Pulse Secure Client

Linux Quick Start Guide

Release, Build	9.0R1, 571
Published	April, 2018
Document Version	1.1

Pulse Secure, LLC
2700 Zanker Road, Suite 200
San Jose, CA 95134

This product includes the Envoy SNMP Engine, developed by Epilogue Technology, an Integrated Systems Company. Copyright © 1986-1997, Epilogue Technology Corporation. All rights reserved. This program and its documentation were developed at private expense, and no part of them is in the public domain.

This product includes memory allocation software developed by Mark Moraes, copyright © 1988, 1989, 1993, University of Toronto.

This product includes FreeBSD software developed by the University of California, Berkeley, and its contributors. All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by the Regents of the University of California. Copyright © 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994. The Regents of the University of California. All rights reserved.

GateD software copyright © 1995, the Regents of the University. All rights reserved. Gate Daemon was originated and developed through release 3.0 by Cornell University and its collaborators. Gated is based on Kirton's EGP, UC Berkeley's routing daemon (routed), and DCN's HELLO routing protocol. Development of Gated has been supported in part by the National Science Foundation. Portions of the GateD software copyright © 1988, Regents of the University of California. All rights reserved. Portions of the GateD software copyright © 1991, D. L. S. Associates.

This product includes software developed by Maker Communications, Inc., copyright © 1996, 1997, Maker Communications, Inc.

Pulse Secure, Pulse and Steel-Belted Radius are registered trademarks of Pulse Secure, LLC. in the United States and other countries. The Pulse Secure Logo, the Pulse logo, and PulseE are trademarks of Pulse Secure, LLC. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

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

Products made or sold by Pulse Secure or components thereof might be covered by one or more of the following patents that are owned by or licensed to Pulse Secure: U.S. Patent Nos. 5,473,599, 5,905,725, 5,909,440, 6,192,051, 6,333,650, 6,359,479, 6,406,312, 6,429,706, 6,459,579, 6,493,347, 6,538,518, 6,538,899, 6,552,918, 6,567,902, 6,578,186, and 6,590,785.

Pulse Secure Client for Linux Quick Start Guide.

The information in this document is current as of the date on the title page.

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. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

© 2018 by Pulse Secure, LLC. All rights reserved

Contents

REQUESTING TECHNICAL SUPPORT 5
PART 1 6
INTRODUCTION 6
Overview 7
Supported Platforms 7
Server Platform Compatibility 7
Supported Features 7
Limitations 8
PART 2 9
CONFIGURING SERVER VPN POLICY 10
INSTALLING PULSE SECURE CLIENT FOR LINUX 11
USING PULSE LINUX CLIENT UI 12
Adding VPN Connections 13
Modifying VPN Connection 14
Deleting VPN Connection 15
Launching VPN Connection 16
Initiating VPN Connection 16
Terminating VPN Connection 17
Diagnostics and Status 18
Uploading the Pulse Secure Client Log Files 18
Advanced Connection Details 21
About Pulse Linux Client 21
USING PULSE LINUX CLIENT COMMAND LINE 23
Initiating VPN connection 23
Launch the Command Line Client 23
Terminating VPN connection 23
Diagnostics and Status 23
Uploading Pulse Secure Client Log Files 23
Checking Pulse Secure Client Status 24
MANAGING CERTIFICATES ON LINUX 25
HOST CHECKING SUPPORT ON LINUX 27
Configuring PCS Host Checker Policy for Pulse Linux Client 27
DIAGNOSTICS AND STATUS 29
SAML SUPPORT 30
CERTIFICATE AUTHENTICATION SUPPORT 31
Configuring Client Certificate in Pulse Connect Secure 31
Configuring Authentication with the Certificate Server 32
Client Certificate Installation 34
Certificate Types Supported 34

© 2018 by Pulse Secure, LLC. All rights reserved

Public Certificates....34
Private Keys....34
PFX file (Contains both Private Key and Public Keys)....34
DEFAULT CERTIFICATE SELECTION....35
REVISION HISTORY....37

© 2018 by Pulse Secure, LLC. All rights reserved

Requesting Technical Support

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

- Product warranties—For product warranty information, visit https://www.pulsesecure.net
- Find product documentation: https://www.pulsesecure.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: https://www.pulsesecure.net/support

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://www.pulsesecure.net/support.
- Call Phone: 1-844-751-7629 (Toll Free, US).

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

© 2018 by Pulse Secure, LLC. All rights reserved

Part 1

Introduction

- Overview
- Supported Platforms
- Server Platform Compatibility
- Supported Features
- Limitations

© 2018 by Pulse Secure, LLC. All rights reserved

Overview

Pulse Secure client for Linux provides secure connectivity between a device running Linux and Pulse Connect Secure. Pulse Secure client for Linux is available from the PCS Admin Installers Page. After installing the Pulse Secure client VPN package on a Linux device, the user can configure a connection and establish Layer 3 VPN communications.

Configuration on the Pulse Connect Secure gateway to support Pulse Secure clients for Linux is the same as that of Pulse for Windows and Mac OSX. Use the sign-in policies, authentication realms, roles and VPN tunnel policies to define authentication and access permissions. A typical Pulse server configuration for Linux access is to create a realm, a role and a remediation role that are designed for Linux users.

Supported Platforms

Table 1: lists supported Linux Platforms

Platform	Qualified / Compatible
Ubuntu 16.04 LTS 32 bit, 64 bit	Qualified
Ubuntu 15.04 32 bit, 64 bit	Qualified
Ubuntu 14.04 LTS 32 bit, 64 bit	Qualified
CentOS 6.9 32 bit, 64 bit	Qualified
Fedora 27, 32 bit, 64 bit	Qualified
Fedora 23, 32 bit, 64 bit	Qualified
RHEL7 64 bit	Qualified
CentOS7 64 bit	Qualified
Debian 8 32 bit, 64 bit	Qualified
Debian 9 32 bit, 64 bit	Qualified

Server Platform Compatibility

This topic provides the server platforms that were tested with the current release of the Pulse Secure Desktop Clients for Linux.

Table 2: Server Compatibility for Pulse Secure Client for Linux

Product	Qualified	Compatible
Pulse Connect Secure	8.3R4, 8.3R3, 8.3R2, 8.3R1, 8.2Rx, 8.1Rx	8.0Rx, 7.4RX

Note: 8.0Rx and 7.4RX with Host Checker is not supported.

Supported Features

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

- Pulse Linux Client Usability Improvements
  - The Pulse Linux client will now remain connected if the UI is closed. When the UI is re-launched, updated statistics are presented.
- SAML support for Linux
- Java Free Pulse Linux Client Support
- 64-bit Operating Systems Support
- Multi-Factor Authentication (MFA) Support
- Host Checker
- Session Timeout Warning Feature

© 2018 by Pulse Secure, LLC. All rights reserved

- Command Line Support
- RPM/DEB Package Manager Support
- Pre- and post-authentication sign-in notification messages
- Client Certificate Authentication Support.

Limitations

The following are the limitations to the Pulse Secure Client for Linux.

- Refer to KB40238 for Pulse Secure Linux on CentOS clients cannot connect using TLS 1.2

  FIPS or NDcPP mode is not supported on Pulse Linux Clients

  Note: The certificate authentication through UI will be supported only on the machines using libsoup 2.48 and above. Centos 6.4 and Ubuntu 14.04 doesn't meet the above requirements.

