulse Client interface (Windows and OS X) displays a system tray icon (Windows) or a menu bar icon (OS X) that indicates connection status, provides access to menu items that let the user connect and disconnect from networks and meetings, and enables quick access to the program interface. Only one icon is visible even when there are multiple connections. One icon provides the status for all connections by indicating the most important connection state information.
Table 7  Pulse Client Icon States (Windows Tray and OS X Menu Bar)

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![checkmark]</td>
<td>Connected.</td>
</tr>
<tr>
<td>![checkmark]</td>
<td>Connecting.</td>
</tr>
<tr>
<td>![checkmark]</td>
<td>Connected with limitations</td>
</tr>
<tr>
<td>![error]</td>
<td>Connection attempt failed.</td>
</tr>
<tr>
<td>![info]</td>
<td>Connection suspended.</td>
</tr>
</tbody>
</table>

Connected to the local network but no Internet access available. Public WiFi locations often deploy a captive portal that requires the user to enter authentication information or to accept terms of service before network access is granted. Pulse Client detects the presence of captive portals and does not initiate a connection to a Pulse Secure server until Internet access is granted.

Installation Requirements
For detailed information about supported platforms and installation requirements, see the Pulse Secure Supported Platforms Guide, available from the Pulse Secure website (www.pulsesecure.net).

Pulse Client Error Messages Overview
Pulse Client error and warning messages reside in message catalog files on the endpoint. Each message includes a short description that states the problem and a long description that provides more details and suggests actions the user can take to resolve the issue.

You can edit Pulse Client messages by using the optional Pulse Client branding tool, BrandPackager. See “Editing Pulse Client Messages” on page 93 for more information.

All message catalog files are localized. The filename indicates the language. For example, MessageCatalogConnMgr_EN.txt is the English-language version of the file. The following filename conventions indicate the language:

- DE-German
- EN-English
- ES-Spanish
- FR-French
- IT-Italian
- JA-Japanese
Accessing Pulse Client Error Messages on macOS Endpoints

Pulse Client error and warning messages reside in message catalog files on the OS X endpoint. Each message includes a short description that states the problem and a long description that provides more details and suggests actions to resolve the issue.

You can edit Pulse Client messages by using the optional Pulse Client branding tool, BrandPackager. See “Editing Pulse Client Messages” on page 93 for more information.

All message catalog files are localized. The filename indicates the language. For example, MessageCatalogPulseUI_EN.txt is the English-language version of the file. The following filename conventions indicate the language:

- DE-German
- EN-English
- ES-Spanish
- FR-French
- IT-Italian
- JA-Japanese
- KO-Korean
- PL-Polish
- ZH-Chinese (Traditional)
- ZH-CN-Chinese (Simplified)

To view Pulse Client catalog files on macOS endpoint, use Finder to display the package contents of the Pulse Client application.

Pulse Client Log Files

Pulse Client writes information to log files on Windows and Apple OS X endpoints. If you need to investigate a problem with connectivity on a Pulse Client endpoint, you can instruct the user to save the client logs and e-mail them to you.

The user saves logging information by opening Pulse Client and then clicking File > Logs > Save As. All relevant log files are added to a single file, LogsAndDiagnostics.zip. The user saves the .zip file and then makes it available to you.

Pulse Client maintains its own log files on all supported platforms. On Windows, Pulse Client also logs its major operational events into Windows Event Log. Network administrators can review the Pulse Client event log to help troubleshoot problems. Table 8 lists the Pulse Client messages that can appear in the Windows event log.
To view the Pulse Client messages:

1. Open the Windows Event Viewer.
2. Click **Applications and Services > Pulse Secure > Operational.**
<table>
<thead>
<tr>
<th>ID</th>
<th>Level</th>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>600</td>
<td>error</td>
<td>The connection &lt;ID&gt; failed authentication: Error &lt;ID&gt;.</td>
<td>802.1X EAP authentication failure.</td>
</tr>
<tr>
<td>601</td>
<td>informational</td>
<td>User has canceled authentication of the connection &lt;ID&gt;.</td>
<td>The user canceled 802.1X EAP authentication.</td>
</tr>
<tr>
<td>602</td>
<td>error</td>
<td>Failure writing wireless LAN profile for connection &lt;ID&gt; Error &lt;ID&gt;: Reason &lt;ID&gt;: Profile: &lt;ID&gt;.</td>
<td>A failure occurred while a wireless LAN profile was being created or modified.</td>
</tr>
<tr>
<td>603</td>
<td>error</td>
<td>Failure writing wireless LAN profile for connection &lt;ID&gt; Error &lt;ID&gt;.</td>
<td>A failure occurred while a wireless LAN profile was being deleted.</td>
</tr>
<tr>
<td>604</td>
<td>error</td>
<td>Failure writing wired LAN profile for connection &lt;ID&gt; Error &lt;ID&gt;: Profile: &lt;ID&gt;.</td>
<td>A failure occurred while a wired LAN profile was being created or modified.</td>
</tr>
<tr>
<td>605</td>
<td>error</td>
<td>Failure writing wired LAN profile for connection &lt;ID&gt; Error &lt;ID&gt;.</td>
<td>A failure while a wired LAN profile was being deleted.</td>
</tr>
<tr>
<td>500</td>
<td>informational</td>
<td>Pulse servicing has completed successfully. All components are up to date.</td>
<td>Pulse Client servicing was successful.</td>
</tr>
<tr>
<td>501</td>
<td>informational</td>
<td>Pulse servicing has completed successfully. All components are up to date.</td>
<td>Servicing was requested but all components were up to date.</td>
</tr>
<tr>
<td>502</td>
<td>error</td>
<td>Pulse servicing has failed. Failure summary:</td>
<td>Pulse Client servicing failed.</td>
</tr>
<tr>
<td>100</td>
<td>informational</td>
<td>User requested connection &lt;ID&gt; to start.</td>
<td>The user initiated a connect request.</td>
</tr>
<tr>
<td>101</td>
<td>informational</td>
<td>User requested connection &lt;ID&gt; to stop.</td>
<td>The user initiated a disconnect request.</td>
</tr>
<tr>
<td>102</td>
<td>informational</td>
<td>Connection &lt;ID&gt; is starting because its policy requirements have been met. Connection Policy: &lt;ID&gt;.</td>
<td>A connection was started because of a policy evaluation.</td>
</tr>
<tr>
<td>103</td>
<td>informational</td>
<td>Connection &lt;ID&gt;) is stopping because of its policy requirements. Connection Policy: &lt;ID&gt;.</td>
<td>A connection was stopped because of a policy evaluation.</td>
</tr>
<tr>
<td>104</td>
<td>informational</td>
<td>Connection &lt;ID&gt; is stopping because of transition to context &lt;ID&gt;.</td>
<td>The machine-to-user connection was disconnected to transition to another identity.</td>
</tr>
<tr>
<td>105</td>
<td>informational</td>
<td>Connection &lt;ID&gt; is starting because of transition to context &lt;ID&gt;.</td>
<td>The machine-to-user connection was connected as part of the transition to another identity.</td>
</tr>
<tr>
<td>106</td>
<td>informational</td>
<td>Connection &lt;ID&gt; is disconnected due to computer suspend.</td>
<td>The connection to Pulse Connect Secure was disconnected because the computer is being suspended.</td>
</tr>
<tr>
<td>ID</td>
<td>Level</td>
<td>Message</td>
<td>Description</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>107</td>
<td>informational</td>
<td>Connection &lt;ID&gt; is disconnected due to login error.</td>
<td>A credential provider connection was disconnected because of a login error.</td>
</tr>
<tr>
<td>108</td>
<td>informational</td>
<td>Connection &lt;ID&gt; is disconnected because it was modified.</td>
<td>A connection was disconnected because it was modified.</td>
</tr>
<tr>
<td>109</td>
<td>informational</td>
<td>User requested connection &lt;ID&gt; to suspend.</td>
<td>The user initiated a suspend request.</td>
</tr>
<tr>
<td>110</td>
<td>informational</td>
<td>User requested connection &lt;ID&gt; to resume.</td>
<td>The user initiated a resume request.</td>
</tr>
<tr>
<td>1</td>
<td>informational</td>
<td>The Pulse Secure service version &lt;ID&gt; has successfully started.</td>
<td>The Pulse Client service started.</td>
</tr>
<tr>
<td>2</td>
<td>informational</td>
<td>The Pulse Secure service has stopped.</td>
<td>The Pulse Client service stopped.</td>
</tr>
<tr>
<td>200</td>
<td>error</td>
<td>No connections matching URL &lt;ID&gt; were found in Pulse database. Request to start a connection from the browser has failed.</td>
<td>Pulse Client failed to resume a connection from the browser.</td>
</tr>
<tr>
<td>400</td>
<td>error</td>
<td>The host check for connection &lt;ID&gt; has failed. Failed policies: &lt;ID&gt;.</td>
<td>Host Checker failed one or more policies.</td>
</tr>
<tr>
<td>300</td>
<td>informational</td>
<td>The connection &lt;ID&gt; was established successfully through web-proxy &lt;ID&gt;.</td>
<td>Pulse Client established a connection to Pulse Connect Secure or Pulse Policy Secure through a Web proxy.</td>
</tr>
<tr>
<td>301</td>
<td>informational</td>
<td>The connection &lt;ID&gt; was established successfully to address &lt;ID&gt;.</td>
<td>Pulse Client established a direct (nonproxy) connection to Pulse Connect Secure or Pulse Policy Secure.</td>
</tr>
<tr>
<td>302</td>
<td>informational</td>
<td>The connection &lt;ID&gt; was disconnected.</td>
<td>The Pulse Client connection was disconnected from the Pulse Secure server.</td>
</tr>
<tr>
<td>303</td>
<td>error</td>
<td>The connection &lt;ID&gt; encountered an error: &lt;ID&gt; Peer address: &lt;ID&gt;.</td>
<td>A connection encountered an error.</td>
</tr>
<tr>
<td>304</td>
<td>error</td>
<td>The connection &lt;ID&gt; was disconnected due to change in routing table. Interface address changed from &lt;ID&gt; to &lt;ID&gt;.</td>
<td>Pulse Client detected a change in the route to the Pulse Secure server.</td>
</tr>
<tr>
<td>305</td>
<td>informational</td>
<td>VPN tunnel transport for connection &lt;ID&gt; switched from ESP to SSL mode due to missing ESP heartbeat.</td>
<td>ESP to SSL fallback occurred because of missing ESP heartbeats.</td>
</tr>
<tr>
<td>306</td>
<td>informational</td>
<td>VPN tunnel for connection &lt;ID&gt; is switched to ESP mode.</td>
<td>Tunnel transport switched to ESP mode.</td>
</tr>
<tr>
<td>307</td>
<td>error</td>
<td>The connection &lt;ID&gt; encountered an error: System error: &lt;ID&gt; Peer address: &lt;ID&gt;.</td>
<td>The Pulse Client connection failed because of a system error.</td>
</tr>
<tr>
<td>308</td>
<td>error</td>
<td>The server disconnected connection &lt;ID&gt; Reason &lt;ID&gt;: Peer address: &lt;ID&gt;.</td>
<td>The server disconnected a connection.</td>
</tr>
</tbody>
</table>
Deleting Pulse Client Log Files

**Note:** Pulse Secure recommends that you do not delete Pulse Client log files.

Pulse Client controls log file size automatically. When a current log file reaches 10MB, a new one is created and the oldest log file is deleted. If you need to delete Pulse Client log files, do not delete the file without first moving it to the Recycle Bin or renaming it.

To safely delete Pulse Client log files on a Windows endpoint:

1. Use a command line or Windows Explorer to locate and delete `debuglog.log` and, optionally, `debuglog.log.old`. When prompted if you want to move the file to the Recycle Bin, answer Yes. Do not press Shift+Delete, which permanently deletes a file without moving it to the Recycle bin.

   The file location varies depending on which version of Windows the endpoint is running. For example, the following path is valid for a Windows 8.1 Enterprise 64-bit endpoint: `C:\ProgramData\Pulse Secure\Logging`.

2. Empty the Recycle Bin.

   Alternatively, you could first rename `debuglog.log` and then delete it. After you delete the log file, Pulse Client creates a new one. However, that operation might take some time depending on the activities of Pulse Client.

Uploading Pulse Client Log Files

The Pulse Client for Windows makes it easy to transmit diagnostic log bundles to PCS gateways for analysis by system administrators. To send a log bundle to the PCS, when a VPN connection is active, run the following from the desktop client user interface: **File > Logs > Upload**.

![Uploading Pulse Client Log Files](image)

The user must select the server to send the logs to. A dialog will appear that shows the progress of the upload.

Note that a system administrator must enable this feature on the server side before an end user can upload log files to the Pulse Secure gateway. To do this, the system administrator must launch the Pulse Secure server administrative console and navigate to **Users > Roles > General > Session Options > Enable Upload Logs**.
The admin must check the "Enable Upload Logs" checkbox, as shown below:

Figure 27   Enable Upload Logs

The admin must also enable which clients can send log files by traversing the following menus in the admin console and clicking on Pulse Client: System > Log/Monitoring > Client-Side Log > Settings

Once this work is done, the system administrator can view uploaded logs in the administrative console here: System > Log/Monitoring > Client-Side Log > Uploaded Logs

Figure 28   Viewing Upload logs

Migrating from Odyssey Access Client to Pulse Client

Odyssey Access Client® (OAC) is 802.1X network access client software that supports the Extensible Authentication Protocol (EAP) for secure wireless LAN access. Together with an 802.1X-compatible authentication server, OAC secures WLAN communications. OAC also serves as a client for enterprises that are deploying identity-based (wired 802.1X) networking. OAC provides wireless access to enterprise networks, home Wi-Fi networks, and public hotspots.

Pulse Client is an extensible multiservice network client that supports integrated connectivity and secure location-aware network access. Pulse Client simplifies the user experience by letting the network administrator configure, deploy, and control Pulse Client software and the Pulse Client connection configurations that reside on the endpoint. Pulse Client can provide 802.1X authentication and Layer 3 access services.
Like OAC, Pulse Client software is bundled with Pulse Policy Secure software. However, there are significant differences between OAC and Pulse Client and you should be aware of these differences when you plan a migration from OAC to Pulse Client. The following list includes planning considerations and best-practices for a migration project. See the related topics list for details about the Pulse Client configuration tasks.

- The 802.1X communication protocol that you use with OAC might need to be changed to support Pulse Client. OAC supports the full range of 802.1X protocols; Pulse Client supports only EAP-TTLS/EAP-JUAC. See “Comparing Odyssey Access Client and Pulse Secure Desktop Client” on page 100, which lists the 802.1X protocols supported by OAC and Pulse Client.

- One common migration practice is to create new sign-in policies, user realms, and user roles for Pulse Client, and then control the cut-over to Pulse Client by enabling Pulse Client sign-in policies and disabling OAC sign-in policies. The new policies, realms, and roles can be clones of the existing OAC policies, realms, and roles as a starting point. However, Pulse Client has more robust connection decision capabilities so you will probably want to edit your Pulse Client roles to take advantage of the Pulse Client capabilities. For example, you can replace both OAC and Network Connect with Pulse Client and use one client for authenticated LAN access and secure SSL VPN access. Location awareness rules allow Pulse Client to detect the network environment and choose a network connection based on current location.

- How many OAC configuration do you use? You need a Pulse Client configuration for each of the OAC configurations you currently use. A Pulse Client access configuration is called a connection. It comprises properties that define how, when, and where a connection is established with a Pulse Secure gateway. When you create the Pulse Client connections that you distribute to Pulse Clients, you configure how the connection can be established. Pulse Client connections support machine authentication and credential provider authentication. Figure 29 shows an instance of Pulse Client for Windows that includes multiple connections.

Figure 29 Pulse Client Interface (Windows Version)
• Odyssey Access Client is a wireless supplicant. Pulse Client, by design, is not a wireless supplicant. Pulse Client uses the underlying wireless supplicant on the endpoint, which is typically provided by the endpoint’s OS X or Windows operating system. When you migrate to Pulse Client and uninstall OAC, you remove the OAC wireless supplicant and the endpoint falls back to using wireless connectivity provided by the OS. You define 802.1X authentication connections for Pulse Client to enable authenticated 802.1X connectivity in the enterprise network. Any custom network configurations that users added to their local OAC configuration are lost when OAC is removed. For example, if a user added connection information to connect to a home wireless network, the user will need to redefine that connection in the endpoint's wireless supplicant. A best practice is to mention this needed configuration to users as part of the Pulse Client roll-out. In OAC, network auto-scan lists are defined on the client. With Pulse Client, you can define an auto-scan list as part of an 802.1X connection that is pushed to Pulse Client.