© 2018 by Pulse Secure, LLC. All rights reserved

Part 2

- Configuring Server VPN Policy
- Installing Pulse Secure Client for Linux
- Using Pulse Linux Client UI
- Using Pulse Linux Client Command Line
- Managing Certificates on Linux
- Host Checking Support on Linux
- SAML Support
- Certificate Authentication Support

© 2018 by Pulse Secure, LLC. All rights reserved

Configuring Server VPN Policy

The Pulse Secure client enables you to secure your company resources using authentication realms, user roles and resource policies. For complete information on the Pulse Connect Secure gateway, see the Pulse Connect Secure documentation.

The Pulse Connect Secure gateway checks the authentication policy defined for the authentication realm. The user must meet the security requirements that are defined for a realm's authentication policy. At the realm level, you can specify security requirements based on various elements, such as the user's source IP address or the possession of a client-side certificate. If the user meets the requirements specified by the realm's authentication policy, the gateway forwards the user's credentials to the appropriate authentication server. If this server successfully authenticates the user, then the gateway evaluates the role-mapping rules defined for the realm to determine which roles to assign to the user.

The following is a generalized example of configuring a Pulse Connect Secure gateway for the Pulse Client for Linux.

1. Click **Users > User Roles** and then either select an existing role (preferred) or create a new role.
2. If creating a new role, specify a name and optional description for the role, for example:

   Linux Users Role, Linux Users VPN Role.
3. Enable **VPN Tunneling** and **Save Changes**.
4. Click **Users > User Realms** to create new realm or select an existing realm.
5. Configure and save your options on the General and Authentication Policy tabs.
6. To Sign In, enable primary/secondary authentication by selecting Servers from Authentication Server list.

   On the Role Mapping tab, click New Rule to create a new role-mapping rule.

   One option for a role-mapping rule is to create a custom expression that uses the user agent string to identify a Linux device. The Pulse Secure client for Linux user agent string has a form like this:

   ```
   DSClient; PulseLinux
   ```

   You can use all or part of the string in a custom expression that uses the userAgent variable. For example:

   ```
   userAgent = '*PulseLinux*'
   ```
7. Select the role that you created earlier for the Linux users, add it to the Selected Roles list.
8. Click **Save Changes.**

User sign-in policies determine the realm(s) that users can access.

1. To create a new sign-in policy, click New URL. Or, to edit an existing policy, click a URL in the User URLs column.
2. Modify an existing sign-in page or create a new one using options in the **Authentication > Signing In > Sign-in Pages** page of the admin console.
3. Specify a sign-in policy that associates a realm, sign-in URL, and sign-in page using settings in the **Authentication > Signing In > Sign-in Policies** page of the admin console. To create or configure user sign-in policies, Click **New URL** in **Authentication > Signing In > Sign-in Policies**.
4. Under Authentication realm, specify which realm(s) map to the policy, and how users and administrators should pick from amongst realms.
5. Click **Save Changes**.

© 2018 by Pulse Secure, LLC. All rights reserved

Installing Pulse Secure Client for Linux

The Pulse Secure client for Linux currently is designed only for CLI-based installation.

The Installation Command:

- Debian – based installation

  dpkg -i <package name>
- RPM – based installation

  rpm -ivh <package name>

The Upgrade Command:

- Debian – based installation

  dpkg -i <package name>
- RPM – based installation

  rpm -Uvh <package name>

The Uninstallation Command:

- Debian – based installation

  dpkg -r <package name>
- RPM – based installation

  rpm -e <package name>

The Dependencies Installation Command:

- 64 bit machines

/usr/local/pulse/PulseClient_x86_64.sh install_dependency_packages

- 32 bit machines:

/usr/local/pulse/PulseClient.sh install_dependency_packages

Note: On Ubuntu distributions, installing through Ubuntu Software Center is supported.

© 2018 by Pulse Secure, LLC. All rights reserved

Using Pulse Linux Client UI

To launch UI from the Desktop

Pulse Linux Client UI can be launched by searching for Pulse Secure Icon under Applications List.

Figure 1 Pulse Secure Client for Linux Application

To launch UI from the Terminal

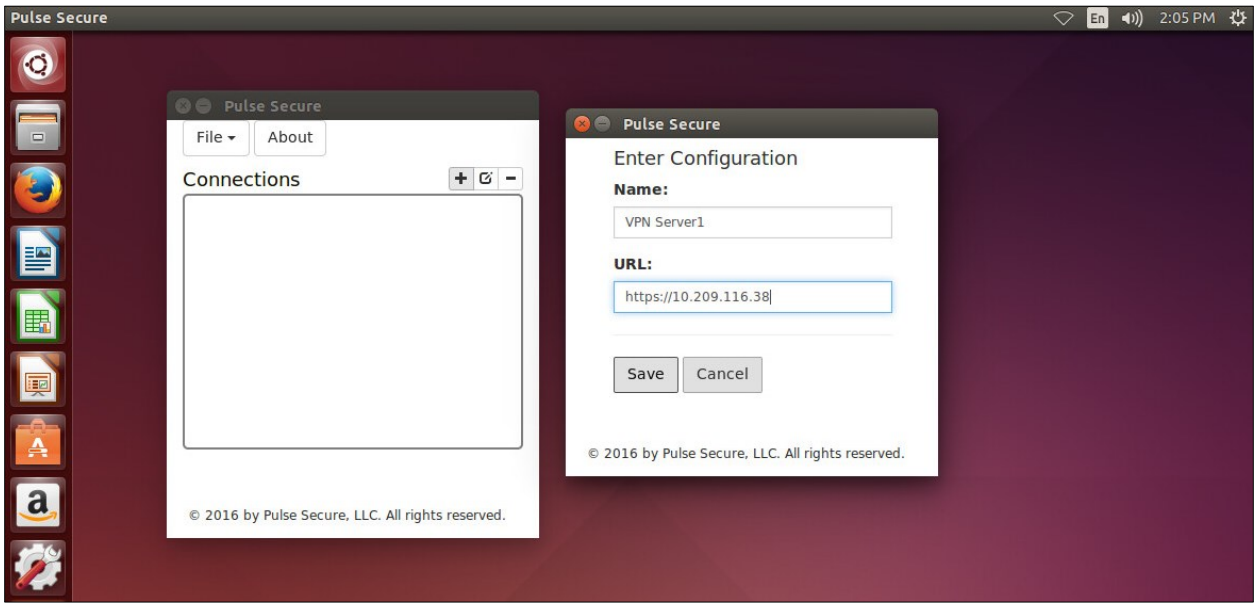
1. Append /usr/local/pulse to LD_LIBRARY_PATH
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/pulse
2. Launch the UI by executing the below command
   /usr/local/pulse/pulseUi

© 2018 by Pulse Secure, LLC. All rights reserved

Adding VPN Connections

To create a Pulse Secure VPN connection on a Linux device:

Figure 2 Adding VPN Connection



- Click on the Add (+) option on the top-right-hand corner of the main Pulse Secure Linux client screen.
  1. In the Name field, specify the name for the Pulse Connect Secure gateway.
  2. In the 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.)
- New VPN connection appears in the VPN list after user click the save button on the above screen