• Do you use wireless suppression in your OAC environment? Wireless suppression disables wireless connections as long as the client has a wired network connection. You enable wireless suppression as part of a Pulse Client connection set. Pulse Client connection set properties define the decision process that Pulse Client uses to establish network connections.

• If you are using OAC FIPS Edition, you need to deploy Pulse Client 5.0 or later to support the same level of FIPS compliance that is supported by OAC.

• Do you allow users to modify configuration settings after you deploy them in your OAC environment? When you create a Pulse Client connection, you can define whether users can override the connection decision that has been defined by the Pulse Secure administrator as part of the Pulse Client connection. You can also disable the user's ability to create new connections. Connections created by users are manual connections, that is, the connection is not tried unless the user opens Pulse Client and selects it.

• Do you allow OAC users to add, remove, or modify trusted servers and certificates? Pulse Client does not expose this functionality to users. Pulse Client handles certificates in the same fashion as a browser. When you define a Pulse Client connection you can allow users to choose to accept an unverified certificate, which allows users to connect to servers that use a self-signed certificate.

**Migrating from Network Connect to Pulse Client**

Pulse Client and Network Connect (NC) can run at the same time on an endpoint.

**Note:** The Pulse Client installation program checks for NC. If the installation program finds NC Release 6.3 or later, the Pulse Client installation proceeds. If NC is not at least Release 6.3, the program displays a message telling the user to upgrade NC. For detailed information about supported platforms and installation requirements, see the *Pulse Secure Supported Platforms Guide*, available from the Pulse Secure website (www.pulsesecure.net).

On endpoints that connect to Pulse Connect Secure, if Pulse Client is running on the Windows main desktop, you cannot launch Pulse Client within Secure Virtual Workspace (SVW). SVW is not supported with Pulse Client.

**Note:** SVW is not supported by Pulse Policy Secure 5.1 and later and Pulse Connect Secure 8.1 and later. If a Pulse Secure server has SVW policies configured, those policies are removed during the upgrade.
Predictable Pulse Secure Server Hostname Resolution with IPv6

When connecting to a Pulse Secure server, Pulse Client uses the services of the endpoint operating system to resolve the hostname to an IP address. If a Pulse Secure server hostname resolves to both IPv4 and IPv6 addresses, an IPv4 or an IPv6 address is presented to Pulse Client as the preferred IP address. The behavior depends on the operating system and how it is configured. For example, Windows 8.1 adheres to IETF standards that define how to establish the default address selection for IPv6. macOS 10.6 does not support that standard. Additionally, Windows 8.1 default settings can be changed by netsh commands, so RFC compliance can be modified on the endpoint. For these and other reasons, it is difficult to predict which Pulse Secure server IP address would get resolved to on a given client machine.

For predictable hostname resolution, we recommend that you use different Pulse Secure server hostnames for IPv6 and IPv4 addresses. For example, configure myserver1.mycompany.com for IPv4 addresses and myserver1-v6.mycompany.com for IPv6 addresses. The Pulse Secure server administrator can publish myserver1-v6.mycompany.com to the Pulse Client users who are expected to connect over IPv6, and others will continue using myserver1.mycompany.com.
Configuring Pulse Secure Desktop Client on SRX Series Gateways

• Pulse Client and SRX Series Gateways ............................................. 46
• Pulse Client and Dynamic VPN Configuration Overview ..................... 47

Pulse Client and SRX Series Gateways

The dynamic virtual private network (VPN) feature of SRX Series gateways simplifies remote access by enabling users to establish Internet Protocol Security (IPsec) VPN tunnels without having to manually configure VPN settings on their endpoints. Pulse Client for Windows and Pulse Client for Mac support dynamic VPN connectivity to SRX Series gateways. The VPN settings are part of a Pulse Client SRX connection. Depending on the version of Junos OS on the SRX gateway, you might be able to deploy Pulse Client to endpoints from the SRX Series gateway through a Web portal. An endpoint accesses the SRX Web portal and, after the user is authenticated, Pulse Client is downloaded and installed. The default installation includes a Pulse Client connection to the SRX Series gateway. Alternatively, you can create and deploy SRX connections from Pulse Policy Secure and Pulse Connect Secure. For details on the Junos OS versions that are able to deploy Pulse Client, see the Pulse Secure Supported Platform Guide available from the Pulse Secure website (www.pulsesecure.net).

To configure a firewall access environment for Pulse Client, you must configure the VPN settings on the SRX Series gateway and create and deploy an SRX connection on the Pulse Client.

Note: Pulse Client for mobile devices can access Pulse Connect Secure only.

For SRX Series gateways that cannot deploy Pulse Client software, you have the following configuration and deployment options:

• In an environment that includes Pulse Connect Secure and Pulse Policy Secure, create connections of the type SRX with a target address of your SRX Series Services gateway. Users could then install the Pulse Client software and the connection configurations by logging in to the Web portal of the Pulse Connect Secure or Pulse Policy Secure and being assigned to a role that installs Pulse Client. After the installation, the endpoint has the Pulse Client software and the connection information required to connect to the SRX Series Services gateways.

• Install the default Pulse Client software package, and then have users create new connections that point to the SRX Series gateway.

SRX Series gateways supported an earlier access client called Juniper Networks Access Manager. You must uninstall Access Manager before you deploy Pulse Client to endpoints. The Pulse Client installation program checks for Access Manager. If Access Manager is present, the program displays a message instructing the user to uninstall Access Manager before installing Pulse Client.

Note: The automatic update feature is supported on Pulse Connect Secure and Pulse Policy Secure servers only. SRX gateways do not support automatic Pulse Client software updates.
Pulse Client and Dynamic VPN Configuration Overview

A dynamic VPN allows administrators to provide IPsec access for Windows endpoints to a Juniper Networks SRX gateway device while also providing a way to distribute the Dynamic VPN software to remote clients through the use of a Web portal.

The following procedure lists the tasks for configuring a dynamic VPN. For detailed information on these topics, see the Junos OS documentation.

1. Configure authentication and address assignment for the remote clients:
   1. Configure an XAuth profile to authenticate users and assign addresses. You can use local authentication or an external RADIUS server. Use the `profile` configuration statement at the `[edit access]` hierarchy level to configure the XAuth profile.

      To use the XAuth profile for Web authentication, use the `web-authentication` configuration statement at the `[edit access firewall-authentication]` hierarchy level.

   2. Assign IP addresses from a local address pool if local authentication is used. Use the `address-assignment pool` configuration statement at the `[edit access]` hierarchy level. You can specify a subnet or a range of IP addresses. Or you can specify IP addresses for DNS and WINS servers.

2. Configure the VPN tunnel:
   1. Configure the IKE policy. The mode must be aggressive. You can use basic, compatible, or standard proposal sets. Only preshared keys are supported for phase 1 authentication. Use the `policy` configuration statement at the `[edit security ike]` hierarchy level.

   2. Configure the IKE gateway. Either shared or group IKE IDs can be used. You can configure the maximum number of simultaneous connections to the gateway. Use the `gateway` configuration statement at the `[edit security ike]` hierarchy level.

   3. Configure the IPsec VPN. You can use basic, compatible, or standard proposal sets with the `policy` configuration statement at the `[edit security ipsec]` hierarchy level. Use the `vpn` configuration statement at the `[edit security ipsec]` hierarchy level to configure the IPsec gateway and policy.

   4. Configure a security policy to allow traffic from the remote clients to the IKE gateway. Use the `policy` configuration statement at the `[edit security policies from-zone <zone> to-zone <zone>]` hierarchy level.

   **Note:** The placement of this security policy is important. You must place it above more specific, non-VPN policies so that traffic that is intended to be sent over the VPN tunnel is processed correctly.

   5. Configure host inbound traffic to allow specific traffic to reach the device from systems that are connected to its interfaces. For example, IKE and HTTPS traffic must be allowed.

   6. (Optional) If the client address pool belongs to a subnet that is directly connected to the device, the device would need to respond to ARP requests to addresses in the pool from other devices in the same zone. Use the `proxy-arp` configuration statement at the `[edit security nat]` hierarchy level. Specify the interface that directly connects the subnet to the device and the addresses in the pool.

3. Associate the dynamic VPN with remote clients:
1. Specify the access profile for use with dynamic VPN. Use the **access-profile** configuration statement at the `[edit security dynamic-vpn]` hierarchy level.

2. Configure the clients who can use the dynamic VPN. Specify protected resources (traffic to the protected resource travels through the specified dynamic VPN tunnel and is therefore protected by the firewall's security policies) or exceptions to the protected resources list (traffic that does not travel through the dynamic VPN tunnel and is sent in clear text). These options control the routes that are pushed to the client when the tunnel is up, therefore controlling the traffic that is sent through the tunnel. Use the **clients** configuration statement at the `[edit security dynamic-vpn]` hierarchy level.
Session Migration

- Understanding Session Migration ............................................................... 49
- Task Summary: Configuring Session Migration .............................................. 52
- Configuring Session Migration for Pulse Client ........................................... 52
- Configuring an IF-MAP Federated Network for Session Migration .................. 53

Understanding Session Migration
This topic describes the session migration feature.

Session Migration Overview
When you enable session migration on two or more Pulse Secure servers, a Pulse Secure Desktop Client (Pulse Client) endpoint can be moved from one location to another and connect to a different Pulse Secure server without providing additional authentication. For example, a user can be connected from home through Pulse Connect Secure, and then arrive at work and connect to Pulse Policy Secure without being reauthenticated. If session migration is not enabled, Pulse Client users must be reauthenticated each time they attempt to access the network through a different Pulse Secure server.

Sessions can be migrated between Pulse Policy Secure and Pulse Connect Secure servers that are in the same IF-MAP federated network: using either the same IF-MAP server, or using IF-MAP servers that are replicas of one another.

The servers must be in the same authentication group. Authentication groups are configured through authentication realms. An authentication group is a string that you define for common usage. You can use authentication groups to group together realms with similar authentication methods. Such as, one authentication group for SecurID authentication, another authentication group for AD. A single gateway can belong to more than one authentication group, with a different authentication group per realm.

The Pulse Secure server to which a user authenticates publishes session information to the IF-MAP server. Other IF-MAP clients in the federated network can use the information to permit access without additional authentication to users.

When a user session is migrated to another Pulse Secure server, the new session information is pushed to the IF-MAP server. The IF-MAP server notifies the authenticating server, and information about the session that existed on the original server is removed leaving only session information about the current authenticating server on the IF-MAP server. The authenticating server removes information about the session from its local session table.

When a session is migrated, realm role-mapping rules determine user access capabilities. You can import user attributes when a session is migrated, or you can configure a dedicated directory server to look up attributes for migrated user sessions. To ensure that session migration retains user sessions, configure a limited access remediation role that does not require a Host Checker policy. This role is necessary because the Host Checker timeout can be exceeded if an endpoint is in hibernation or asleep. With the new remediation role, the user’s session is maintained.
If additional Host Checker policies are configured on a role or realm to which a migrated session applies, the policies are performed before allowing the user to access the role or realm. Administrators of different Pulse Secure servers should ensure that Host Checker policies are appropriately configured for endpoint compatibility.

**Figure 30** illustrates the task flow for enabling session migration for Pulse Client.

**Figure 30  Requirements for Pulse Client Session Migration**

### Session Migration and Session Timeout

Session timeout on the authenticating server does not apply to a migrated session. Instead, session start time is applicable. The inbound server evaluates session timeout using the start time of the original session on the original server.

When a user reboots an endpoint for which session migration is enabled, the session is retained for a short time on the server. For sessions on the Pulse Policy Secure, sessions are retained until the heartbeat timeout expires. For Pulse Connect Secure sessions, the idle timeout determines how long the session is retained.

If an endpoint that is connected to a Pulse Policy Secure or Pulse Connect Secure is rebooted and the user does not sign out, when the endpoint is restarted and the user attempts to connect to the same access gateway, Pulse Client resumes the previous session without requesting user credentials if the previous session is still active.

### How Session Migration Works

Session migration uses IF-MAP Federation to coordinate between servers.
When a session is established, the authenticating gateway publishes the session information, including a session identifier, to the IF-MAP server. The session identifier is also communicated to the Pulse Client.

When Pulse Client connects to a migrating gateway in the same authentication group, Pulse Client sends the session identifier to the migrating gateway. The migrating gateway uses the session identifier to look up the session information in the IF-MAP server. If the session information is valid, the migrating gateway uses the session identifier to establish a local session for the endpoint that Pulse Client is running on.

The IF-MAP server notifies the authenticating gateway that the user session has migrated, and the authenticating gateway deletes the session information from the IF-MAP server.

### Session Migration and Session Lifetime

Session migration is designed to give users maximum flexibility and mobility. Users are no longer tied to the office. The workplace can travel with the user, and electronic chores such as online banking can come to work. Because of this flexibility, users might be away from their machines for long periods of time, allowing their active session to expire. Session migration requires users to have an active session on Pulse Policy Secure or Pulse Connect Secure.

You can adjust session lifetime to ensure that sessions do not time out while users are away from their machines. You adjust session lifetime on the gateway by selecting Users > User Roles > [Role Name] > General > Session Options in the admin console.

### Session Migration and Load Balancers

A Pulse Client that connects to a Pulse Secure server that is behind a load balancer will attempt to migrate the network connection if the connected server fails. The Pulse Secure servers must be federated and configured for session migration. For example, a load balancer that balances to 2 Pulse Secure servers (non-clustered) balances to Server1. If Server1 fails, the load balancer then balances to Server2. A Pulse Client that is connected to Server1 is migrated to Server2 without re-authentication.

### Authentication Server Support

The behavior of session migration depends to some extent on the authentication server on the inbound side. The following list provides a summary of authentication server support:

- **Local authentication server**: Migration succeeds if the username is valid on the local authentication server.
- **LDAP server**: Migration succeeds if the LDAP authentication server can resolve the username to a distinguished name (DN).
- **NIS server**: Migration succeeds if the NIS authentication server can find the username on the NIS server.
- **ACE server**: Migration always succeeds.
- **RADIUS server**: Migration always succeeds. If you select Lookup Attributes using Directory Server, no attributes are present in the user context data.
- **Active Directory**: Migration always succeeds. The Lookup Attributes using Directory Server option might not work, depending on your configuration.
• **Anonymous**: No support for migrating sessions because sessions are not authenticated.
• **Siteminder**: No support for migrating sessions because Siteminder SSO is used instead.
• **Certificate**: No support for migrating sessions because sessions are authenticated using certificates.
• **SAML**: No support for migrating sessions because SAML SSO is used instead.

**Note**: For local, NIS, and LDAP authentication servers, the inbound username must reflect an existing account.

---

**Task Summary: Configuring Session Migration**

To permit session migration for users with the Pulse Client, perform the following tasks:

1. Configure location awareness rules within a Pulse Client connection set to specify locations included in the scope of session migration for users. For example, configure location awareness rules for a corporate Pulse Policy Secure server connection and a Pulse Connect Secure server connection.
2. Configure an IF-MAP federated network, with the applicable Pulse Secure servers as IF-MAP Federation clients of the same IF-MAP Federation server.
3. Ensure that user entries are configured on the authentication server for each gateway.
4. Ensure that user roles are configured for all users on each gateway.
5. Define a remediation role with no Host Checker policies to allow user sessions to be maintained when an endpoint is sleeping or hibernating.
6. Configure role-mapping rules that permit users to access resources on each gateway.
7. Enable and configure session migration from the User Realms page of the admin console.
8. Distribute Pulse Client to users.

---

**Configuring Session Migration for Pulse Client**

**Note**: Ensure that all of the Pulse Policy Secure servers and Pulse Connect Secure servers for which you want to enable session migration are IF-MAP Federation clients of the same IF-MAP Federation server. Additionally, make sure that each gateway is configured according to the procedures outlined in this section.

To configure session migration:

1. In the admin console, select **Users > User Realms**.
2. Select an existing realm, or create a new realm.
3. On the General page, select the **Session Migration** check box. Additional options appear.
4. In the **Authentication Group** box, enter a string that is common to all of the gateways that provision session migration for users. The authentication group is used as an identifier.
5. Select for either the **Use Attributes from IF-MAP** option button or the **Lookup Attributes using Directory Server** option.
Configuring an IF-MAP Federated Network for Session Migration

To successfully deploy session migration, you configure a Pulse Policy Secure IF-MAP server, and you configure all of the connected Pulse Secure servers that users access as IF-MAP clients. A Pulse Connect Secure server cannot be an IF-MAP server.

To add clients, you must specify the IP address and the security mechanism and credentials for the client.