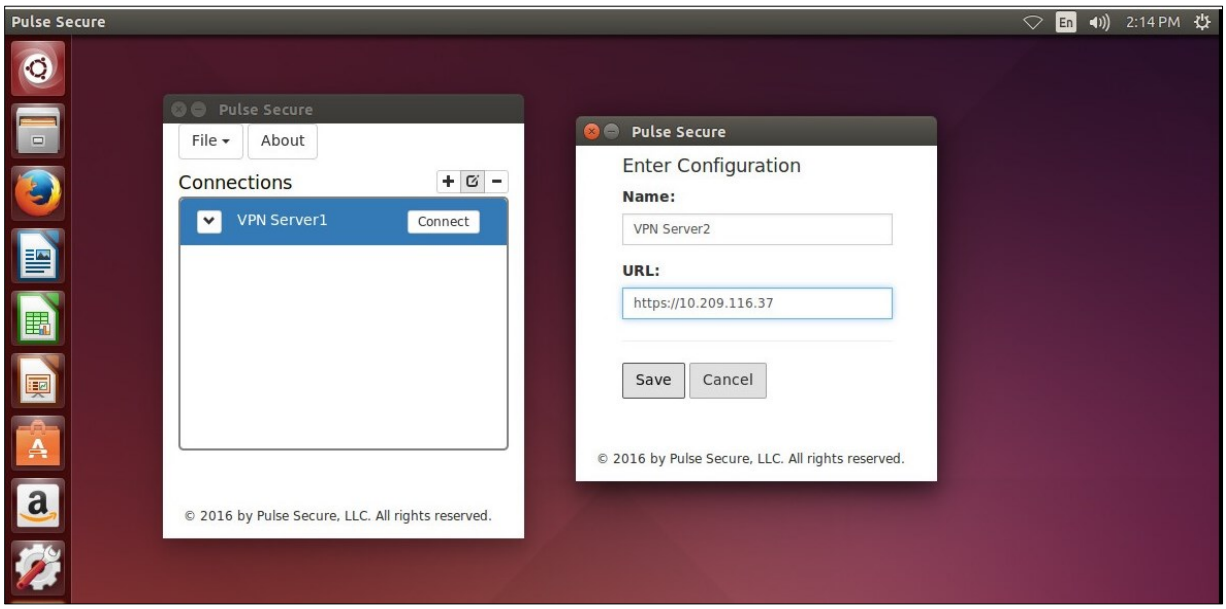
  The user can tap connect to initiate a VPN connection. The VPN connection state is indicated in the VPN dropdown menu on the VPN list.

© 2018 by Pulse Secure, LLC. All rights reserved

Modifying VPN Connection

To modify a Pulse Secure VPN connection on a Linux device:

Figure 3 Modifying VPN Connection



- Select the VPN connection and Click on the edit ([]) option on the top-right-hand corner of the main Pulse Secure Linux client screen.
  1. In the Name field, specify the name for the Pulse Connect Secure gateway.
  2. In the 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.)
- Modified VPN connection appears in the VPN list after user click the save button on the above screen

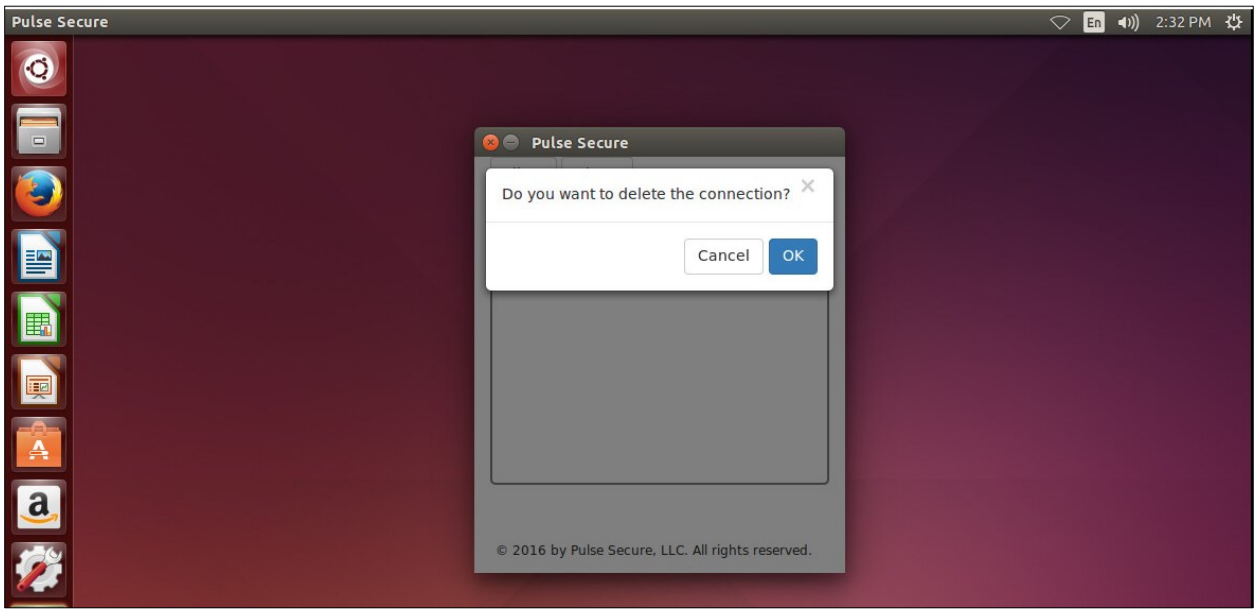
  The user can tap connect to initiate a VPN connection. The VPN connection state is indicated in the VPN dropdown menu on the VPN list.

© 2018 by Pulse Secure, LLC. All rights reserved

Deleting VPN Connection

To delete a Pulse Secure VPN connection on a Linux device:

Figure 4 Deleting VPN Connection



1. Select the VPN connection and Click on the delete (-) option on the top-right-hand corner of the main Pulse Secure Linux client screen.
2. VPN connection is removed from the VPN list after user click the OK button on the above screen.

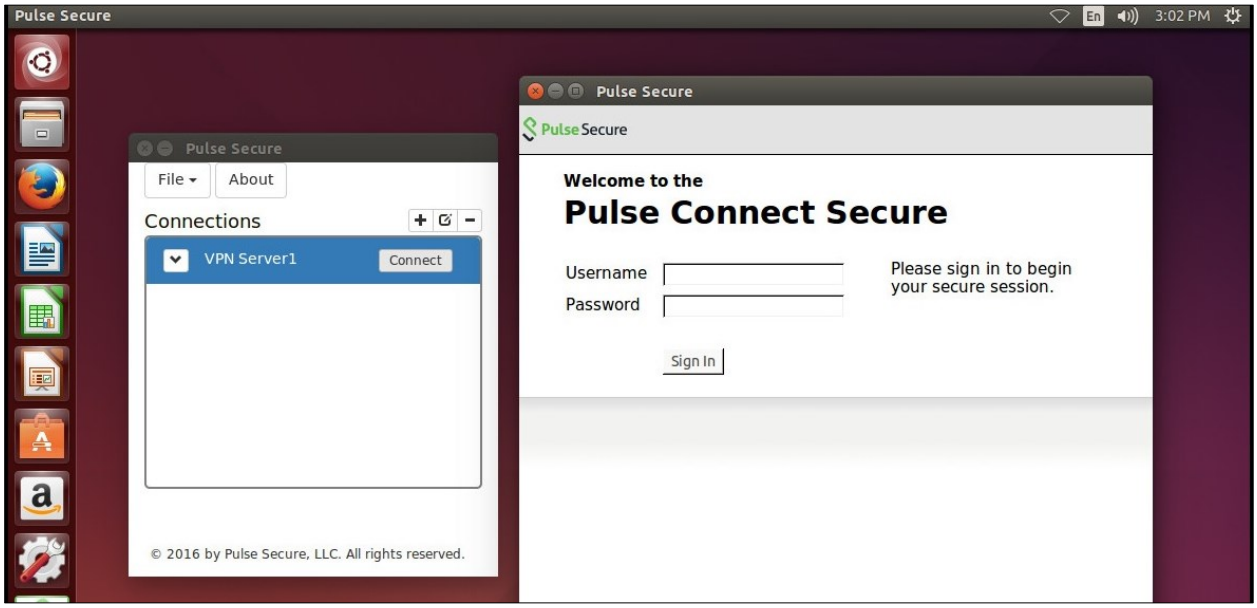
© 2018 by Pulse Secure, LLC. All rights reserved