An IF-MAP server certificate must be installed on the IF-MAP server. The client verifies the server certificate when it connects to the server. The server certificate must be signed by a Certificate Authority (CA), the client must be configured to trust certificates signed by that CA, and the hostname in the server certificate must match the hostname in the IF-MAP URL on the client.

You must identify the IF-MAP server to each Pulse Secure server IF-MAP client. To add the server, you specify the IF-MAP URL of the server and how to authenticate to the server. Match the URL and security settings to equal those on the IF-MAP server to which the IF-MAP clients will connect.

To configure IF-MAP server settings on the Pulse Policy Secure server:

1. From the admin console select System > IF-MAP Federation > Overview.
2. Under Choose whether this Pulse Policy Secure server runs an IF-MAP Server, an IF-MAP client, or no IF-MAP, select the IF-MAP Server option button.
3. Click Save Changes.
4. From the admin console select System > IF-MAP Federation > This Server > Clients.
5. Under IF-MAP Client, enter a Name and an optional Description for this client.
   For example, enter the name CS-access1.corporate.com and the description Connect Secure 1.
6. Type one or more IP addresses of the client. If the client is multi-homed, for best results list all of its physical network interfaces. If the client is a Pulse Policy Secure server or Pulse Connect Secure cluster, list the internal and external network interfaces of all nodes. You must enter all of the IP addresses for all of the interfaces because equipment failures might cause traffic between the IF-MAP client and the IF-MAP server to be re-routed through a different network interface. Listing all of the IP addresses maximizes the probability that IF-MAP Federation still works in the event of a failure.
   For example, enter 172.16.100.105.
7. Under Authentication, select the Client Authentication Method: Basic or Certificate.
   1. If you select Basic, enter a Username and Password. The same information should be added to the IF-MAP server.
   2. If you select Certificate, choose which Certificate Authority (CA) to use to verify the certificate for this client. Optionally, specify certificate attributes or restrictions to require values for certain client certificate attributes.
8. Click Save Changes to save the IF-MAP Client instance on the IF-MAP server.
To configure IF-MAP client settings on the Pulse Secure server clients:

1. From the admin console select **System > IF-MAP Federation > Overview**.

2. In a Policy Secure server, under Choose whether this server runs an IF-MAP Server, an IF-MAP client, or no IF-MAP, select the **IF-MAP Client** option button. On a Pulse Connect Secure server, select **Enable IF-MAP Client** check box.

3. Type the server URL for IF-MAP Web service on the IF-MAP server. Append the server URL with /dana-ws/soap/dsifmap for all Pulse Secure IF-MAP servers.
   
   For example, https://access2.corporate.com/dana-ws/soap/dsifmap.

4. Select the client authentication method: **Basic** or **Certificate**.

5. If you select **Basic**, enter a username and password. This is the same as the information that was entered on the IF-MAP server.

6. If you select **Certificate**, select the device certificate to use.

   Ensure that the certificate of the CA that signed the IF-MAP server certificate is added from the **System > Configuration > Certificates > Trusted Server CA** page.

   The IF-MAP client validates the IF-MAP server certificate: if validation fails, the connection fails. Ensure that the hostname in the IF-MAP URL on the client machine matches the hostname of the server certificate on the IF-MAP server, and that the CA that signed the server certificate is configured as a trusted server CA on the IF-MAP client.

7. Click **Save Changes**.
Deploying Unified Pulse Secure Client

- Unified Pulse Secure Client Installation Overview ........................................ 55
- Adding a Configuration to a New Unified Pulse Secure Client Installation ........ 57
- Installing Unified Pulse Secure Client from the Web .................................... 61
- Launching Unified Pulse Secure Client from the Pulse Secure server Web Portal ..., 62
- Launching Unified Pulse Secure Client using URL ........................................... 63
- Installing Unified Pulse Secure Client on Windows Endpoints Using a Preconfiguration File . 69
- Installing Unified Pulse Secure Client on OS X Endpoints Using a Preconfiguration File ... 72
- Unified Pulse Secure Client Command-line Launcher ..................................... 74
- Using jamCommand to Import Pulse Secure Connections ............................... 77
- jamCommand Reference .............................................................................. 78

Unified Pulse Secure Client Installation Overview

This section describes how to deploy Unified Pulse Secure Client (Unified Pulse Secure Client) for Windows and Unified Pulse Secure Client for macOS client software from Pulse Policy Secure and Pulse Connect Secure platforms.

Pulse Policy Secure and Pulse Connect Secure include a default connection set and a default component set. These defaults enable you to deploy Unified Pulse Secure Client to users without creating new connection sets or component sets. The default settings for the client permit dynamic connections, install only the components required for the connection, and permit an automatic connection to Pulse Connect Secure or Pulse Policy Secure to which the endpoint connects.

In all deployment scenarios, you must have already configured authentication settings, realms, and roles.

You can deploy Unified Pulse Secure Client to endpoints from Pulse Connect Secure and Pulse Policy Secure in the following ways:

- **Web install**: With a Web install (also called a server-based installation), users log in to the Pulse Secure server’s Web portal and are assigned to a role that supports a Unified Pulse Secure Client installation. When a user clicks the link to run Unified Pulse Secure Client, the default installation program adds Unified Pulse Secure Client to the endpoint and adds the default component set and the default connection set. If you do not make any changes to the defaults, the endpoint receives a Unified Pulse Secure Client installation in which a connection to the Pulse Secure server is set to connect automatically. You can edit the default connection set to add connections of other Pulse Secure servers and change the default options.

**Note**: The exact mechanism used to launch and install a particular Unified Pulse Secure Client from a web browser depends on a number of factors, including:

- The Unified Pulse Secure Client (Windows/Mac desktop client, Network Connect, Host Checker, WSAM, Windows Terminal Services, Secure Meeting client) being launched/installed.
- The endpoint operating system type and version.
- The web browser type and version.
The security settings of the endpoint operating system and browser.

For a particular client/OS/browser combination, you may need to enable the appropriate technology on the endpoint device. For example, to launch the Unified Pulse Secure Client from Firefox on Windows, you will need to ensure that Java is enabled in Firefox on the end user's endpoint device. For more information, consult the "Adaptive Delivery" section of the Pulse Secure Supported Platforms Guide.

Note: A Web install is not compatible with the Pulse Secure rebranding tool, BrandPackager.

- **Preconfigured installer**: Create the connections that an endpoint needs for connectivity and services, download the settings file (.pulsepreconfig), and download default Unified Pulse Secure Client installation program. For Windows endpoints you run the Unified Pulse Secure Client installation program by using an msiexec command with the settings file as an option. For OS X endpoints, you run the default installer and then import the .pulsepreconfig file using a separate command.

- **Default installer**: You can download the default Unified Pulse Secure Client installation program and distribute it to endpoints using your local organization’s standard software distribution method (such as Microsoft SMS/SCCM). Unified Pulse Secure Client software is installed with all components and no connections. After users install a default Unified Pulse Secure Client installation, they can add new connections manually through Unified Pulse Secure Client user interface or by using a browser to access a Pulse Secure server's Web portal. For the latter, the Pulse Secure server’s dynamic connection is downloaded automatically and the new connection is added to Unified Pulse Secure Client’s connections list when the user starts Unified Pulse Secure Client by using the Pulse Secure server's Web portal interface. Dynamic connections are created as manual rather than automatic connections, which means that they are run only when the user initiates the connection or the user browses to a Pulse Secure server and launches Unified Pulse Secure Client from the server's Web interface.

If the Windows endpoints in your environment do not have admin privileges, you can use the Unified Pulse Secure Client Installer program, which is available on the admin console System Maintenance Installers page. The Unified Pulse Secure Client Installer allows users to download, install, upgrade, and run client applications without administrator privileges. In order to perform tasks that require administrator privileges, the Unified Pulse Secure Client Installer runs under the client's Local System account (a powerful account with full access to the system) and registers itself with Windows' Service Control Manager (SCM). An Active-X control or a Java applet running inside the user's Web browser communicates the details of the installation processes to be performed through a secure channel between the Pulse Secure server and the client system.

- Installing the Unified Pulse Secure Client Installer MSI package requires administrator rights to install onto your client systems. If you plan to use the EXE version, administrator rights are not needed as long as a previous version of the access service component (deployed through, for example, JIS, Unified Pulse Secure Client, and so forth) is already present. If policies are defined for your client with the group policy "Run only Allowed Windows Application", the following files must be allowed to run in the group policy. If not, client applications might not install.
  - dsmmf.exe
  - PulseCompMgrInstaller.exe
  - PulseSetupClient.exe
  - PulseSetupClientOCX.exe
  - PulseSetupXP.exe
  - uninstall.exe
  - x86.Microsoft.*.exe
• You should ensure that the Microsoft Windows Installer exists on the client system prior to installing the Unified Pulse Secure Client Installer.

• Your end-users’ client systems must contain either a valid and enabled Java Runtime Engine (JRE) or a current Pulse Connect Secure ActiveX control. If the client systems do not contain either of these software components, the users will be unable to connect to the gateway. If there is no JRE on your end-users' client systems, you should download an appropriate installer package from Maintenance > System > Installers. The service appears in the Windows Services (Local) list as Neoteris Setup Service. The service starts automatically on install and during client system start up.

Adding a Configuration to a New Unified Pulse Secure Client Installation

When you install Unified Pulse Secure Client for Windows or Unified Pulse Secure Client for macOS client on an endpoint using the default Unified Pulse Secure Client installation program, the endpoint has all the Unified Pulse Secure Client components it needs to connect to Pulse Secure servers. However, Unified Pulse Secure Client needs a configuration that identifies the Pulse Secure servers it can connect to, that is, the connections. Connection properties also define how the connections are to be started, manually, automatically, or according to location awareness rules, and how Unified Pulse Secure Client connections receive updates. These connection set properties are also called machine settings. Figure 95 shows the default Unified Pulse Secure Client connection set properties (machine settings) that are passed to Unified Pulse Secure Client as its configuration. Figure 96 shows the connection set properties as they appear in a Unified Pulse Secure Client preconfiguration file, which you can use to add the Unified Pulse Secure Client configuration when you install Unified Pulse Secure Client. The preconfiguration file also includes Unified Pulse Secure Client connections.
There are two methods for installing an initial configuration on a new Unified Pulse Secure Client:

- Use a Unified Pulse Secure Client preconfiguration file (.pulsepreconfig) when you install Unified Pulse Secure Client on endpoints using the default Unified Pulse Secure Client installer.

- Instruct users to open a browser and login to the Pulse Secure server Web portal where the Unified Pulse Secure Client configuration has been defined. After successful login, the user should start Unified Pulse Secure Client from the Web page. Or you can enable Auto-launch as a role option to have the Unified Pulse Secure Client installation begin automatically after login.
The first time Unified Pulse Secure Client connects to a server that offers a Unified Pulse Secure Client configuration, the configuration settings are installed on the client, and the client is bound to that server, which means that only that server can update the client's configuration. Any Pulse Secure server can update the Unified Pulse Secure Client software version if that feature is enabled, and any Pulse Secure server can add a connection to an existing Unified Pulse Secure Client configuration if the Dynamic connections option is enabled as part of the connection set on the binding server. Only the binding server can update Unified Pulse Secure Client's configuration.

If the Unified Pulse Secure Client configuration has Dynamic connections enabled, then connections from other Pulse Secure servers are automatically added to Unified Pulse Secure Client's connections list when the user connects to the other Pulse Secure server through that server's Web portal, and the user starts Unified Pulse Secure Client using the Pulse Secure server's Web portal interface. For example, a user has a Unified Pulse Secure Client configuration from PulseServerA (the binding server) and the Unified Pulse Secure Client configuration allows dynamic connections. If the user browses to PulseServerB and successfully authenticates through that server's Web portal and clicks the Unified Pulse Secure Client button, the server adds a PulseServerB connection to the Unified Pulse Secure Client configuration, and it appears in Unified Pulse Secure Client's connection list. This new connection is set to start manually so that it does not attempt to connect when the endpoint is restarted or conflict with the connections from the binding server. A dynamic connection is added to Unified Pulse Secure Client's connections list. However, the connection's target URL is Pulse Web server URL; it does not use the URL that is defined for the connection in the server's Unified Pulse Secure Client connection properties. In most cases, these URLs will be the same.

You can see a Unified Pulse Secure Client configuration by creating and viewing a pulsepreconfig file. (To create the file, go to the Unified Pulse Secure Client Component screen, select a component set, and then click the Download Pulse Configuration button.) The .pulsepreconfig file contains a section that defines the machine settings and separate sections for each Unified Pulse Secure Client connection deployed to the client, as shown in Figure 32.
The machine settings and each centrally configured connection include the server ID (server-id) of the binding server. When a user browses to a Pulse Secure server, the server can offer a new configuration, (that is, updates to the machine settings). If the server-id under machine settings matches, Unified Pulse Secure Client accepts the configuration update. If the server-id does not match, Unified Pulse Secure Client ignores the update.
Configuration files have a version number as well. When Unified Pulse Secure Client connects to its binding server, Unified Pulse Secure Client compares the version of its existing configuration to the version on the server. If the server version is later than the existing client version, the client configuration is updated. The update might add, change, or remove connections and change machine settings.

If you have several Pulse Secure servers and you want to provision the same Unified Pulse Secure Client configuration from all of the servers, the server ID of the Unified Pulse Secure Client configuration must be the same across all of the servers. To accomplish this, you create the configuration on one server, and then use the "push config" feature of the Pulse Secure server to push the configuration to the other Pulse Secure servers. This method ensures that the server ID of the configuration file is the same across all of the Pulse Secure servers so that clients can receive a configuration update from any of the Pulse Secure servers.

**Installing Unified Pulse Secure Client from the Web**

For a Web install, you direct users to the Web interface of the Pulse Secure server. After a successful login, a user is assigned to a role that includes an automatic download and installation of Unified Pulse Secure Client software.

**Note:** In order to install the Unified Pulse Secure Client from a web browser, you may need to enable certain browser plugins or other technologies on the endpoint device. For example, Java must be enabled on the endpoint device to install Unified Pulse Secure Client from Firefox, and either ActiveX or Java must be enabled to install Unified Pulse Secure Client from Internet Explorer.

Pulse Connect Secure 8.2r1 and Pulse Policy Secure 5.3r1 introduced a new web-installation option called "Pulse Secure Application Launcher" (PSAL). PSAL leverages "URL handler" functionality by invoking a custom URL in a manner that instructs the web browser to execute a program that launches/installs the appropriate Unified Pulse Secure Client. PSAL was created to address both the restrictions placed on Java on macOS and the deprecation of Java (and Active X) plug-ins in Google Chrome version 45 and the Microsoft Edge browser. To read more about PSAL, see the Pulse Secure Knowledge Center article “KB40102” (https://kb.pulsesecure.net).

For a full discussion of this subject, see the “Adaptive Delivery” section of the Pulse Secure Supported Platforms Guide.

The default Unified Pulse Secure Client installation settings includes minimal components, which includes the Host Checker component, and a connection to the Pulse Secure server. If you want a Web install that has customized settings, you can do any of the following:

- Edit the default connection set and add new connections. The default installer uses the default component set which includes the default connection set.
- Create a new connection set and edit the default component set to include the new connection set.
- Edit the role to specify a component set that includes the connections you want for the default installation.

**Note:** A Unified Pulse Secure Client installation causes a restart of active network connections on a Windows endpoint. When a user initiates a Unified Pulse Secure Client installation through a WAN connection to the Web interface of a Pulse Secure server, the user might need to log in to their service provider again to reestablish network connectivity. Users need to be aware of this issue before they begin the installation.
Launching Unified Pulse Secure Client from the Pulse Secure server Web Portal

One typical method of establishing a VPN connection is for users to browse to the Pulse Secure server's Web portal, login, and then launch Unified Pulse Secure Client from the Web page. (This method is common in environments that used the Network Connect client.)

The following items describe the Unified Pulse Secure Client connection behaviors:

- Unified Pulse Secure Client has been installed on the endpoint by using the default Unified Pulse Secure Client installer. The installed Unified Pulse Secure Client does not yet have any connections. The user browses to the Pulse Secure server, logs into the server, and then clicks the Unified Pulse Secure Client button on the Web portal page. The following action occurs:
  1. The default Unified Pulse Secure Client connection set is automatically deployed to the client.
  2. The connection that has a URL that matches the server URL is launched.

- Unified Pulse Secure Client has been installed on the endpoint and it has a connection from the Pulse Secure server. The user browses to the Pulse Secure server, logs into the server, and then clicks the Unified Pulse Secure Client button on the Web portal page. The following action occurs:
  1. The connection that has a URL that matches the server URL is launched.

- Unified Pulse Secure Client has been installed on the endpoint and it has a connection from two different Pulse Secure servers. The user browses to one of these Pulse Secure servers, logs into the server, and then clicks the Unified Pulse Secure Client button on the Web portal page. The following action occurs:
  1. Only the connection that has a URL that matches the server URL is launched.

- Unified Pulse Secure Client has been installed on the endpoint. It has a connection for one Pulse Secure server but the user browses to a different Pulse Secure server, logs into the server, and then clicks the Unified Pulse Secure Client button on the Web portal page. The following action occurs:
  1. A new dynamic connection is created on Unified Pulse Secure Client for this Pulse Secure server. (Note that the default connection on the server must be configured as a dynamic connection.) This new connection is a manual connection, that is, it does not start automatically when Unified Pulse Secure Client starts.
  2. The new connection for this Pulse Secure server is started based on matching URLs.

Usage Notes

The Web browser method of launching Unified Pulse Secure Client is affected by the following configuration issues:

- The Unified Pulse Secure Client connection URL and the server URL must be an exact match. Unified Pulse Secure Client does not perform reverse DNS lookup to find a match.
- Connections that have the connection property **Allow user to override connection policy** disabled cannot be launched from the browser even if URLs match.
Launching Unified Pulse Secure Client using URL

Launching Unified Pulse Secure Client using URL feature enables the user to launch the Unified Pulse Secure Client using the admin prescribed URL. This feature is supported for Windows only.

Administrator creates a web URL (in a prescribed format), and provides it to the user in the following ways:

- URL is placed in a web page in the form of a link and the address of the link is provided to the user.
- URL itself is provided to the user.

User clicks on the link or types the URL in the browser. Unified Pulse Secure Client gets launched and the connection is redirected to the gateway mentioned in the URL.

User receives a link or a URL which has been crafted by an administrator. Following is the format of the URL:

```
pulsesecureclient://connect?name=NAME&server=SERVERURL&userrealm=REALM&username=USER&store=TRUE
```

Table 9 lists the parameter and their description mentioned in URL:
Table 9 Parameters details of URL

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mandatory/Optional</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>pulsesecureclient</td>
<td>Mandatory</td>
<td>// URI scheme for URL launching.</td>
</tr>
<tr>
<td>connect</td>
<td>Mandatory</td>
<td>This parameter is an action item and establishes the connection.</td>
</tr>
<tr>
<td>name</td>
<td>Mandatory</td>
<td>This parameter is a unique parameter, which defines the name of the connection. This connection name is used to identify a specific connection. Connection name will be suffixed by (Auto Launch) for Unified Pulse Secure Client connection established through URL. Connection name will be displayed as Name(Auto Launch).</td>
</tr>
</tbody>
</table>
| server         | Mandatory          | This parameter defines the sign-in URL, to which Unified Pulse Secure Client should get connected. It can be any one of the following:  
  - FQDN  
  - IP address (IPv6 and IPv4)  
  - A Sign-in URL |
| userrealm      | Optional           | This parameter defines the user realm.                                |
| username       | Optional           | This parameter defines the username.                                 |
| store          | Optional           | If store value is "True", then the connection information gets saved in the connection store.  
  If store value is "False", then the connection information will not be saved in the connection store.  
  It provides the flexibility for the user to save the connection information for future purposes. |

Following is the scenario to understand the behaviour of this feature:

In this scenario, the user establishes a connection named "Pulse Connection" to the Pulse Connect Secure server with userrealm as "Users" and username as "test_user". Also, user wants to store the connection in Unified Pulse Secure Client for future references.

Administrator will craft the URL with the values mentioned in below table:
Table 10 Parameter details

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Pulse Connection</td>
</tr>
<tr>
<td>server</td>
<td><a href="https://pcssamlpcs.psecure.net/">https://pcssamlpcs.psecure.net/</a></td>
</tr>
<tr>
<td>userrealm</td>
<td>Users</td>
</tr>
<tr>
<td>username</td>
<td>test_user</td>
</tr>
<tr>
<td>store</td>
<td>true</td>
</tr>
</tbody>
</table>

1. User receives a link or the below mentioned URL which has been crafted by an administrator.

   pulsesecureclient://connect?name=Pulse Connection&server=https://pcssamlpcs.psecure.net/ &userrealm=Users&username=test_user&store=true

   Figure 33  Browser URL

2. Once the user opens the URL (in Edge browser), following screen appears:

   Figure 34  Switch apps

   A permission dialog box appears to get the confirmation from the user to launch Unified Pulse Secure Client application via URL.

3. User clicks Yes button and Unified Pulse Secure Client gets launched.

   A connection with the name specified in the URL (“Pulse Connection”) is added in the Unified Pulse Secure Client and following screen appears:
Note: Connection name will be displayed as "Pulse Connection(Auto Launch)". Connection name will be suffixed by (Auto Launch) for Unified Pulse Secure Client connection established through URL.

4. User enters the password and clicks the **Connect** button. Following screen appears:

Now, connection named Pulse Connection(Auto Launch) with provided values as mentioned in **Table 10** is established. Full connection name can be viewed in Edit window as shown in the following screen.
After this, Unified Pulse Secure Client is launched, and a connection named Pulse Connection(Auto Launch) is established. This connection is then established with username as test_user.

After successful connection establishment, if user decides to disconnect Pulse Connection(Auto Launch), click Disconnect button. Pulse Connection(Auto Launch) connection gets disconnected and connection details gets stored in the Unified Pulse Secure Client for future references, as store parameter is set to true in this scenario.

Otherwise, if store parameter is set to false, then the connection details of Pulse Connection(Auto Launch) would not be stored after disconnection. Also, next launch of the Unified Pulse Secure Client with same URL will create a new connection.

If the user tries to connect the connection with same connection name but with different server URL, following error message appears:

**Benefits**

Following are the benefits of this feature:

- **Fast Connection**: As URL will handle the Unified Pulse Secure Client launch, user needs not to login through PCS, which reduces number of logins, hence time saving and fast connection.
• **Enhancing User Experience**: When PCS (IP or FQDN based), username and realm are prefilled, user just needs to enter the password to login.

• With the help of Store parameter in launch URL format, it will be possible to have temporary client entries. This ensures that each connection need not to be stored in the PDC and PDC does not get filled up with a pile of entries.

• **Scriptability**: Programmatically driven launch of Unified Pulse Secure Client lessens the burden of the Administrator.

Following is the scenario to understand the scriptability behaviour of this feature:

1. User enters the URL in command Prompt as shown below:

   **Figure 39  Command Prompt Window**

   ![Command Prompt Window](image)

   A permission dialog box appears to get the confirmation from the user to launch Unified Pulse Secure Client application.

   **Figure 40  Pop-up Window**

   ![Pop-up Window](image)

2. User clicks **Allow** button, following authentication screen appears for user to authenticate:
3. Click Connect. Following screen appears:

Figure 42 Test Connection(Auto Launch) established

Connection named Test Connection(Auto Launch) with provided values in Table 10 is established.

Installing Unified Pulse Secure Client on Windows Endpoints Using a Preconfiguration File

Note: The following procedures apply to Windows installations only.

After you create client connection sets and specify the connections to include within a client component set, you can create a preconfiguration file with all of the connections you want to distribute with Unified Pulse Secure Client. You specify the preconfiguration file as an option when you run the Unified Pulse Secure Client MSI installer program using an msiexec (windows\system32\msiexec.exe) command.
To create a preconfigured Unified Pulse Secure Client installer for distribution to Windows endpoints:

1. Select **Users > Pulse Secure > Connections** and create a connection set with the connections that you want to distribute.

2. Select **Users > Pulse Secure > Components**.

3. If necessary, create a new component set with the connection sets you want to distribute.
   
   It does not matter which component option you select, All components or No components. The Unified Pulse Secure Client installer installs all components.

4. Select the check boxes next to the component sets that you want to distribute.

5. Click **Download Installer Configuration**.

   You are prompted to save the preconfiguration. You can also specify the name of the target Pulse Secure server for the connections, which enables you to create configuration files that are the same except for the target server.

   The default filename of the .pulsepreconfig file is the name of the selected component set. Make note of the filename and location where you put the file. The preconfiguration file must be available to the clients either through a network share or distributed along with the Unified Pulse Secure Client installation file.

6. Select **Maintenance > System > Installers**.

   If necessary for your environment, download and install the Unified Pulse Secure Client Installer. To install Unified Pulse Secure Client, users must have appropriate privileges. The Unified Pulse Secure Client Installer allows you to bypass privilege restrictions and allow users with limited privileges to install Unified Pulse Secure Client.

7. Download the appropriate Unified Pulse Secure Client installer for your Windows environment:
   
   • Unified Pulse Secure Client installer (32-bit)
   • Unified Pulse Secure Client installer (64-bit)

**Note:** For a Windows installation (.msi) that uses an automated distribution mechanism and where the users do not have administrator privileges, you should ensure that the installation is run in the proper context, typically the USER context. To install in USER context, first advertise the .msi while in the SYSTEM context. For example, to advertise the 64-bit Windows installation to all users, use the following msiexec command:

```plaintext
msiexec /jm \PulseSecure.x64.msi
```

The advertisement allows the installation to be run in USER context even if the user is a restricted (non-admin) user. The location where the advertisement is run and where the actual installation is run must be the same. If the installation is an upgrade, you must advertise the upgrade version before running it. (Note that it is much easier to upgrade Unified Pulse Secure Client by not disabling the automatic upgrade feature on the Pulse Secure server.) After the installation is run by the user, Unified Pulse Secure Client will use the correct user certificate and context.
Installing Unified Pulse Secure Client Using Advanced Command-Line Options

The Unified Pulse Secure Client installer includes Unified Pulse Secure Client and all the software components for all related services. The preconfiguration file contains the definitions of the Unified Pulse Secure Client connections that provide client access to specific Pulse Secure servers and services.

Usage Notes:

- The preconfigured installer installs all Unified Pulse Secure Client components.
- When you run msiexec, you should append /qn or /qb (msiexec options) to the command line to suppress the installation program user interface. The /qn option specifies a silent install, so no user interface appears. The /qb option also hides the user interface but it displays a progress bar.
- The procedures in this topic are valid with Windows installations only. For information about installing Unified Pulse Secure Client on OS X endpoints, see “Installing Unified Pulse Secure Client on OS X Endpoints Using a Preconfiguration File” on page 72.

You run the Unified Pulse Secure Client preconfigured installer program with msiexec (the command line for launching .msi programs on Windows platforms) and specify the following options.

**Note:** Command-line options CONFIGFILE is case sensitive and must be all caps.

**Note:** If the path to the pulsepreconfig file includes spaces, be sure to use quotes around the path.

- **CONFIGFILE:** This property specifies a configuration file to be imported into Unified Pulse Secure Client during installation. The property must include the full path to the configuration file. For example:

  msiexec /i PulseSecure.x86.msi CONFIGFILE="c:\temp\my configuration..pulsepreconfig"

**Examples**

To install Unified Pulse Secure Client on a 32-bit Windows endpoint using a configuration file:

  msiexec /i PulseSecure.x86.msi CONFIGFILE=c:\temp\myconfiguration.pulsepreconfig /qb

To install Unified Pulse Secure Client on a 64-bit Windows endpoint using a configuration file:

  msiexec /i PulseSecure.x64.msi CONFIGFILE=c:\temp\myconfiguration.pulsepreconfig /qb

**Repairing a Unified Pulse Secure Client Installation on a Windows Endpoint**

Unified Pulse Secure Client uses an MSI installer, which supports a repair function. If problems with Unified Pulse Secure Client on a Windows endpoint indicate missing or damaged files or registry settings, the user can easily run the installation repair program. The repair program performs a reinstallation and replaces any missing files. The repair program does not install any files that were not part of the original installation. For example, if the file that holds Unified Pulse Secure Client connection configurations is damaged, the file installed by the repair program does not replace any Unified Pulse Secure Client connections that were created by the user or deployed to the endpoint after the original Unified Pulse Secure Client installation.

To repair a Unified Pulse Secure Client installation on a Windows endpoint:
1. On the Windows endpoint where Unified Pulse Secure Client is installed, click **Start > Programs > Pulse Secure > Repair Pulse Secure**.

2. Follow the prompts for the installation wizard.

When the program is finished, you might be prompted to reboot the system.

### Installing Unified Pulse Secure Client on OS X Endpoints Using a Preconfiguration File

**Note:** The following procedures apply to OS X installations only.

After you create client connection sets and specify the connections to include within a client component set, you can create a preconfiguration file with all the connections you want to distribute with Unified Pulse Secure Client. After you run the Unified Pulse Secure Client installer on the endpoint, you run a special command that imports the settings from the preconfiguration file into Unified Pulse Secure Client.

To create a preconfigured Unified Pulse Secure Client installer for distribution to OS X endpoints:

1. Select **Users > Pulse Secure > Connections** and create a connection set with the connections that you want to distribute.

2. Select **Users > Pulse Secure > Components**.

3. If necessary, create a new component set with the connection sets you want to distribute.

   The All components or No components options apply to Web-based installations only. The Unified Pulse Secure Client installation program for OS X always installs all components.

4. Select the check boxes next to the component sets that you want to distribute.

5. Click **Download Installer Configuration**.

   You are prompted to save the preconfiguration. You can also specify the name of the target Pulse Secure server for the connections, which enables you to quickly create multiple configuration files that are the same except for the target server.

   The default filename of the ".pulsepreconfig" file is the name of the selected component set. Make note of the filename and location where you put the file. The preconfiguration file must be available to the clients either through a network share or distributed along with the Unified Pulse Secure Client installer file.

6. Select **Maintenance > System > Installers**.

7. Download the Unified Pulse Secure Client installer, "Unified Pulse Secure Client installer (Macintosh)".
Installing Unified Pulse Secure Client on OS X Endpoints Using Command-Line Options

The Unified Pulse Secure Client installer includes Unified Pulse Secure Client and all of the software components for all of the Unified Pulse Secure Client services. The preconfiguration (.pulsepreconfig) file contains the definitions of the Unified Pulse Secure Client connections that provide client access to specific Pulse Secure servers and services. After you distribute the Unified Pulse Secure Client installation package, you must first run the installer, and then run a separate program called jamCommand, which imports the settings from the .pulsepreconfig file. The jamCommand program is part of the Unified Pulse Secure Client installation.

The Unified Pulse Secure Client file you download from the Pulse Secure server is in compressed (.dmg) format. You must unpack the file before you run the Unified Pulse Secure Client installation program.

The following steps include sample commands to install Unified Pulse Secure Client on an OS X endpoint and then import Unified Pulse Secure Client connections from a .pulsepreconfig file.

1. Run the Unified Pulse Secure Client installation program:
   
   ```bash
   sudo /usr/sbin/installer -pkg <full-path-to-the-pulse-install-package> -target /
   ```

2. Import the settings from the .pulsepreconfig file:

   ```bash
   /Applications/PulseSecure.app/Contents/Plugins/JamUI/./jamCommand -importfile /Users/<user profile>/<pre-config file location on local disk>/<preconfig file name>
   ```

Installing Unified Pulse Secure Client on Linux Using Command-Line Options

The nss3-tools and net-tools are dependency packages required to successfully install the Pulse Client. Use the following commands to install these dependency tools manually.

- **Fedora:**
  
  ```bash
  yum install <dependency tool name>
  ```

- **Ubuntu and Debian:**
  
  ```bash
  apt-get install < dependency tool name >
  ```

The Installation Command:

- **Debian – based installation**
  
  ```bash
  dpkg -i <package name>
  ```

- **RPM – based installation**
  
  ```bash
  rpm -ivh <package name>
  ```

The Uninstallation Command:

- **Debian – based installation**
  
  ```bash
  dpkg -r <package name>
  ```

- **RPM – based installation**
  
  ```bash
  rpm -e <package name>
  ```

**Note:** The upgrade from old Pulse client to new Unified Pulse Secure Client is not supported.
Unified Pulse Secure Client Command-line Launcher

The Unified Pulse Secure Client Launcher (pulselauncher.exe) is a standalone client-side command-line program that allows you to launch Unified Pulse Secure Client and connect to or disconnect from a Pulse Secure server (Pulse Connect Secure or Pulse Policy Secure) without displaying the Unified Pulse Secure Client graphical user interface.