Launching VPN Connection

Initiating VPN Connection

To initiate a Pulse Secure VPN connection on a Linux device:

Figure 5 Initiating VPN Connection



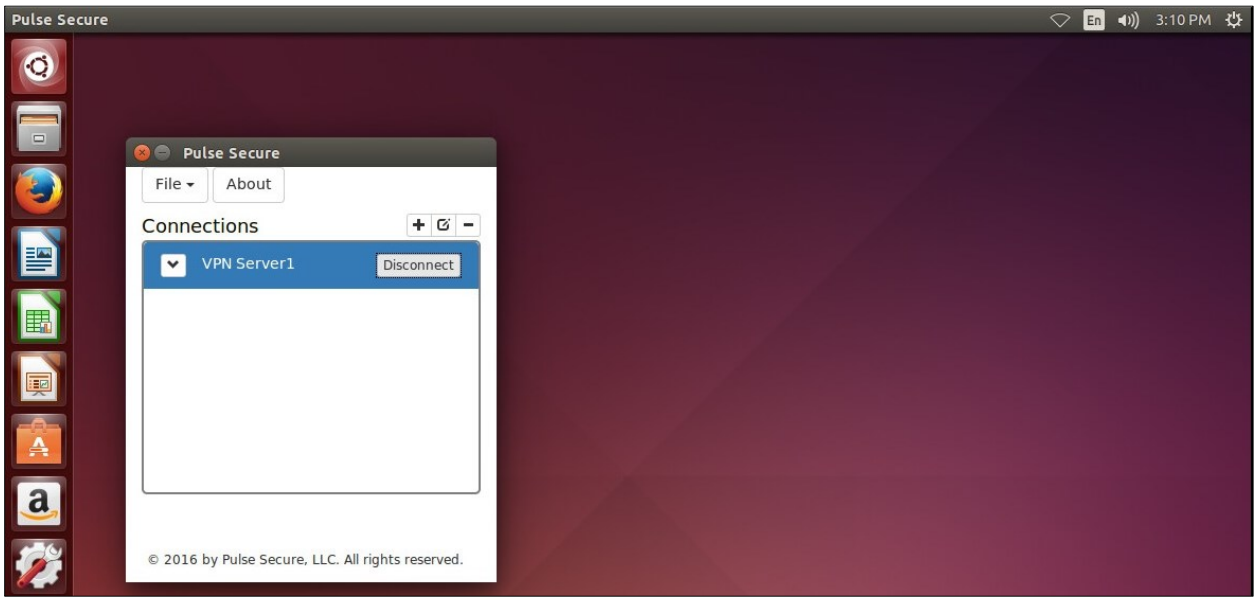
1. Select the VPN connection and Click on the Connect option of the VPN connection entry in the Main Pulse Secure Linux client screen.
2. New window opens up to continue authentication process based on the authentication method configured for the realm.

© 2018 by Pulse Secure, LLC. All rights reserved

Terminating VPN Connection

To terminate a Pulse Secure VPN connection on a Linux device:

Figure 6 Terminating VPN Connection



1. Select the VPN connection and Click on the Disconnect option of the VPN connection entry in the Main Pulse Secure Linux client screen.

© 2018 by Pulse Secure, LLC. All rights reserved

Diagnostics and Status

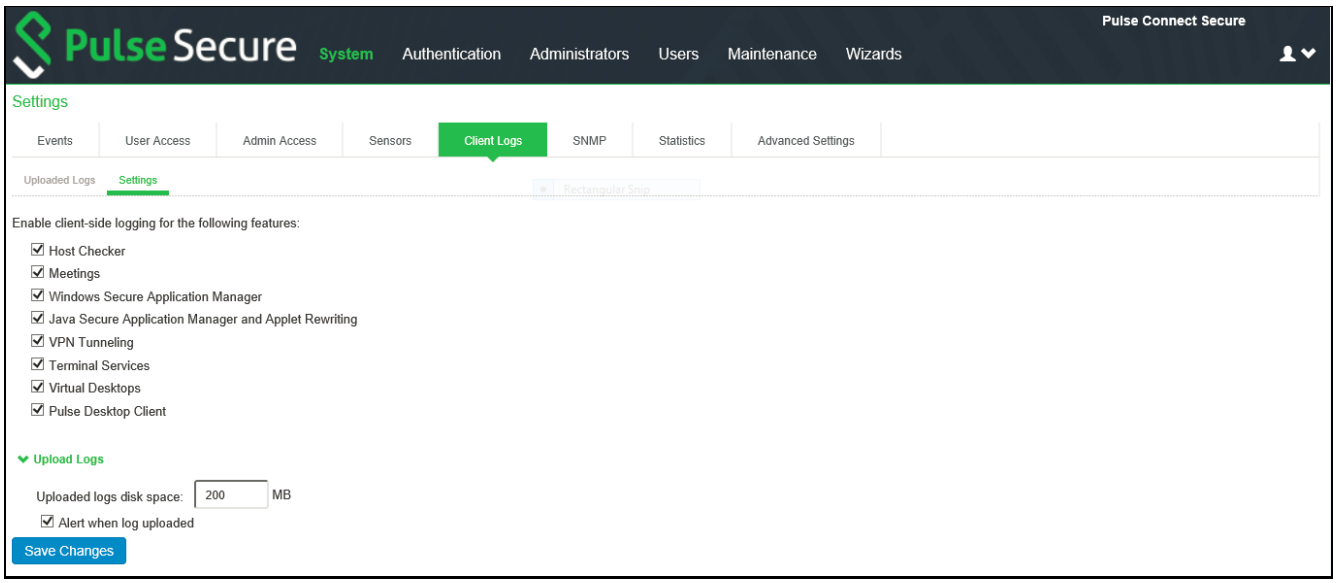
Uploading the Pulse Secure Client Log Files

The Pulse Secure desktop client for Linux makes it easy to transmit diagnostic log bundles to PCS gateways for analysis by system administrators.

The Pulse Connect Secure admin must enable which clients can send log files by traversing the following menus in the admin console and clicking on the Pulse Client:

- System > Log/Monitoring > Client Logs > Settings

Figure 7 Enabling Upload logs Admin Option

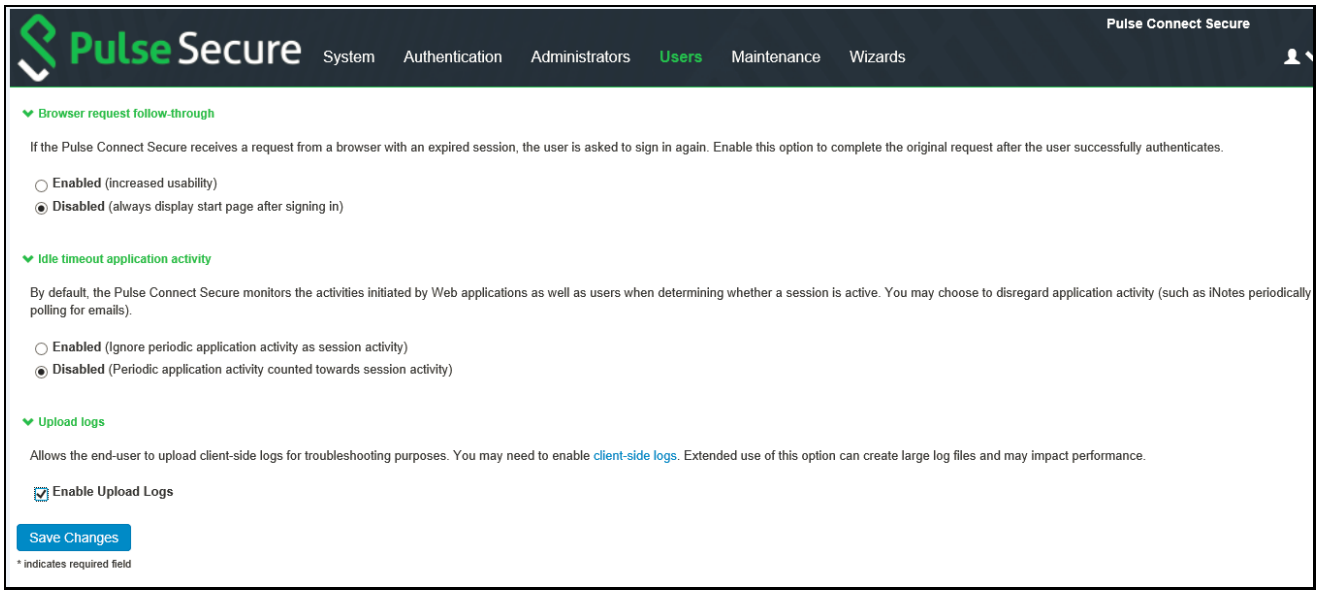


The Pulse Connect Secure admin must enable upload logs feature by traversing the following menus in the admin console and clicking on the Checkbox:

- User Roles -> <Role> -> General -> Session Options

© 2018 by Pulse Secure, LLC. All rights reserved

Figure 8 Enabling Upload logs Admin Option



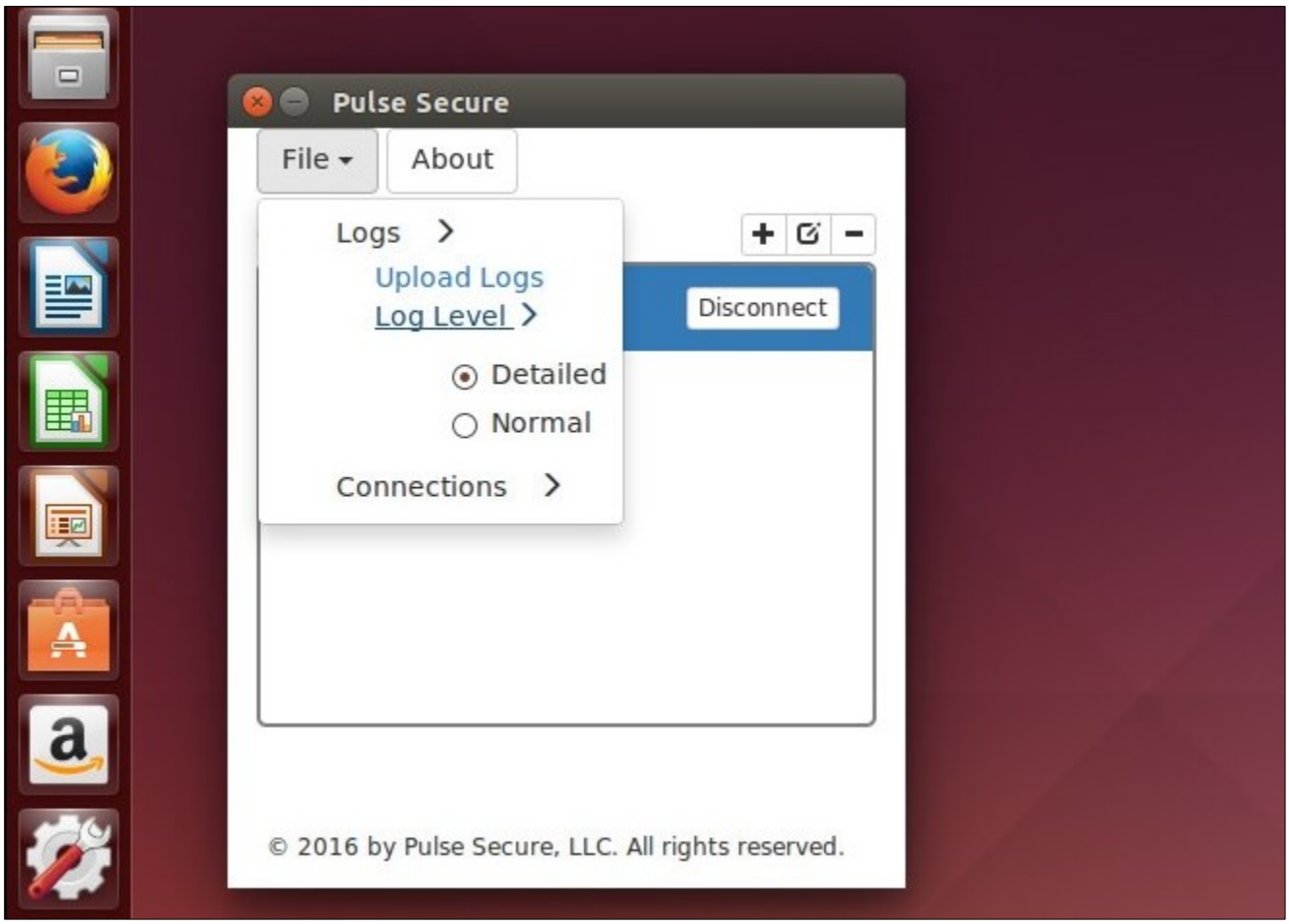
Pulse Secure Linux allows user to select normal/detailed logs to store for organizing issues.

To change the log level, run the following step from the desktop client user interface:

- File -> Logs ->Log Level

© 2018 by Pulse Secure, LLC. All rights reserved

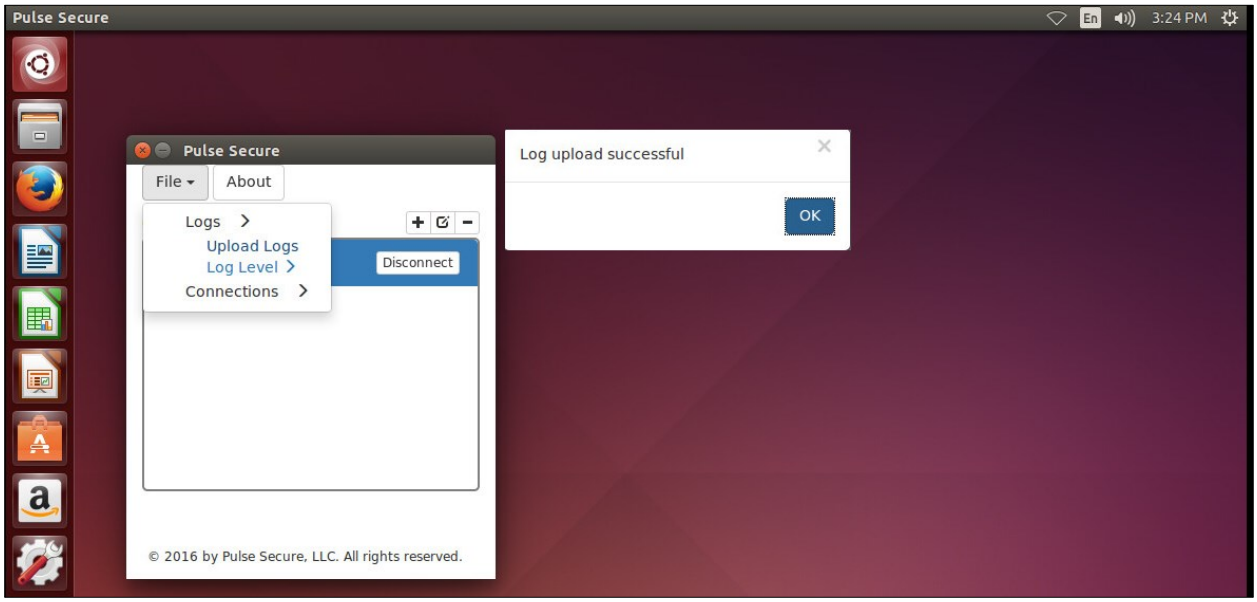
Figure 9 Enabling Detailed log level



To send a log bundle to the PCS, when a VPN connection is selected, run the following step from the desktop client user interface:

- File -> Logs -> Upload.

Figure 10 Uploading Debug Logs

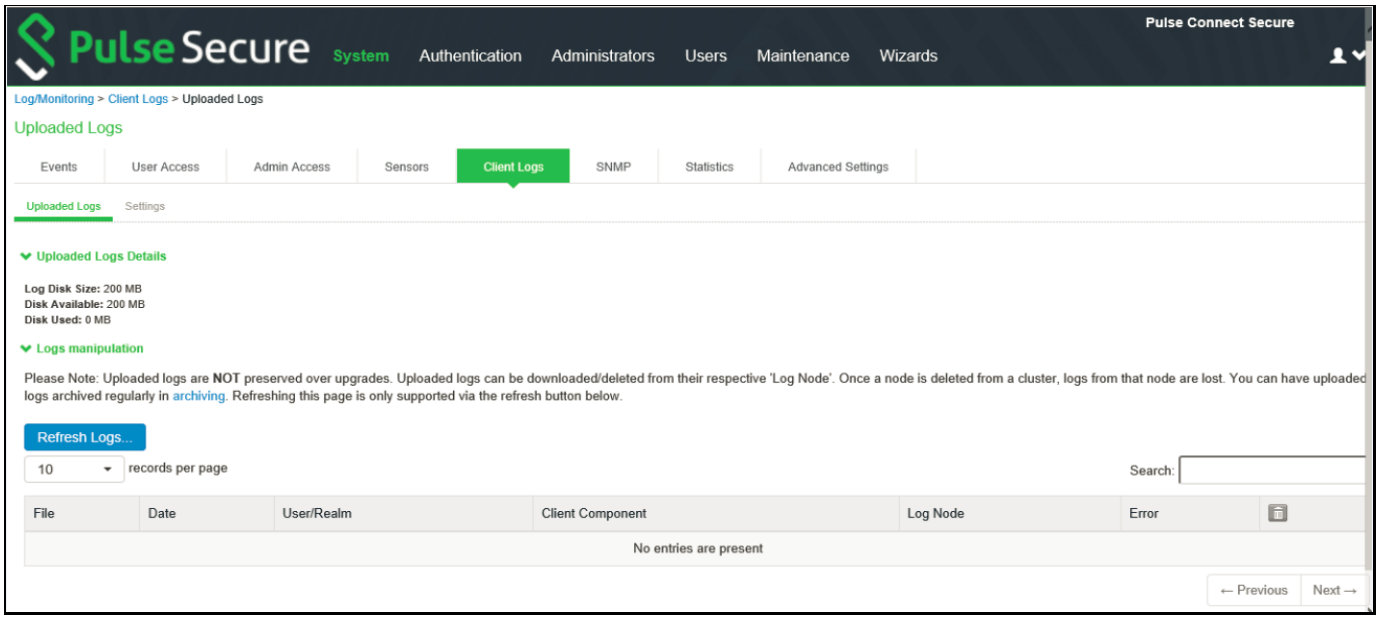


Once this work is done, the system administrator can view uploaded logs in the administrative console here:

© 2018 by Pulse Secure, LLC. All rights reserved

- System > Log/Monitoring > Client Logs > Uploaded Logs

Figure 11 Admin Uploaded logs page

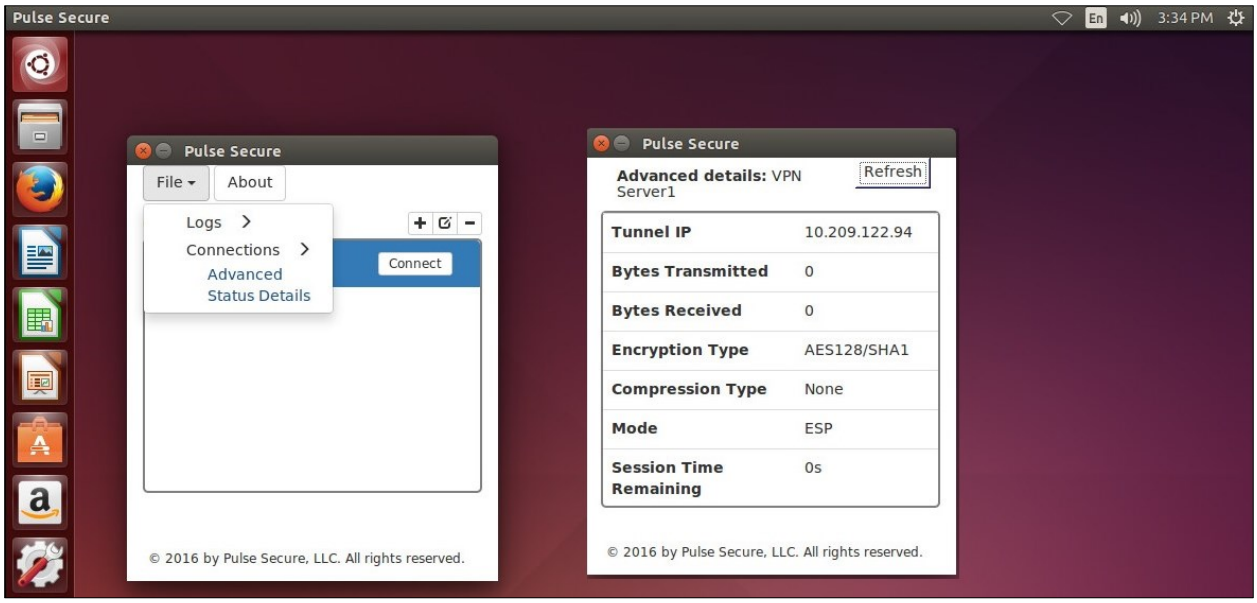


Advanced Connection Details

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

To view advanced connection details, click on File > Connections > Advanced Status Details.

Figure 12 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 on About button on main Pulse Linux client UI.

© 2018 by Pulse Secure, LLC. All rights reserved

Figure 13 About Pulse Linux client

© 2018 by Pulse Secure, LLC. All rights reserved

Using Pulse Linux Client Command Line

Initiating VPN connection

Pulse Secure CLI client can be launched from Linux terminals. The usage information is also available in the readme file inside the package.

The following command gives the usage of the various options of the command line client:

```
/usr/local/pulse/PulseClient.sh -H
```

In the case of 64 bit machines, please execute the following command:

```
/usr/local/pulse/PulseClient_x86_64.sh -H
```

The PulseClient.sh script will install the 32 bit dependent packages if these packages are not already installed.