Unified Pulse Secure Client Launcher Usage Notes:

- Unified Pulse Secure Client Launcher runs on Windows 32-bit and 64-bit endpoints.
- The Unified Pulse Secure Client Launcher program, pulselauncher.exe, is installed as part of a Unified Pulse Secure Client installation in Program Files\Common Files\Pulse Secure\Integration or Program Files (x86)\Common Files\Pulse Secure\Integration.
- Unified Pulse Secure Client Launcher works only for the Connect Secure or Policy Secure (L3) connection type. Unified Pulse Secure Client Launcher does not support the SRX Series or Policy Secure (802.1X) connection types.
- The Unified Pulse Secure Client Launcher program does not support the role mapping option that prompts a user to select from a list of assigned roles. If you use the Unified Pulse Secure Client Launcher and more than one role can be assigned to a user, you must configure the role mapping settings for the realm to merge settings for all assigned roles. If the realm settings require the user to select a role, the Unified Pulse Secure Client Launcher command fails and exits with return code 2.
- Unified Pulse Secure Client Launcher does not support secondary authentication.

To use Unified Pulse Secure Client Launcher:

1. Write a script, batch file, or application.
2. Include a call to the Unified Pulse Secure Client Launcher executable, pulselauncher.exe.
3. Include logic in your script, batch file, or application to handle the possible return codes.

Table 11 lists the Unified Pulse Secure Client Launcher arguments.

The following command shows the complete pulselauncher.exe command syntax:

```
```
### Table 11  Unified Pulse Secure Client Launcher Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>-version</td>
<td>Display the Unified Pulse Secure Client Launcher version information, then exit.</td>
</tr>
<tr>
<td>-help</td>
<td>Display available arguments information.</td>
</tr>
<tr>
<td>-stop</td>
<td>Stop Unified Pulse Secure Client and disconnect all active connections.</td>
</tr>
<tr>
<td>-L loglevel</td>
<td>Specify the log level to show in logs.</td>
</tr>
<tr>
<td>3</td>
<td>Normal - Log Critical, Error, Warning and Info messages (default)</td>
</tr>
<tr>
<td>5</td>
<td>Detailed - Log All messages</td>
</tr>
<tr>
<td>-url &lt;url&gt;</td>
<td>Specify the Pulse Secure server URL.</td>
</tr>
<tr>
<td>-u &lt;user&gt;</td>
<td>Specify the username.</td>
</tr>
<tr>
<td>-p &lt;password&gt;</td>
<td>Specify the password for authentication.</td>
</tr>
<tr>
<td>-r &lt;realm&gt;</td>
<td>Specify the realm on the Pulse Secure server.</td>
</tr>
<tr>
<td>-d &lt;DSID&gt;</td>
<td>Passes a cookie to Unified Pulse Secure Client Launcher for a specified Pulse Secure server from another authentication mechanism when Unified Pulse Secure Client Launcher starts. When you use the -d argument, you must also specify the -url argument to specify the Pulse Secure server.</td>
</tr>
<tr>
<td>-cert &lt;client certificate&gt;</td>
<td>Specify the certificate to use for user authentication. For &lt;client certificate&gt; use the string specified in the Issued To field of the certificate. When using the -cert argument, you must also specify the -url and -r &lt;realm&gt; arguments.</td>
</tr>
<tr>
<td>-signout &lt;url&gt;</td>
<td>Signout disconnects and signs out from a specific server. Suspend puts an active connection in the suspend state without removing the session information from the server. Resume restores a suspended connection. Unified Pulse Secure Client can have multiple simultaneous connections so the -url argument is required when you use -signout, -suspend, or -resume.</td>
</tr>
<tr>
<td>-t &lt;timeout in seconds&gt;</td>
<td>The amount of time allowed for the connection to take place before the attempt fails. Min = 45 (default), Max = 600.</td>
</tr>
</tbody>
</table>
Table 12 Unified Pulse Secure Client Launcher Return Codes

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>Unified Pulse Secure Client is not running.</td>
</tr>
<tr>
<td>0</td>
<td>Success.</td>
</tr>
<tr>
<td>1</td>
<td>A parameter is invalid.</td>
</tr>
<tr>
<td>2</td>
<td>Connection has failed or Unified Pulse Secure Client is unable to connect to the specified gateway.</td>
</tr>
<tr>
<td>3</td>
<td>Connection established with error.</td>
</tr>
<tr>
<td>4</td>
<td>Connection does not exist. Example: the command attempts to sign out from a server that has not been added on the Unified Pulse Secure Client UI.</td>
</tr>
<tr>
<td>5</td>
<td>User cancelled connection.</td>
</tr>
<tr>
<td>6</td>
<td>Client certificate error.</td>
</tr>
<tr>
<td>7</td>
<td>Timeout error.</td>
</tr>
<tr>
<td>8</td>
<td>No user connection allowed from Unified Pulse Secure Client UI.</td>
</tr>
<tr>
<td>9</td>
<td>No policy override from Unified Pulse Secure Client UI.</td>
</tr>
<tr>
<td>25</td>
<td>Invalid action for current connection state. This error code would occur if you tried to suspend or resume a connection that was disconnected.</td>
</tr>
<tr>
<td>100</td>
<td>General error.</td>
</tr>
</tbody>
</table>

Note: The return codes specified in Table 10 refer to the executable’s return codes. On Windows, you can display the last error level with "echo %errorlevel%" (without quotes). On OSX, the command is "echo $?" (without quotes).

Examples

The following command is a simple login application that captures the credentials the user enters, and passes the credentials as arguments to pulselauncher.exe:

```
pulselauncher.exe -u JDoe -p my$Pass84 -url https://int-company.portal.com/usr -r Users
pulselauncher return code: 0
```

The following Unified Pulse Secure Client Launcher example shows a certificate authentication:

```
pulselauncher return code: 0
```

The following example shows a command to use Unified Pulse Secure Client Launcher to specify a cookie (-d) for a specific Pulse Secure server (-url):

```
pulselauncher.exe -d 12adf234nasu234 -url https://int-company.portal.com/usr
pulselauncher return code: 0
```
Using jamCommand to Import Pulse Secure Connections

The jamCommand.exe program is a command line program that imports a .pulsepreconfig file into Unified Pulse Secure Client. The jamCommand program is available for Windows (Vista, Windows 8.1, and later) and macOS.

A .pulsepreconfig file includes Unified Pulse Secure Client connection parameters. You can create a .pulsepreconfig file on the Pulse Secure server, and then use it as part of a Unified Pulse Secure Client installation to ensure that Unified Pulse Secure Client users have one or more Unified Pulse Secure Client connections when they start Unified Pulse Secure Client for the first time.

**Note:** One typical use case for jamCommand on a Windows endpoint is to first run jamCommand to import one or more Unified Pulse Secure Client connections from a .pulsepreconfig file, and then run pulselauncher.exe to start Unified Pulse Secure Client.

To install Unified Pulse Secure Client connections using jamCommand:

1. Create a .pulsepreconfig file on the Pulse Secure server.
   
   In the Pulse Secure server admin console, click **Users > Pulse Secure > Components**.

2. Select the component sets you want, and then click **Download Installer Configuration**.

3. Distribute the .pulsepreconfig file to the Unified Pulse Secure Client endpoints.

4. Run jamCommand with the .pulsepreconfig file as an option. For example:

   On Windows:
   
   `<Program Files\Common Files\Pulse Secure\JamUI\jamCommand -importfile myfile.pulsepreconfig`

   On macOS:
   
   `/Applications/PulseSecure.app/Contents/Plugins/JamUI/./jamCommand -importfile /Users/<user profile>/<pre-config file location on local disk>/<preconfig file name>`

   On Linux
   
   `/opt/pulsesecure/bin/jamCommand /ImportFile ~/Downloads/pulsepreconfig`

If Unified Pulse Secure Client is running when you run jamCommand, the new Unified Pulse Secure Client connection or connections appear immediately. The connection name appears as it was defined when you created the connection in the Pulse Secure server admin console.
### jamCommand Reference

**Syntax**

```
jamCommand [-import [script]] [-tray] [-log[level]]
```

- `/import`
- `/importFile <script>`
- `/tray`
- `/log <level>`
- `/stop`
- `/suspend <GUIDS>`
- `/resume <GUIDS>`
- `/resume`
- `/brand <brandfile>`
- `/unbrand`
- `/norestart`

**Release Information**

- Introduced with Unified Pulse Secure Client R1.0.
- Unified Pulse Secure Client R3.1 introduced the suspend and resume options.
- Unified Pulse Secure Client R4.0.3 introduced new options to support Unified Pulse Secure Client Customization tool.

**Description**

The `jamCommand.exe` program is a command line program that imports a `.pulsepreconfig` or a "Branding.PulseBrandingPackage" file into Unified Pulse Secure Client. The `jamCommand` program is available for Windows and macOS.

**Options**

- `import`: Import script from the default memory-mapped file.
- `importFile <script>`: Import script from the specified file.
- `tray`: Launch the tray notify application.
- `log`: Set the global log level.
- `stop`: Stop the Unified Pulse Secure Client UI.
- `suspend <GUIDS>`: Suspend the Unified Pulse Secure Client UI.
- `resume <GUIDS>`: Resume a suspended Unified Pulse Secure Client UI.
- `brand <brandfile>`: Install the Unified Pulse Secure Client UI changes defined in the Unified Pulse Secure Client branding file.
- `unbrand`: Remove the changes applied by the Unified Pulse Secure Client branding file.
- `norestart`: Do not restart Unified Pulse Secure Client after applying the Unified Pulse Secure Client branding file.
Managing Server Certificate Authorities
Pulse Secure Linux clients verifies server certificate with trusted Certificate Authorities (CA) store in the system. Follow the instructions to add issuing CA certificate to store.

Note: CA certificates are stored in PEM format in trusted CA store. Following command is used to convert CA certificates to PEM format from DER format.
openssl x509 -in cert.crt -inform der -outform pem -out cert.pem

Linux (Ubuntu, Debian)
To Add CA certificate into system store:
1. Install the ca-certificate package
   sudo apt-get install ca-certificates
2. Copy the CA certificate which has been used to sign the device certificate, to /usr/local/share/ca-certificates directory:
   sudo cp device-ca.crt /usr/local/share/ca-certificates/device-ca.crt
3. Copy the CA certificate which has been used to sign the certificate of Identity Provider (IdP) (in case of SAML authentication), to /usr/local/share/ca-certificates directory:
   sudo cp idp-ca.crt /usr/local/share/ca-certificates/idp-ca.crt
4. Update the CA store:
   sudo update-ca-certificates

Linux (Fedora)
To add CA certificate into system store:
1. Become Super User of the machine using the following command:
   su-
2. Install the ca-certificates package:
   yum install ca-certificates
3. Copy the CA certificate which has been used to sign the device certificate, to /usr/local/share/ca-certificates directory:
   sudo cp device-ca.crt /etc/pki/ca-trust/source/anchors/
4. Enable the dynamic CA configuration feature:
   update-ca-trust force-enable
5. Copy the CA certificate which has been used to sign the certificate of Identity Provider (IdP) (in case of SAML authentication), to /usr/local/share/ca-certificates directory:

```
sudo cp idp-ca.crt /usr/local/share/ca-certificates/idp-ca.crt
```

6. Use command:

```
update-ca-trust extract
```

If the user connects to servers which have certificates not trusted by the machine the following error message displays:

**Figure 43** Error Message—Pulse Client

If connecting to untrusted sites through Embedded browser, the following error displays:

**Figure 44** Error Message—Embedded browser
Chromium Embedded Framework (CEF) Support

**Note:** This feature is supported for Unified Pulse Secure Client on Linux only.

Chromium Embedded Framework (CEF) is used as the embedded browser for custom sign-in, SAML Authentication, on all the platform to work with FIDO U2F.

**CEF installation on UI**

To install CEF browser using Pulse UI, use the following procedure.

1. Launch Pulse Linux Client application and select a connection and click **Connect**.

2. An authentication confirmation window appears. Click **OK** to continue.
3. A CEF download confirmation window appears, click **OK** to download and install CEF browser.

**Note:** The installation progress and status displays. Ensure not to initiate any other connection when CEF installation is in progress.
User is prompted for authentication and the Connection proceeds.

**Note:** If Pulse Linux Client is not closed, and opened again, the client authenticates the user without prompting for credentials after the first successful authentication. The client uses the cached cookies. This also applies in the case of Multi-factor authentication.

### CEF Installation Using scripts

Use the following util scripts “setup_cef.sh” to manage CEF using the scripts.

```
/opt/pulsesecure/bin$ ./setup_cef.sh <install|reinstall|uninstall > [-tmpDirPath <Path>]
```

The CEF package downloads and extracts to a temporary directory `-tmpDirPath` This directory is cleared upon installation.

- **install**: installs CEF only if not already installed.
- **reinstall**: removes and reinstalls CEF.
- **Uninstall**: removes the CEF.

**Note:** CEF reinstall is supported only using scripts.

- The **install** option runs only with root privileges.
- Installation requires 1276 MB of free space in the tmpDirPath. This space is used only during installation and freed upon installation.
- 1063 MB of free space is needed in the `/opt`
To check if CEF is installed

```
/opt/pulsesecure/bin$ ./setup_cef.sh check_installed
```

**Note:** Uninstalling Pulse Client does not remove CEF library or the client certificates used for Certificate Authentication.
Customizing Pulse Secure Desktop Client

- Customizing Pulse Secure Desktop Client Overview ........................................... 85
- BrandPackager Workflow ................................................................. 86
- Setting Up the Pulse Client Customization Environment ................................. 87
- Initializing the Pulse Client Customization Environment ............................... 88
- Importing an Existing Customized Pulse Client Package ............................... 89
- Editing Pulse Client User Interface Labels .................................................. 89
- Editing Pulse Client Messages ..................................................................... 93
- Adding Custom Graphics to Pulse Client ...................................................... 94
- Customizing Pulse Client for Apple OS X Online Help ................................. 95
- Validating Customizations to Pulse Client .................................................... 96
- Building the New Pulse Client Package ...................................................... 96
- Testing the Pulse Client Package .............................................................. 96
- Installing or Upgrading Pulse Client for Windows with a Branding Package .... 97
- Installing or Upgrading Pulse Client for Apple OS X with a Branding Package 97
- Installing a Branding Package Only ............................................................ 98

Customizing Pulse Secure Desktop Client Overview

The Pulse Secure Desktop Client (Pulse Client) customization tool BrandPackager enables you to customize the appearance of the Pulse Client for Windows and Apple OS X. You can add your own identity graphic to the Pulse Client splash screen, to the program interface, and to Windows credential provider tiles. Figure 107 shows graphic customizations applied to the Pulse Client for Windows. You can also customize error and informational message text, the text that appears in dialog boxes and on buttons, and make limited changes to Pulse Client online Help. For example, you might want to add your help desk phone number to Pulse Client error messages and the Pulse Client online Help.

BrandPackager is available for download from the Pulse Secure website (www.pulsesecure.net).

BrandPackager runs on Windows only, but you use it to create the package files for Pulse Client for Windows and Pulse Client for OS X. A package file contains your edits to Pulse Client resource files. The edited resource files are installed into a special folder on the client. When Pulse Client needs to access a particular file, it checks this special folder first and uses the file if it is present. If Pulse Client does not find the file there, it uses the file that resides in the normal Pulse Client resource file location.

For Windows, you deploy the package to endpoints and use an MSIEEXEC command-line installation option to instruct the installation program to apply your package file on the endpoint. For OS X, you copy the package file, the Pulse Client installation program, and a script file from the BrandPackager file set to an OS X computer, and then use them to add the package file to the Pulse Client installation file.
You can apply your changes to new or upgrade installations. You can also apply your customizations to an existing Pulse Client installation without installing or upgrading Pulse Client. Your changes to the Pulse Client user interface, message text, and online Help persist through normal client software upgrades.

Figure 45  Pulse Client Interface and Splash Screen with Branding Graphics

BrandPackager Usage Notes:

- BrandPackager supports Pulse Client for Windows and Pulse Client for macOS.
- BrandPackager is compatible with Pulse Client R5.0 or later.
- Pulse Client customizations cannot be installed through Pulse Secure Web portal (server) installations.
- When you edit Pulse Client resource files, you must preserve the UTF-8 encoding. UTF-8 files include 3 bytes {0xEF, 0xBB, 0xBF}, the Byte Order Mark (BOM), at the beginning of the file.
- The Pulse Client interface and the online Help include separate resource files for each of the supported languages. If you make a change in the English file, you should make the same change in the files for the other languages that you support in your environment. If you do not do so, then the edited English version is always used.
- Pulse Client online Help can include new information with each new release. If you edit a Help topic, your changes are retained during a Pulse Client upgrade. However, if Pulse Secure changes that topic in the new release, that new information will not be available, because your edited topic will be used instead. For this reason we recommend that you make only limited changes to the online Help. For example, you can change the topic that describes how to contact customer support to direct users to contact your own help desk.

BrandPackager Workflow

To create a rebranded Pulse Client, you use the BrandPackager tool. The following procedure summarizes the steps from tool installation to client deployment. See the related documentation list for links to detailed information about the steps that are summarized here.
1. Download PulseBrandingTools.zip from the Pulse Secure website (www.pulsesecure.net). Create a folder on a Windows 8 or later version for PulseBrandingTools.zip, and then unzip it. Make sure that the host computer has Pulse Client installed, and that the version of Pulse Client is the one that you want to customize and distribute to users.

   Set up the customization environment by installing 7Zip, a free open-source archive file program, and by running the BrandPackager initialization command to copy Pulse Client resource files to local work folders. To edit an existing package file, first import the file as part of the initialization process.

2. Edit the Pulse Client user interface files as needed.
3. Edit the Pulse Client message text files as needed.
4. Add your customization graphics.
5. Optionally, edit the Pulse Client online Help. There are separate procedures for the Windows and OS X online Help systems.
6. Run the BrandPackager script file to verify the structure of your changes and to create your package files.
7. Test your packages. The BrandPackager tool set provides a script to quickly activate your changes on the local machine for testing.
8. Deploying the package file is different depending on the platform:
   • For a Windows deployment, you install the package file by using an MSIEXEC command option when you run the Pulse Client installer.
   • For an OS X deployment, you copy the branding package, the default Pulse Client for OS X installation file (PulseSecure.dmg), and ConfigureInstaller to the Mac, and then run ConfigureInstaller. ConfigureInstaller is a Python script that adds the package file to the Pulse Client installation program. You can then run the Pulse Client for OS X installation.

Setting Up the Pulse Client Customization Environment

The Pulse Client BrandPackager customization tool must be run on a windows 8.1 or later version computer that has Pulse Client 5.0 or later installed. Make sure that the Pulse Client installation includes all Pulse Client components to ensure that you have access to all of the Pulse Client resource files. BrandPackager creates the package files for Pulse Client for Windows and Pulse Client for OS X. A package file contains your edits to Pulse Client resource files.

To create the Pulse Client customization environment:

1. If you have not already done so, download PulseBrandingTools.zip from the Pulse Secure website (www.pulsesecure.net). Create a folder for PulseBrandingTools.zip, and then unzip it. Make sure that the host computer has Pulse Client installed, and that the version of Pulse Client is the one that you want to customize and distribute to users.
2. Install 7Zip.

   7Zip is a free open-source archive file program. It is used during the process of creating the Pulse Client customization package. You can download 7Zip from http://7-zip.org/.
3. If you have not already done so, install Pulse Client 5.0 or later on the endpoint where you will do the Pulse Client customization work.

### Initializing the Pulse Client Customization Environment

The message text and user interface strings that appear in Pulse Client reside in text files that reside in different Pulse Client installation directories. After you install the BrandPackager tool, you run an initialization command that copies all the strings from the Pulse Client installation directories to two language-specific files in a reference directory called `StringReference`. The Pulse Client resource files are identical on Windows and OS X installations so the files from your Pulse Client for Windows installation can be used for both Windows and OS X customizations.

During initiation, the Pulse Client customization tool creates the `PulseBranding` directory and copies Pulse Client strings from an active Pulse Client installation to the `StringReference` directory area for customization.

BrandPackager copies files from the local Pulse Client installation, so make sure that you have the Pulse Client version installed that you want to customize and distribute.

Make sure that the Pulse Client installation includes all Pulse Client components. You can download the Pulse Client installation program from a Pulse Policy Secure server or from a Pulse Connect Secure server. You can configure and include Pulse Client connections in the installation before you edit Pulse Client files.

To initialize the Pulse Client customization environment:

1. Run the following command:

   ```
   BrandPackager -init
   ```

   The `-init` option does not overwrite files. If there is already a `PulseBranding` directory, only missing files are written to it.

   By default, Pulse Client online Help files are not included. To include the Help files, specify the `-help` option:

   ```
   BrandPackager -init -help
   ```

   The online Help files are different between Windows and OS X. BrandPackager uses the Windows files from the local Pulse Client installation. The OS X files are included as part of the BrandPackager file set. The `-help` option creates two directories. The `help` directory holds the Windows files. The `PulseSecureHelp.Help` directory holds the OS X online Help files.

   You can run `BrandPackager -init -help` if you have already run the `-init` option and want to just add the Help files.

   Localized files in the `StringReference` directory are identified by a language identifier:

   - DE - German
   - EN - English
   - ES - Spanish
   - FR - French
Importing an Existing Customized Pulse Client Package

If you already have a customized BrandPackager package, you can import it and make further changes to it without starting over. Also, changes to Pulse Client Help are not retained during a Pulse Client software upgrade operation. You should import the old package that has the Help file changes, create a new package, and then include that with the upgrade.

**Note:** If you are upgrading to a new major release of Pulse Client, make sure you have the latest version of BrandPackager before you create a new BrandPackager package.

To import an existing customized BrandPackager package into the `PulseBranding` directory:

1. Open a Command Prompt window and make the `PulseBranding` directory your working directory.
2. Run the following commands:
   ```
   BrandPackager -init
   BrandPackager -import <filename>
   ```
   The `-import` option must include the filename of your existing BrandPackager package file. For example:
   ```
   BrandPackager -import C:/Staging/PulseWin.PulseBranding
   ```
   If your original BrandPackager package included changes to the online Help, run the optional `-help` option:
   ```
   BrandPackager -init -help
   BrandPackager -import <filename>
   ```
   The `-import` option overwrites any files in the `PulseBranding` directory. The program prompts you for confirmation before it makes any changes.

Editing Pulse Client User Interface Labels

You can modify any text string that appears in the Pulse Client user interface. Pulse Client user interface strings reside in the `StringReference\PulseResource_XX.txt` file. Your modified strings must reside in the `PulseBranding\BrandingResourceCatalog_XX.txt` file. (XX indicates the language.)
Note: If your Pulse Client environment uses Security Assertion Markup Language (SAML) for a Single Sign-on (SSO) authentication environment, the Pulse Client user sees a credential dialog box that is served from the Pulse Secure server instead of the local Pulse Client credential dialog box. The sign-in page is defined as part of the sign-in policy on the Pulse Secure server and Pulse Client embeds the sign-in page within a Pulse Client dialog box. To change the appearance of the SAML credential dialog, you must edit or create a new sign-in page on the Pulse Secure server.

The BrandingResourceCatalog files hold only the strings you modify. The default strings in their normal files are used for all strings that you do not modify.

The following procedure describes the workflow for modifying user interface strings using the English language version of the Pulse Client Edit Connection dialog box as an example:

1. Start Pulse Client and then display the Pulse Client string that you want to modify. For example, in the Pulse Client main window, select a connection and then click **File > Connection > Edit**. Press the **Alt** key to show shortcut characters.

   ![Pulse Client Dialog with Shortcut Keys Underlined](image)

   **Figure 46  Pulse Client Dialog with Shortcut Keys Underlined**

2. Take a screen shot of the screen that you want to modify.

   The screen shot is not required but it can help you maintain or create a new shortcut character when you edit the string in the catalog file. It is good practice to keep track of what you change so you can verify your changes later.

3. Find the string that you want to modify.

   Search `StringReference\PulseResource_EN.txt` for the string. The string might appear more than once. For example, the string "Server URL" appears twice as a value in `PulseResource_EN.txt` because that string appears in two different dialog boxes. In general, the resource ID indicates where the value is used.
Many strings use an ampersand (&) to designate a keyboard shortcut key. The ampersand causes the character that follows it to appear as an underlined character in the user interface. The presence of the ampersand can affect your results when you use the editor's search function.

4. Open PulseBranding\BrandingResourceCatalog_EN.txt with a text editor.

5. Copy the string that you want to edit from PulseResource_EN.txt to BrandingResourceCatalog_EN.txt. Be sure to copy/paste the entire entry. For example:

```
;IDS_CONNECTION_DLST_URL [184] Value = &Server URL:
```

6. Modify the string in BrandingResourceCatalog_EN.txt. For example:

```
;IDS_CONNECTION_DLST_URL [184] Value = &Server URL:
```

Modify only the value. Do not change the string identifiers, ;IDS_CONNECTION_DLST_URL and [184].

We suggest that you keep the same letter for the shortcut to avoid a conflict with other strings on the screen. If the shortcut key letter does not appear in the new string, you can include it by putting it in parentheses. For example, the following entries show how to change Close to Exit and retain the "C" as a shortcut key:

```
;IDS_MAIN_DLGT_BTN_CANC
[188] Value = &Close
```

```
;IDS_MAIN_DLGT_BTN_CANC
[188] Value = Exit(&C)
```

You should change the shortcut letter only if you are certain that the new letter is not used elsewhere in that dialog box.

Each shortcut key on the screen must be unique. You can eliminate the shortcut by deleting the ampersand. However, shortcut keys are a part of good user interface design.

7. Edit that same resource ID in each of the language files that your organization supports.

The Pulse Client interface includes separate files for each of the 10 supported languages. If you make a change in the English file, you should make the same change for the other languages that you support in your environment. If you do not do so, then the edited English version is always used.
After initialization, there are two files for each language in the StringReference directory:

PulseCombinedMessages_XX.txt

Message catalog files hold the text that appears in the Pulse Client program interface and dialog boxes.

PulseResource_XX.txt

Resource catalog files hold the text that appears in Pulse Client message boxes.

To customize a particular string, you find the string you want to customize in PulseCombinedMessages_XX.txt or PulseResource_XX.txt, and then copy and paste that entire string and its resource ID to a corresponding resource or message file in the PulseBranding directory, where you edit it. This directory holds all of the files that make up your customization package.

**Note:** You must use a text editor, such as Visual Studio IDE or Notepad++ that retains the byte order mark (BOM) in the resource files. (Notepad++ is free open source software available at [http://notepad-plus-plus.org/](http://notepad-plus-plus.org/)).

**Note:** See the Sample directory for an example of a customized Pulse Client file set.
Editing Pulse Client Messages

Pulse Client message strings reside in the StringReference\PulseCombinedMessages_XX.txt file. Modified message strings must reside in the PulseBranding\BrandingMessageCatalog_XX.txt file. (XX indicates the language.)

The BrandingMessageCatalog files hold only the strings that you modify. The default strings in the installed resource files are used for all strings that you do not modify.

It is not always possible to set up the conditions that cause a particular message to appear in Pulse Client. Browsing the contents of BrandingMessageCatalog_XX.txt is the easiest way to identify the strings you might want to change.

You can use HTML tags within the BrandingMessageCatalog entries. For example, you can use <b> and </b> tags to make text appear in bold type. You can use <A href=> tags to include a link to other HTML text you want. Make sure that your link displays the text in a new window. For example:

```html
<A HREF="Http://www.myserver/my-message.html" target="_blank">
```

Keep in mind that Pulse Client might not be connected to the Internet when the error occurs.

Each message includes a short description and a long description. The short description is shown as a title to the longer description. There are no limits to the number of characters that you can include as the long description. However, the long description must be on one line in the message catalog file. Use HTML <br> and </br> tags to insert line breaks when the message is displayed.

To modify a message:

1. Find the string that you want to modify.

   Search PulseCombinedMessages_XX.txt for the string. In general, the resource ID indicates where the value is used.

2. Open PulseBranding\BrandingMessageCatalog_XX.txt with a text editor.

3. Copy the string that you want to edit from StringReference\PulseCombinedMessages_XX.txt to BrandingMessageCatalog_XX.txt. Be sure to copy/paste the entire entry. For example:

   ```
   [1731] ;kMsgCommonCertTrustPulseAuthServerIdentityNotFoundShort-desc = Authentication server not trusted. Long-desc = Authentication server identity not found in client's "Trusted Server List". Contact your network administrator.
   ```

4. Modify the string in BrandingMessageCatalog_XX.txt. For example:

   ```
   ```

5. Edit that same resource ID in each of the language files that your organization supports.

The Pulse Client interface includes separate files for each of the 10 supported languages. If you make a change in the English file, you should make the same change for the other languages you support in your environment. If you do not do so, then the edited English version is always used.
Adding Custom Graphics to Pulse Client

The PulseBranding directory also includes default graphics. To add your custom graphics to the Pulse Client interface, simply replace the default graphics with your custom graphics.

You can add a graphic to the following areas:

- Next to the Pulse Secure logo on the main screen
- In dialog boxes
- On the About screen
- On the Pulse Client splash screen

**Note:** The Pulse Client connection set properties, which you define on the Pulse Secure server, include an option to suppress the Pulse Client splash screen.

The PulseBranding directory includes two graphics:

- **BrandingLogo.png**: Appears on the Pulse Client splash screen and program interface. The default BrandingLogo.png file is an empty file with a transparent background. For best results, your graphic image should have a transparent background. The file must be a PNG file.
  
  The default BrandingLogo.png file is 19 by 52 pixels. The maximum height is 37 pixels, which corresponds to the size of the Pulse Secure logo. Maximum width is 100 pixels. A graphic larger than the recommended size might be clipped or it could obscure other graphic elements.

- **BrandingCredProv.png**: Appears as the image on credential provider tiles.

To add a custom graphic:

1. Replace \PulseBranding\BrandingLogo.png with your graphic.
2. Replace \PulseBranding\BrandingCredProv.png with your graphic.

If you do not want to include a custom graphic, you should delete default graphics from PulseBranding.

The Pulse Client online Help provides reference and procedural information for users. Pulse Client users can access the Help by clicking the Help button in the Pulse Client program interface. Pulse Client online Help is a collection of standard HTML files with CSS formatting and javascript navigation. Updating the Help requires knowledge of basic HTML coding. To edit the online Help, you must include the Help when you initialize the Pulse Client customization environment.

**Note:** If you edit a Help topic, your edited topic is used instead of the original topic. Your edited topic is retained during an upgrade. Pulse Client online Help can include new information with each new release. If Pulse Secure changes a topic in the new release, that new information will not be available because your edited topic is used instead. To avoid this problem, we recommend that you make only the Help topic changes described in this guide.

You might want to edit that topic and substitute your own help desk contact information. Use an HTML editor to make your changes. Do not change the filename or any of the javascript code within the topic.

Pulse Client Help includes the following language versions:

- DE - German
Be sure to edit the same topics for all the languages that you support.

The Pulse Client Help viewer includes a menu item labeled Feedback, which links to a documentation comments page on www.pulsesecure.net.

To change the Feedback destination URL or to remove the menu item:

1. Open j_header.html with an HTML editor.
2. Search for the following string:
   https://support.pulsesecure.net
3. Either edit or remove the link.

Customizing Pulse Client for Apple OS X Online Help

The Pulse Client online Help provides reference and procedural information for users. Pulse Client users can access the Help by clicking the Help button in the Apple menu bar. Pulse Client online Help is a collection of standard HTML files with CSS formatting. Apple Help application acceleration includes special metadata in the header of each topic and a particular directory structure to properly interact with OS X. Updating the Help requires knowledge of basic HTML coding. To edit the online Help, you must include the Help when you initialize the Pulse Client customization environment.

**Note:** If you edit a Help topic, your edited topic is used instead of the original topic. Your edited topic is retained during an upgrade. Pulse Client online Help can include new information with each new release. If Pulse Secure changes a topic in the new release, that new information will not be available because your edited topic is used instead. To avoid this problem, we recommend that you make only the Help topic changes described in this guide.

You might want to edit that topic and substitute your own help desk contact information. The file resides in PulseSecureHelp.Help\Contents\Resources\<language>.lproj\pages. Filenames in OS X are case-sensitive.

Pulse Client for OS X online Help includes the following language versions:

- DE.lproj - German
- English.lproj - English
Validating Customizations to Pulse Client

The validation process examines the files in the PulseBranding directory to ensure that they can be added to the Pulse Client installation package.

To validate your changes before building the BrandPackager package, run the following command:

```
BrandPackager -validate
```

Validation is a basic level of checking. After you build the new Pulse Client installation package, you should test the package before you deploy it.

Building the New Pulse Client Package

The packaging process creates two package files, one for Windows and one for OS X, that include your changes. It does not include the Pulse Client installation files. You include a package when you install Pulse Client. Or you can apply your changes to a Pulse Client without installing or upgrading Pulse Client.

To create a package, run the following command:

```
BrandPackager -package
```

When the command finishes, it creates two package files, PulseWin.PulseBranding and PulseMac.PulseBranding. To apply your changes on a Pulse Client endpoint, you include a package file when you install or upgrade Pulse Client.

Testing the Pulse Client Package

Before you deploy the new Pulse Client installation package, you should verify that your changes work correctly. BrandInstaller.bat installs the BrandPackager package on the local machine. BrandInstaller.bat employs jamCommand.exe, which is a program that resides in the Pulse Client program directory.

**Note:** You must be an administrator to run BrandInstaller.

To install your BrandPackager package on the machine where you created it, run the following command:
BrandInstaller -brand

You can now view your changes on the local Pulse Client to make sure that you have made all the modifications correctly. Verify Pulse Client by checking the following:

- View the main dialog and the About screen to make sure that the branding logo appears as you want.
- View the screens that contain any of the user interface strings that you changed.
- If you have updated the Pulse Client for Windows Help, invoke the Help to make sure your changes are correct.

If you are satisfied, you can install the package on endpoints.

**Installing or Upgrading Pulse Client for Windows with a Branding Package**

You install or upgrade Pulse Client and apply the changes in PulseWin.PulseBranding to Pulse Client for Windows by using Microsoft Exec (msiexec) and setting the `BRANDINGFILE` attribute to point to the branding file. This installation requires administrative privileges.

The following example shows the msiexec command to install or upgrade Pulse Client and to apply the customizations in `PulseWin.PulseBranding`:

```
msiexec /i c:\staging\PulseSecure.x64.msi BRANDINGFILE=c:\staging\PulseWin.PulseBranding
```

**Installing or Upgrading Pulse Client for Apple OS X with a Branding Package**

To apply the branding package changes to an Apple OS X endpoint, you must copy the necessary files to an OS X endpoint and use them to update the Pulse Client installation program. You can also use this process to add Pulse Client configurations (a .pulsepreconfig file) to the Pulse Client installation program. You can then use that Pulse Client installation program to install or update Pulse Client on OS X endpoints. If the specified branding package is present in the Pulse Client installation program, the installation process creates the following directory:

```
/Library/Application Support/Pulse Secure/PulseBranding
```

The `PulseBranding` directory holds the changes you made to Pulse Client resource files and graphics. When Pulse Client must access a resource file, it checks this directory first.

To add `PulseMac.PulseBranding` to `PulseSecure.dmg`, perform the following steps on an OS X endpoint:

1. Create a directory on an OS X endpoint and copy the following files to it:
   - **PulseMac.PulseBranding**: The file created for OS X by BrandPackager that contains all of your client customizations. After you edit the resource files and run BrandPackager, `PulseMac.PulseBranding` is available in the same directory as BrandPackager.
   - **PulseSecure.dmg**: The Pulse Client installation program. You can download `PulseSecure.dmg` from the Downloads page of Pulse Connect Secure or Pulse Policy Secure.
• **ConfigureInstaller**: A Python script that adds the package file to PulseSecure.dmg. ConfigureInstaller is available in the same directory as BrandPackager. Python is part of OS X 10.2 and greater and is included in the system PATH.

2. Open a terminal window and make the directory that holds ConfigureInstaller your current directory.

3. Run ConfigureInstaller. You can run ConfigureInstaller with no options to see the command summary:

   ```
   python ./ConfigureInstaller
   usage -s <source dmg> -b <brandingfile> -c <configfile> -t <target dmg>
   usage -s <source dmg> -b <brandingfile> -t <target dmg>
   usage -s <source dmg> -c <configfile> -t <target dmg>
   ```

   The following example shows a command for adding a branding file and a Pulse Client config file to the Pulse Client installation program:

   ```
   python ./ConfigureInstaller -s PulseSecure.dmg -b ~/Staging/PulseMac.PulseBranding -c ~/Staging/myfile.pulsepreconfig -t PulseSecure-new.dmg
   ```

   When the operation completes successfully, the new Pulse Client installation program is ready for use.

---

### Installing a Branding Package Only

You can add or remove the contents of a Pulse Client branding package on a client machine by using `jamCommand`. The `jamCommand` program is part of every Pulse Client installation. On Windows endpoints, `jamCommand` is located in the 32-bit program files directory:

```
Program Files (x86)\Common Files\Pulse Secure\JamUI\jamCommand.exe
```

On OS X endpoints, `jamCommand` is located in the Applications folder:

```
/Applications/PulseSecure.app/Contents/Plugins/JamUI/./jamCommand
```

**Note:** The `jamCommand` program must be run with administrator privileges.

To apply the customizations in `PulseWin.PulseBranding` (Windows) or `PulseMac.PulseBranding` (OS X):

1. Run the following command:
   ```
   jamCommand -brand
   ```

   To remove your customized Pulse Client user interface from the endpoint and allow Pulse Client to use default strings:

2. Run the following command:
   ```
   jamCommand -unbrand
   ```

   `jamCommand` Usage Notes:
• Running jamCommand with the -brand or -unbrand option causes Pulse Client to restart. Connections are maintained and should be active after the restart. A restart is required to allow Pulse Client to access the customized settings. If you will be rebooting the system manually, or if there is no logged in user, then you can use the -norestart option. To avoid a restart when you run jamCommand, use the following option:

  jamCommand -norestart

• jamCommand reports its results using the following numeric error codes:

  0 - Success.
  1 - General branding error.
  2 - Error deleting branding files. This error can also occur when you install new files because the first action -brand performs is to remove the old files.
  3 - Error branding Pulse Client. The new branding files cannot be written.

• The Pulse Client version must be R5.0 or later. To verify your current version of Pulse Client, run jamCommand with no parameters. If the result (displayed in a window) shows the branding options (-brand, -unbrand, -norestart), then branding is supported.

The jamCommand errors are not written to the console. To see jamCommand errors, include a script that checks error codes. Additional error message information is written to the Pulse Client log files.
Client Software Feature Comparison

- Comparing Odyssey Access Client and Pulse Secure Desktop Client .......................... 100
- Comparing Network Connect and Pulse Client ......................................................... 104

Comparing Odyssey Access Client and Pulse Secure Desktop Client

Pulse Secure Desktop Client (Pulse Client) is a single integrated, multiservice network client that provides the basic services provided by the older Network Connect and Odyssey Access Client software. Pulse Client supports dynamic connectivity and secure access control for Microsoft Windows and macOS devices, and connectivity, and mobile device management (MDM) for mobile devices, all with a simple, easy to use, elegant user experience.

Table 13 compares the features in Odyssey Access Client (OAC) and Pulse Client to help you transition to Pulse Client. For detailed information about supported platforms and installation requirements, see the Pulse Secure Supported Platforms Guide available from the Pulse Secure website (www.pulsesecure.net).
Table 13  Odyssey Access client and Pulse Client Feature Comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client for OSX</th>
<th>Pulse Client for Windows</th>
<th>Odyssey Access Client</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Wired/Wireless 802.1X Features</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wired 802.1X support</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(with Microsoft</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows supplicant)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Auto scan lists</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(with Microsoft</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows supplicant)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wireless suppression</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(with Microsoft</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows supplicant)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Support for Network Provider (scraping</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>passwords, listing)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Association Mode and Encryption Methods</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Association mode support (for open, shared,</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>WPA/WPA2)</td>
<td>(with Microsoft</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows supplicant)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Encryption (for WEP, TKIP, AES, WEP with</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>preconfigured key, WPA/WPA2 with pre-shared</td>
<td>(with Microsoft</td>
<td></td>
<td></td>
</tr>
<tr>
<td>key)</td>
<td>Windows supplicant)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>EAP Methods</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EAP-TLS outer authentication</td>
<td></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>EAP-TTLS outer authentication</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>• With EAP-JUAC inner authentication</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>• With EAP-MSCHAPv2 inner authentication</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>• With EAP-GTC inner authentication</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>• With EAP-MD5 inner authentication</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>• With PAP inner authentication</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>• With CHAP inner authentication</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>• With MSCHAP inner authentication</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>• With MSCHAPv2 inner authentication</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>EAP-PEAP outer authentication</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>
### Feature | Pulse Client for OSX | Pulse Client for Windows | Odyssey Access Client
--- | --- | --- | ---
• With EAP-JUAC inner authentication | Yes | Yes | Yes
• With EAP-DD5 inner authentication | Yes | Yes | Yes
• With EAP-GTC inner authentication | Yes | Yes | Yes

### Authentication Methods

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client for OSX</th>
<th>Pulse Client for Windows</th>
<th>Odyssey Access Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prompt for user name and password</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Certificate support (automatic, specific)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Certificates from smart card reader</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Soft token support</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Machine login support</td>
<td>N/A</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Machine authentication followed by user authentication</td>
<td>N/A</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Credential provider on 32- and 64-bit Windows Vista, and Windows 8.1 and later</td>
<td>N/A</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Pre-desktop login</td>
<td>N/A</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Configurable UAC Layer 2 connection</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Configurable connection association modes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Certifications

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client for OSX</th>
<th>Pulse Client for Windows</th>
<th>Odyssey Access Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIPS compliance</td>
<td>Pulse Client SSL-VPN mode connection is fully FIPS compliant. Pulse Client server certificate verification and private key signing is FIPS compliant. Pulse Client IPSec is FIPS compatible. Pulse Client wireless is FIPS compatible. (WPA encryption is controlled by Windows.)</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Installation and Upgrade Methods

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client for OSX</th>
<th>Pulse Client for Windows</th>
<th>Odyssey Access Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto-upgrade</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Web-based installation</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Feature</td>
<td>Pulse Client for OSX</td>
<td>Pulse Client for Windows</td>
<td>Odyssey Access Client</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-----------------------</td>
<td>--------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>Standalone installation</td>
<td>Yes (.dmg)</td>
<td>Yes (.msi)</td>
<td>Yes</td>
</tr>
<tr>
<td>Upgrade/coordinate with previous versions</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Manual Uninstall</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Browser based installation and upgrades</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Diagnostics and Logging</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPSec diagnostics and configuration</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Host Enforcer</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Log viewer</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Logging and Diagnostics</td>
<td>Yes Set debug level</td>
<td>Yes Set debug level, set file size limits</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In addition to the Pulse Client log files, Pulse Client writes events to the Windows application event log. (Windows Vista and Windows 8.1 systems only.)</td>
<td></td>
</tr>
<tr>
<td>Other Features</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OPSWAT IMV support</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Automatic patch remediation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Yes via SMS/SCCM</td>
<td>Yes via SMS/SCCM</td>
<td></td>
</tr>
<tr>
<td>Host Checker support</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPsec tunneling to Policy Enforcement Points with NAT-T</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Access service and plug-ins</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Block 3rd party EAP messages</td>
<td>N/A</td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Layer 3 authentication</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Server-based pre-configuration of realm/role</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Extend session duration</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IC cardinality</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Comparing Network Connect and Pulse Client

Pulse Client is an integrated, multiservice network client that replaces Network Connect (NC) and Odyssey Access Client (OAC) software. Pulse Client provides dynamic connectivity, access control, and security, with a simple, easy to use, elegant user experience.

Table 14 compares the features of NC to Pulse Client for Windows and Pulse Client for OS X to help you transition from NC to Pulse Client. For detailed information about supported platforms and installation requirements, see the Pulse Secure Supported Platforms Guide available from the Pulse Secure website (www.pulsesecure.net).

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client for OSX</th>
<th>Pulse Client for Windows</th>
<th>Odyssey Access Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client-site management of clustered Pulse Secure servers</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Kerberos SSO</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Initial configuration (intervention-less client provisioning)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Dynamically configurable</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Table 14  Network Connect and Pulse Client Feature Comparison

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client Release 5.1</th>
<th>Network Connect Release 6.3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mac 10.8 and later</td>
<td>Win 7 and later Mac</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Release 6.3 Win</td>
</tr>
<tr>
<td><strong>Proxy Support</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse Client respects the system's understanding of the web proxy.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internet Explorer</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Mozilla Firefox</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Apple Safari</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Google Chrome</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Split Tunneling Options</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Disable split tunneling without route monitor</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Disable split tunneling with route monitor</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Enable split tunneling with route monitors</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Enable split tunneling without route monitors</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Enable split tunneling with allowed access to local subnet</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Disable split tunneling with allowed access to local subnet</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Disable split tunneling but allow directly connected local subnet access</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Client Launch Options</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command line launcher</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Log out on connect</td>
<td>Pulse Client implements this behavior through machine authentication.</td>
<td>Yes</td>
</tr>
<tr>
<td>Launch as a standalone client</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Launch from browser</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Feature</td>
<td>Pulse Client Release 5.1</td>
<td>Network Connect Release 6.3</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>--------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>Transport Mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSL fallback mode</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Note: If ESP is not available, the connection uses SSL. After a connection switches to SSL it does not go back to ESP until the connection is restarted.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ESP</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Other Features</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OPSWAT IMV support</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Patch automatic remediation</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Host Checker support</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enhanced Endpoint Security support</td>
<td>Note: Enhanced Endpoint Security was discontinued in February 2013.</td>
<td>Yes</td>
</tr>
<tr>
<td>Run configured scripts when client connects/disconnects</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Modify DNS server search order based on server configuration</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Reconnect automatically if connection breaks</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Dial-up adapter support</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>3G wireless adapter support</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Max/Idle Session Time-outs</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>IPv6</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Location awareness</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Feature</td>
<td>Pulse Client Release 5.1</td>
<td>Network Connect Release 6.3</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>Customizable user interface, including message text (all supported languages)</td>
<td>Yes</td>
<td>yes</td>
</tr>
</tbody>
</table>

### Authentication

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client Release 5.1</th>
<th>Network Connect Release 6.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Machine authentication</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Credential provider</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Smart cards</td>
<td>Yes</td>
<td>?</td>
</tr>
<tr>
<td>Soft token Auth - RSA and others</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Soft token integration - RSA</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Certificate Auth</td>
<td>Yes</td>
<td>yes</td>
</tr>
<tr>
<td>Login-logoff script support</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Start before log on</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Password expiration notification</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Password management (pass-through)</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

### Logging

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client Release 5.1</th>
<th>Network Connect Release 6.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log to file</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Set logging level</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Certifications

<table>
<thead>
<tr>
<th>Feature</th>
<th>Pulse Client Release 5.1</th>
<th>Network Connect Release 6.3</th>
</tr>
</thead>
</table>
### Pulse Client Split Tunneling

Table 15 lists the Network Connect split tunneling options and shows how they map to Pulse Client split tunneling options.

<table>
<thead>
<tr>
<th>NC Split Tunnel Option</th>
<th>Pulse Client Split Tunnel Setting</th>
<th>Route Override State</th>
<th>Route Monitor State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable split tunnel</td>
<td>Disabled</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Disable split tunneling but allow local access</td>
<td>Disabled</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Enable split tunnel</td>
<td>Enable</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Enable split tunnel with route monitor</td>
<td>Enable</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Enable split tunnel, allow local access</td>
<td>Enable</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
Unified Pulse Secure Client Authentication Types

- RSA Authentication ................................................................. 109
- Google Authentication ......................................................... 110
- Certificate Authentication Support ......................................... 111
- Yubikey Authentication Support ............................................. 116

**RSA Authentication**

RSA Authentication Manager (formerly known as ACE/Server) is an authentication and authorization server that allows user authentication based on credentials from the RSA SecurID® product from RSA Security Inc.

When you use RSA Authentication Manager as the authentication and authorization service for your Pulse Secure access management framework, users can sign into PPS using the same username and password stored in the backend server.

Table 16  RSA SecurID user sign-in methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using a hardware token and the standard system sign-in page</td>
<td>The user browses to the standard system sign-in page, and then enters the username and password (consisting of the concatenation of the PIN and the RSA SecurID hardware token's current value). The system then forwards the user's credentials to the authentication server.</td>
</tr>
<tr>
<td>Using a software token and the custom SoftID system sign-in page</td>
<td>The user browses to the SoftID custom sign-in page. Then, using the SoftID plug-in, the user enters the username and PIN. The SoftID plug-in generates a passphrase by concatenating the user's PIN and token and passes the passphrase to the authentication server.</td>
</tr>
</tbody>
</table>

If the RSA Authentication Manager positively authenticates the user, the user gains access to the system. Otherwise, the RSA Authentication Manager:

- Denies the user access to the system.
- Prompts the user to generate a new PIN (New PIN mode) if the user is signing into the system for the first time. Users see different prompts depending on the method they use to sign in.
- If the user signs in using the SoftID plug-in, then the RSA prompts the user to create a new PIN.
- Prompts the user to enter the next token (Next Token mode) if the token entered by the user is out of sync with the token expected by RSA Authentication Manager. Next Token mode is transparent to users signing in using a SoftID token. The RSA SecurID software passes the token through the system to RSA Authentication Manager without user interaction.
- Redirects the user to the standard system sign-in page (SoftID only) if the user tries to sign-in to the RSA SecurID Authentication page on a computer that does not have the SecurID software installed.
Google Authentication

The admin can associate an end-user to a realm that has a secondary authentication server configured as TOTP authentication server.

For first time registration through web, perform the following steps:

For example: Admin associates an end-user User1 to a user-realm that has the TOTP authentication-server configured as the secondary authentication-server.

When User1 for the first time, performs a login to the above configured user-realm:

1. After successful authentication with primary authentication-server, User1 is shown the TOTP registration page.

2. User1 is given a TOTP registration key in text form/QR image form and 10 backup codes. User saves 10 backup codes in a safe place for using it later during authentication when end-user device (where Google Authenticator app is installed) is not available (in emergency).

3. Now, User1 opens the device where Google Authenticator app is installed, then either scans the QR image (or) manually adds a new user (for example: GA-User1) by entering the above given secret registration key.

4. The Google-Authentication app (for GA-User1) generates a new 6-digit number called as a token once in every 30 seconds.

5. Enter the current token in the registration page. Click on Sign In. On successful authentication with that token, User1 will be taken to his/her home page.

Figure 49 First Time Registration to a TOTP Server

For already registered user, perform the following steps:
1. The already-registered user (For example: User1), whose realm was associated with secondary authentication server configured as TOTP authentication server, accesses PPS URL via web (User1 has already registered TOTP user in Google Authenticator app.)

2. After successful authentication with primary authentication server, user1 is shown TOTP Token entry page as seen in Figure 29

3. User1 opens Google Authentication app that was installed in mobile (or PC), enters the current token to the

4. Authentication Code. If mobile is not available, user can enter any of the unused backup codes.

5. On successful authentication with the token, User1 can enter any of the unused backup codes.

6. A backup code can be used only once to successfully authenticate with the TOTP authentication server. Once used, the same backup code cannot be reused.

Figure 50  Google Authentication Token

Certificate Authentication Support

This feature enables users to login to the client using their certificates. The supported scenario is “certificate-based login only” the Pulse Linux Client setup is now switched to this authentication method. In a typical enterprise environment, each user will be provided with certificate which can be used for VPN login. This mechanism can be used as a primary or secondary authentication mechanism.

Configuring Client Certificate in Pulse Connect Secure

To configure trusted client CA certificate:
1. Select **System > Configuration > Certificates > Trusted Client CAs.**

   Figure 51  Trusted Client CA Management

2. Click Import CA Certificate to display the configuration page.

   Figure 52  Import Trusted Client CA

3. Browse to the certificate file and select it.

4. Click Import Certificate to complete the import operation.

5. Click the link for the Trusted Client CA to configure.
Configuring Authentication with the Certificate Server

To configure authentication with the certificate server, follow the steps below:

1. Select **Authentication > Auth Servers**.

2. Select Certificate Server and Click **New Server** to display the configuration page.

3. Complete the configuration as described in the following table:

   - **Administrators**
     - Type: Local Authentication
   - **System Local**
     - Type: Local Authentication

---

**Figure 53** Trusted Client CA Configuration

**Figure 54** Authenticating Servers
Table 17  Certificate Server Settings

<table>
<thead>
<tr>
<th>Settings</th>
<th>Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a name to identify the server within the system</td>
</tr>
<tr>
<td>User Name Template</td>
<td>Specify a username template. Specify how the system should construct a username. You may use any combination of certificate variables contained in angle brackets and plain text. NOTE: This value populates the &lt;USER&gt; and &lt;USERNAME&gt; session variables for use throughout the rest of the system configuration.</td>
</tr>
<tr>
<td>User Record Synchronization</td>
<td>This applies only to Connect Secure.</td>
</tr>
<tr>
<td>Enable User Record Synchronization</td>
<td>Select this option to retain the bookmarks and individual preferences regardless of which system you log in to.</td>
</tr>
<tr>
<td>Logical Auth Server Name</td>
<td>Specify a logical authentication server name.</td>
</tr>
</tbody>
</table>

Figure 55  Configuring Certificate Server

4. Save the configuration.

Client Certificate Installation

The installation of the certificates can be facilitated through a script. Client certificates can be installed using util script “certificate_installer.sh”. Use the following commands to install or delete the certificates:

- To install the certificate:
/opt/pulsesecure/bin/certificate_installer.sh install_certificates [-inpf < PFX /P12 file >] [-inpriv <private file> -inpub <public file>]

Note: Password is required to install private and public keys separately.

- To list the certificates on the certificate store
  /opt/pulsesecure/bin/certificate_installer.sh list_installed_certificates
- To delete the Certificate from certificate store
  /opt/pulsesecure/bin/certificate_installer.sh delete_certificates -certName <certificate name>

Note: To delete certificates from CEF certificate store:
/usr/bin/certutil -d sql:/ /$HOME/.pki/nssdb -D -n <Nickname>
where, 'Nickname' is available in list of installed certificates

Public Certificates

<table>
<thead>
<tr>
<th>Extensions</th>
<th>Certificate Formats</th>
</tr>
</thead>
<tbody>
<tr>
<td>der, cer</td>
<td>DER</td>
</tr>
<tr>
<td>pem, crt, key, pub</td>
<td>PEM</td>
</tr>
</tbody>
</table>

Private Keys

<table>
<thead>
<tr>
<th>Extensions</th>
<th>Certificate Formats</th>
</tr>
</thead>
<tbody>
<tr>
<td>der, cer</td>
<td>DER</td>
</tr>
<tr>
<td>pem, crt, key</td>
<td>PEM</td>
</tr>
</tbody>
</table>

PFX file (Contains both Private Key and Public Keys)

<table>
<thead>
<tr>
<th>Extensions</th>
<th>Certificate Formats</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pfx, p12</td>
<td>PFX</td>
</tr>
</tbody>
</table>

Default Certificate Selection

If multiple certificates are available for a connection, the certificates list allows the user to select the certificate and authenticate to establish the connection.
Figure 56  Certificate List

Note: Client certificate authentication through smart cards is not supported.

Yubikey Authentication Support

Note: This feature is applicable for Unified Pulse Secure Client on Linux only.

Yubikey is a hardware token for Multifactor Authentication that supports OTP, with plans to adopt modern authentication approaches such as FIDO U2F with single security key.

Chromium Embedded Framework (CEF) is used as the embedded browser for custom sign-in, SAML Authentication on Linux to work with FIDO U2F.

Pulse Linux client integrates Yubikey for MFA with CEF to redirect to the IDP such as Netflix and Okta.

On PCS, enable “Enable embedded browser for authentication” option in Connections settings for Pulse Linux Client to launch CEF for sign in.

To set up Yubikey for authentication and install CEF browser, use the following procedure.

1. Launch Pulse Linux Client application and select a connection and click Connect.

2. An authentication confirmation window appears. Click OK to continue.
3. A CEF download confirmation window appears, click **OK** to download and install CEF browser.

**Note:** The installation progress and status displays. Ensure not to initiate any other connection when CEF installation is in progress.
4. On successful installation of CEF Browser, Yubikey authentication window appears. Enter **Username** and **Password** to Sign In if already registered. If not registered, registration page displays.

5. On “Set up multifactor authentication” window, click **Configure factor**.

6. On “Set up security key or biometric authenticator” window, click **Enroll**.

7. On “Set up multifactor authentication” window, check the enrolled factors and click **Finish**.
8. The connection is established and the connection details display.
Using Unified Pulse Secure Client with PZTA

- PZTA Overview ........................................................................................................................................ 120
- On-Demand and Simultaneous Connection Handling. ............................................................................. 120
- Disabling the PZTA Connection ............................................................................................................. 121
- Dynamic Policy Update and CARTA ....................................................................................................... 123
- Enrolling a User Device. .......................................................................................................................... 124

PZTA Overview

Pulse Secure provides a PZTA-ready version of the Pulse Client software required for end-user devices to be able to connect to your secure applications and resources.

Pulse Client connects to PZTA services, by default, through an on-demand connection basis and can handle multiple simultaneous PZTA and non-PZTA connections. To learn more, see On-Demand and Simultaneous Connection Handling.

Pulse Client maintains communication with the PZTA Controller to continuously-enable synchronization of policy and configuration updates. Through this mechanism, user requests to access resources and applications are subject to continuous assessment for risk and authorization. For more details, see Dynamic Policy Update and CARTA.

To learn more about enrolling user devices for use with PZTA, see Enrolling a User Device.

On-Demand and Simultaneous Connection Handling

While active, Pulse Client maintains two connection channels for PZTA services, a control channel to the PZTA Controller, and a data channel to your PZTA Gateways. For more details on networking considerations when deploying Gateways, see Working with Gateways.

The control channel connection to the PZTA Controller is activated when Pulse Client is started up and remains in an always-on state, silently in the background. If Pulse Client is able to locate a valid session cookie from an earlier session, the connection is re-established automatically. If no valid cookie is present, Pulse Client requests re-authentication from the user. The PZTA Controller connection is terminated when Pulse Client is shut down.

Pulse Client creates data channel connections to PZTA Gateways as an on-demand service. That is, connections to resources and applications controlled by PZTA Gateways become active only when required, and the connection is suspended after a period of inactivity. The user remains unaware of the connection state, unless re-authentication becomes necessary. As a user makes a request for a resource, Pulse Client transitions automatically from disconnected to connected. The connection remains in this state for the duration of the session, or until one of the following events occurs:

- An idle time-out occurs (after 5 minutes)
- The connection is actively placed in a disconnected state
• Pulse Client is shut down

To avoid the data channel being reconnected unnecessarily, non-PZTA DNS traffic is redirected to the device's physical network adapter.

Applicable Pulse Client versions can manage simultaneous connections with the PZTA Controller, and with other Pulse Secure services such as Pulse Connect Secure (PCS). While PCS connections must be activated and deactivated by the user, connections to PZTA are provided on-demand, as mentioned. Therefore, a PZTA connection in the Pulse Client does not provide the same Connect and Disconnect controls. Instead, PZTA connections include only a ZTA button to provide access to the PZTA Applications page. If this button is active, the connection to the Controller has been established. If the button is inactive, the connection to the Controller has not yet been established, or a communication problem has occurred. In this case, access to your applications is prevented.

When running active connections to both PZTA and PCS simultaneously, note that the following PCS features are not supported:

• Route Monitoring
• Traffic Enforcement
• Stealth Mode
• Always on VPN/LockDown
• Location awareness
• IPv6 support

Disabling the PZTA Connection

Pulse Client additionally provides the ability to actively disable the on-demand connection feature. Use of this facility disables the PZTA connection, avoiding the scenario where Pulse Client attempts to repeatedly request authentication even after the user might be unable to authenticate due to too many failed attempts, or where the user just does not require access to any PZTA-controlled resources during that session.

If a user attempts to request a PZTA-controlled resource during the period a PZTA connection is disabled, the request fails. Other Pulse Client connections are unaffected.

For Pulse Client on macOS and Windows, click Disconnect in the Pulse Client connection list context menu. Right-click a PZTA connection profile to see the available options.

Figure 1 Manually disabling a PZTA connection through the Pulse Client connection list context menu

For Pulse Client on macOS and Windows, click Disconnect from the System Tray Pulse icon. View the sub-menu for the PZTA connection you want to disconnect.
For Pulse Client on macOS and Windows, click Disconnect through the Pulse Client application menu. Open Pulse Client and select the PZTA connection profile. Then click **File > Connections > Disconnect**.

**Figure 3** Manually disabling a PZTA connection through the Pulse Client application menu

By setting the PZTA connection to be disconnected, Pulse Client suspends both the control channel and the data channel (where either are active). If the control channel was previously logged-in to the PZTA Controller, this remains the case to facilitate session resumption through a subsequent reconnect.

**Note:** The disconnect feature is not activated by clicking or tapping Cancel in the PZTA authentication dialog. Canceling an authentication request triggers a timeout interval, after which Pulse Client re-displays the authentication dialog. The disconnect feature instead disables the authentication request process until the user manually reinstates it.

To reinstate the PZTA connection on macOS and Windows devices, use the Launch ZTA option in the Pulse Client system tray menu or tap the **ZTA** button in the PZTA connection profile in the Pulse Client application.

If the existing session cookie is still valid, the control channel is re-established. If the session is now invalid, Pulse Client prompts the user for their PZTA credentials as normal. On successful re-establishment of the PZTA session, the user is presented with the PZTA End User Portal in the default browser.

When restarting Pulse Client, PZTA connections default to being on-demand services. That is, a previously disabled PZTA connection is re-enabled when Pulse Client starts.
Dynamic Policy Update and CARTA

To complement the zero-trust approach, PZTA supports dynamic policy updates and CARTA (Continuous Adaptive Risk and Trust Assessment) for your end user devices. This framework establishes an approach of continuous assessment and updating of secure access policies on the client, without the requirement to disconnect and reconnect to establish an updated authorization posture.

As your policies, applications, and authentication configuration are updated by the administrator on the PZTA Controller, changes are synchronized out to client devices dynamically and take effect immediately. Pulse Client ensures that any application updates are applied and any new authentication requirements are met before continuing the session, providing the end user with a seamlessly-updated experience. This method ensures that Pulse Client is always updated at the point of change, and not just when establishing a connection to a PZTA Gateway to access an affected resource.

The CARTA implementation in Pulse Client means that the security posture of the end user is continuously assessed in conjunction with policies configured in the Controller, with allow or deny decisions enforced through dynamic assessment and updating of the current policy set. Where application access is denied or restricted, Pulse Client informs the user of any access restrictions or policy contravention at the point of use. For example, the PZTA Desktop Client Home page updates to provide visual cues with applicable error messages whenever a specific application becomes unavailable:

Figure 4 FIGURE 15 Access restricted to the “myhealth” application

By hovering your pointer over the warning symbol in the inactive application, PZTA provides an explanatory message.

Furthermore, user attempts to access a restricted web resource in a browser trigger a CARTA response with Pulse Client presenting a pop-up resource blocked message:
Figure 5    Pulse Client prevents browser access to a particular web resource where a device policy has been breached

Pulse Client implements a no-repeat interval for resource blocked messages of 2 minutes, to avoid a user repeatedly seeing the same pop-up message for every browser request for the same restricted resource. While the resource remains blocked to further access attempts, no further messages are displayed by Pulse Client until after 2 minutes has elapsed. You can force Pulse Client to continue hiding blocked resource messages indefinitely by right-clicking the connection in the Pulse Client dialog and selecting **Disable Block Messages**. To re-enable showing blocked resource messages, select **Enable Block Messages**.

**Enrolling a User Device**

To use PZTA, end users must first enroll their devices with the PZTA Controller. This process installs the Pulse Client software and establishes a connection to the Controller in order to obtain policies and details for a user’s authorized resources.

Pulse Client uses this configuration to establish a secure connection to the PZTA Gateways you deploy to control access to your applications. Through this process, the user is provided a seamless connection to the resources they need and is never aware of the location or extent of the organization’s application infrastructure.

A new user might arrive at this scenario from one of the following routes:

- An existing Pulse Secure user, with a previous Pulse Client connection to Pulse Connect Secure (PCS) or similar, see Existing Pulse Client Users.
- A first time PZTA user, with no previous Pulse Secure software installed, see “Enrolling First Time Users”.
- An existing PZTA user enrolling a new device, or upgrading a previous version of Pulse Client, see “Enrolling Existing PZTA Users”.

**Existing Pulse Client Users**

Your users might have a previous version of Pulse Client installed if, for example, they are existing PCS users.
To enroll existing PCS users into PZTA, a PCS administrator must first push out the PZTA-ready edition of the client software to the user base. An admin uploads the new client software to the PCS server and activates the PZTA-ready version of Pulse Client from the PCS management console in the same way as any other version. This process ensures that when your users next activate a Pulse Client connection to the server, their device is prompted to download and install the new version.

For more details on this process, see the Pulse Connect Secure documentation at https://www.pulsesecure.net/techpubs.

After the new PZTA-ready version of Pulse Client is installed, the user can configure a PZTA connection using the same process used for other, existing, connections. To create a PZTA connection, compatible Pulse Client versions offer a specific connection type: “Zero Trust Access”.

The tenant administrator must then supply the PZTA enrollment URL to their users to create the new PZTA connection.

Enrolling First Time Users

When enrolling a new device, an authorized user contacts the PZTA Controller to activate an initial first-time enrollment of their client device. The Controller responds to a valid enrollment request by activating a download of Pulse Client along with a suitable client certificate.

After Pulse Client is installed, a secure connection request is attempted with the Controller. The request is validated against the designated authentication policy applicable to that combination of user and device and, where successful, a connection profile is downloaded to the client. This profile enables Pulse Client to set up a secure tunnel directly to the PZTA Gateway serving the resource set the client is authorized to view.

Enrolling Existing PZTA Users

After you have enrolled the new device, Pulse Client is installed and configured with the policies and settings relevant to the device type. Your application and resource access rights should be duplicated to the new device.

Note: If a user device is currently using a Beta version of the PZTA-ready Pulse Client, Pulse Secure advises to remove the PZTA connection from Pulse Client and to re-perform the enrollment procedure through a web browser. For more details, see your support representative.