Prerequisites to Run the Command Line Client:

- Pulse Connect Secure (PCS, formerly SA/IVE) IP address or hostname
- VPN user name and password
- PCS certificate (Contact PCS administrator to get certificate in DER format)
- PCS sign-in URL
- Proxy details, if applicable, (IP address/hostname, proxy username and password)
- Realm name to connect

Launch the Command Line Client

Use the following command format to launch the command line client:

```
/usr/local/pulse/PulseClient.sh -h <PCS appliance IP/hostname> -u <vpn username> -p
<vpn password> -r <realm>
```

Example:

```
/usr/local/pulse/PulseClient.sh -h vpn.pulsesecure.net -u user1 -p PulseSecure -r users
```

Terminating VPN connection

Run the following command to terminate the command line client to terminate the VPN connection.

```
/usr/local/pulse/PulseClient.sh -K
```

Diagnostics and Status

Uploading Pulse Secure Client Log Files

Use the following command format to launch the command line client:

```
/usr/local/pulse/PulseClient.sh -h <PCS appliance IP/hostname> -u <vpn username> -p
<vpn password> -r <realm> -g
```

Example:

```
/usr/local/pulse/PulseClient.sh -h vpn.pulsesecure.net -u user1 -p PulseSecure -r
users -g
```

Note: -g option connects to Pulse Connect Secure (PCS) with the provided credentials and uploads the logs into Pulse Connect Secure.

© 2018 by Pulse Secure, LLC. All rights reserved

Checking Pulse Secure Client Status

Run the following command to see the status of the VPN connection:

```
/usr/local/pulse/PulseClient.sh -S
```

Note: The above command could take up to 30 seconds to reflect the current state of the Pulse client.

© 2018 by Pulse Secure, LLC. All rights reserved

Managing Certificates on Linux

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 to be stored as 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 (CentOS/RHEL/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. Enable the dynamic CA configuration feature:

   ```
   update-ca-trust force-enable
   ```

4. 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/
   ```

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

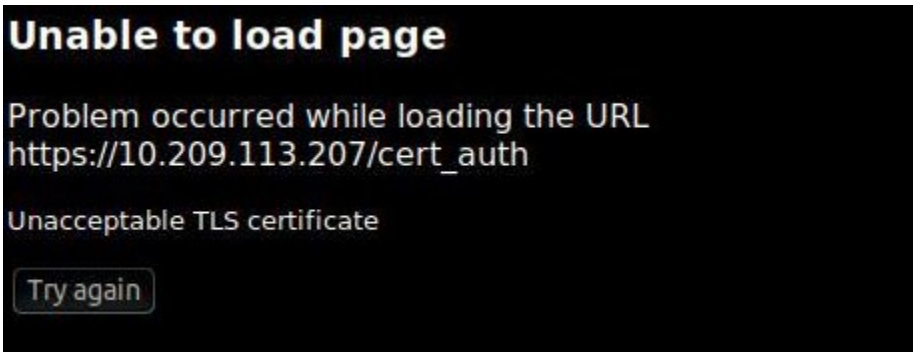
Note: Pulse Secure Linux client command line "-f" option is deprecated from 8.1R8 release.

Note: From 9.0 R1 release onwards, web browser invoked for establishing connections will display

© 2018 by Pulse Secure, LLC. All rights reserved

following error message, if the user visits or forwarded to the websites which have certificates not trusted by the machine.

Figure 14 Error Message



© 2018 by Pulse Secure, LLC. All rights reserved

Host Checking Support on Linux

All Host Checker rules are implemented through IMCs and IMVs based on the TNC open architecture. IMCs are software modules that Host Checker runs on the client machine. IMCs are responsible for collecting information, such as antivirus, antispyware, patch management, firewall, and other configuration and security information for a client machine.

Configuring PCS Host Checker Policy for Pulse Linux Client

To configure a Host Checker policy, perform these tasks:

1. Create and enable Host Checker policies through the Authentication > Endpoint Security > Host Checker page of the admin console.
2. In the admin console, select Authentication > Endpoint Security > Host Checker.
3. Under Policies, click New.
4. Enter a name in the Policy Name field and then click Continue. (Users see this name on the Host Checker remediation page if you enable custom instructions for this policy.)
5. Create one or more rules to associate with the policy.
6. Configure additional system-level options on the Authentication > Endpoint Security > Host Checker page of the admin console as necessary:
    a. If you want to display remediation information to users if they fail to meet the requirements of a Host Checker policy, configure remediation options through the Authentication > Endpoint Security > Host Checker page of the admin console.
    b. To change default Host Checker settings, configure settings through the Authentication > Endpoint Security > Host Checker page of the admin console.
7. Determine the level you that you want to enforce Host Checker policies:
    a. To enforce Host Checker policies when the user initially accesses the device, implement the policy at the realm level by selecting the policy at the Users > User Realms > Select Realm > Authentication Policy > Host Checker page of the admin console.
    b. To allow or deny users access to specific roles based on compliance with Host Checker policies, implement the policies at the role level by using the Users > User Roles > Select Role > General > Restrictions > Host Checker page of the admin console.
    c. To map users to roles based on their compliance with Host Checker policies, use custom expressions in the Users > User Realms > Select Realm > Role Mapping page of the admin console.
    d. To allow or deny users access to individual resources based on their compliance with Host Checker policies, use conditions in the Users > Resource Policies > Select Resource > Select Policy > Detailed Rules > Select|Create Rule page of the admin console.

Pulse Linux Client supports Files, Ports and Processes IMCs currently

- Ports—Use this rule type to control the network connections that a client can generate during a session. This rule type ensures that certain ports are open or closed on the client machine before the user can access the device. In the Ports configuration page:
    1. Enter a name for the port rule.
    2. Enter a comma delimited list (without spaces) of ports or port ranges, such as: 1234,11000-11999,1235.
    3. Select Required to require that these ports are open on the client machine or Deny to require that they are closed.
    4. Click Save Changes.
- Process—Use this rule type to control the software that a client may run during a session. This rule type

© 2018 by Pulse Secure, LLC. All rights reserved

ensures that certain processes are running or not running on the client machine before the user can access resources protected by the system. In the Processes configuration page:

1. Enter a name for the process rule.
2. Enter the name of a process (executable file), such as: good-app.exe.

   Note: For Linux systems, the process that is being detected must be started using an absolute path. You can use a wildcard character to specify the process name.

   For example: /usr/local/bin/pulseUi

3. Select Required to require that this process is running or Deny to require that this process is not running.
4. Specify the MD5 checksum value of each executable file to which you want the policy to apply (optional). For example, an executable may have different MD5 checksum values on a desktop, laptop, or different operating systems. On a system with OpenSSL installed—many Linux systems have OpenSSL installed by default—you can determine the MD5 checksum by using this command: openssl md5 <processFilePath>
5. Click Save Changes.

- File—Use this rule type to ensure that certain files are present or not present on the client machine before the user can access the device. You may also use file checks to evaluate the age and content (through MD5 checksums) of required files and allow or deny access accordingly. In the Files configuration page:

1. Enter a name for the file rule.
2. Enter the name of a file (any file type), such as: /tmp/bad-file.txt.
   a. You can use a wildcard character to specify the file name.

      For example: *.txt

   b. You can also use an environment variable to specify the directory path to the file. (You cannot use a wildcard character in the directory path.) Enclose the variable between the <% and %> characters.

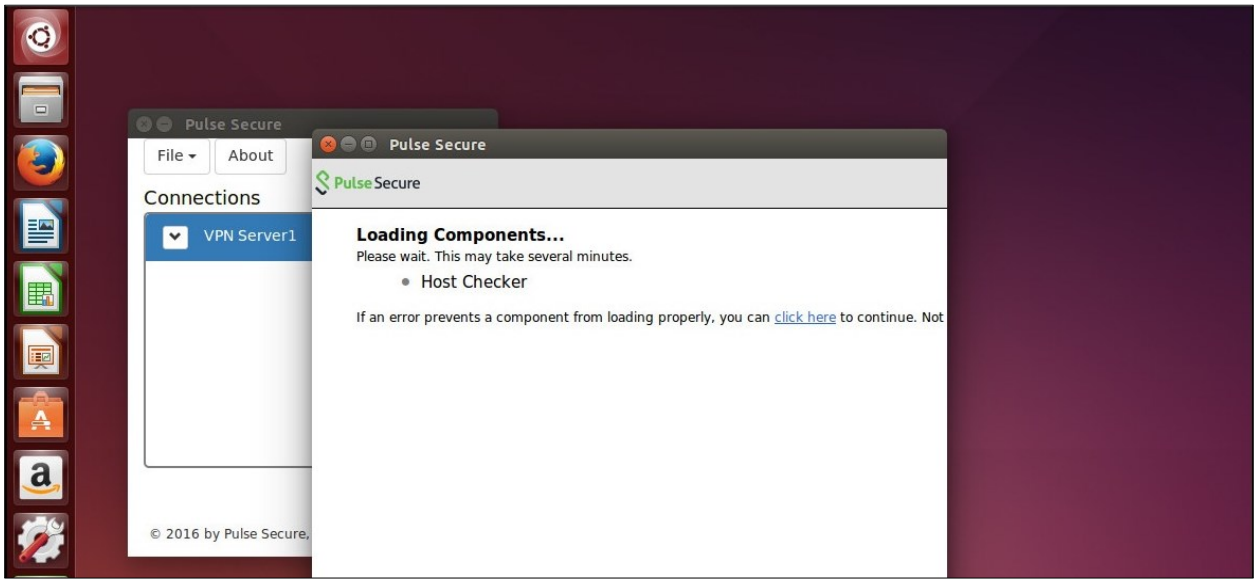
      For example: $FILEPATH\bad-file.txt
3. Select Required to require that this file is present on the client machine or Deny to require that this file is not present.

© 2018 by Pulse Secure, LLC. All rights reserved

Diagnostics and Status

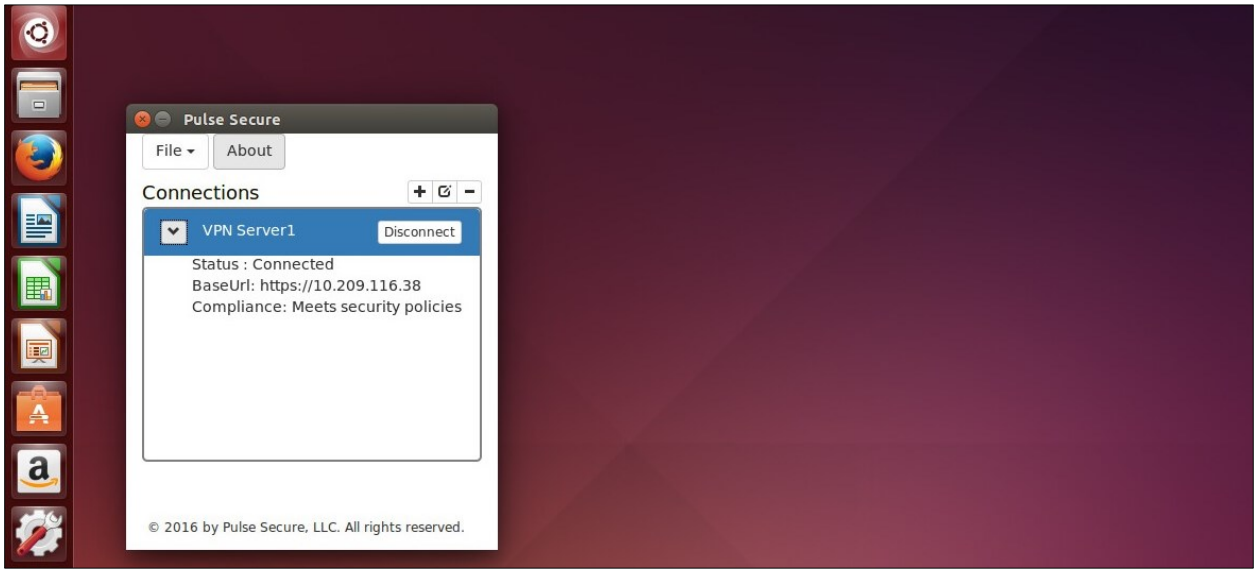
Pulse Linux Client updates the status of host checker in the VPN connection window

Figure 15 Host Checker Status on VPN Connection Window



Pulse Linux Client shows compliance status for connected session when user clicks on drop down option for the same VPN connection entry.

Figure 16 Host Checker Compliance Status Window



© 2018 by Pulse Secure, LLC. All rights reserved

SAML Support

SAML is an XML-based framework for communicating user authentication, entitlement and attribute information. It is used primarily to implement Web browser single sign-on (SSO). This feature allows users to authenticate over SAML.

This feature has been tested and qualified for the following Identity Providers (IdPs):

- G Suite
- Okta

For more information on SAML Support, see Pulse Secure Linux Client SAML Deployment Guide.

© 2018 by Pulse Secure, LLC. All rights reserved

Certificate Authentication Support

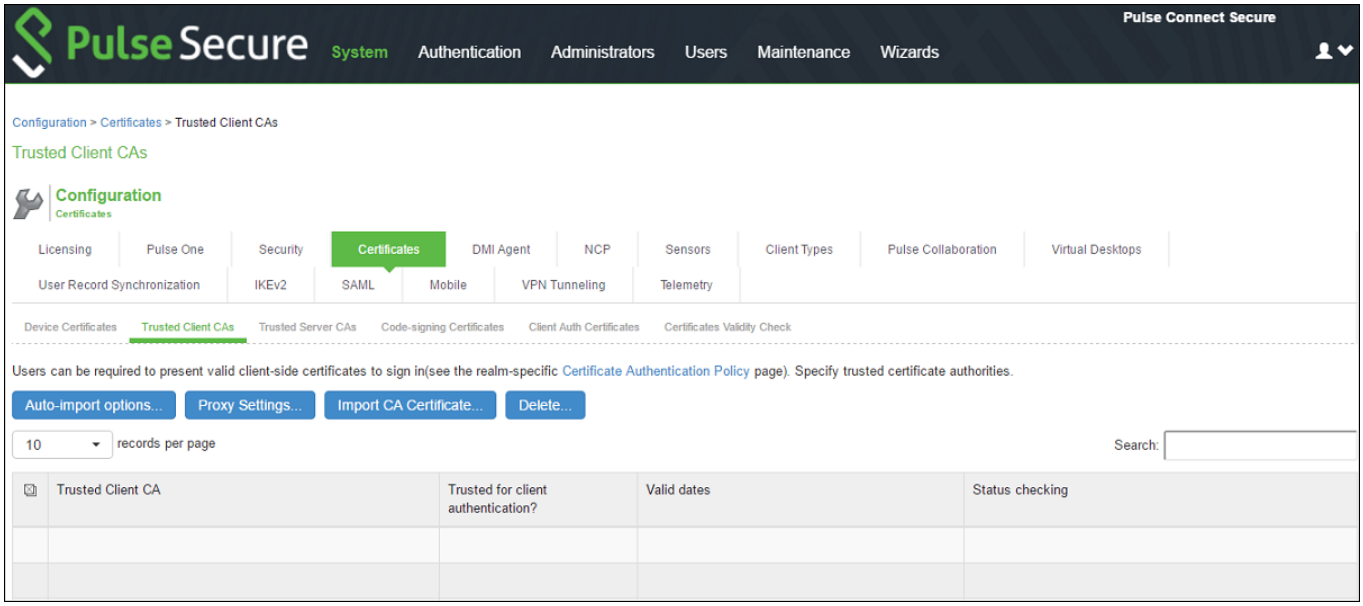
This feature enables users to login to the client using their certificates. The supported scenario is “certificate based login only" the Pulse Linux Client setup is now switched to this authentication method. In a typical enterprise environment, each user will be provided with certificate which can be 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:

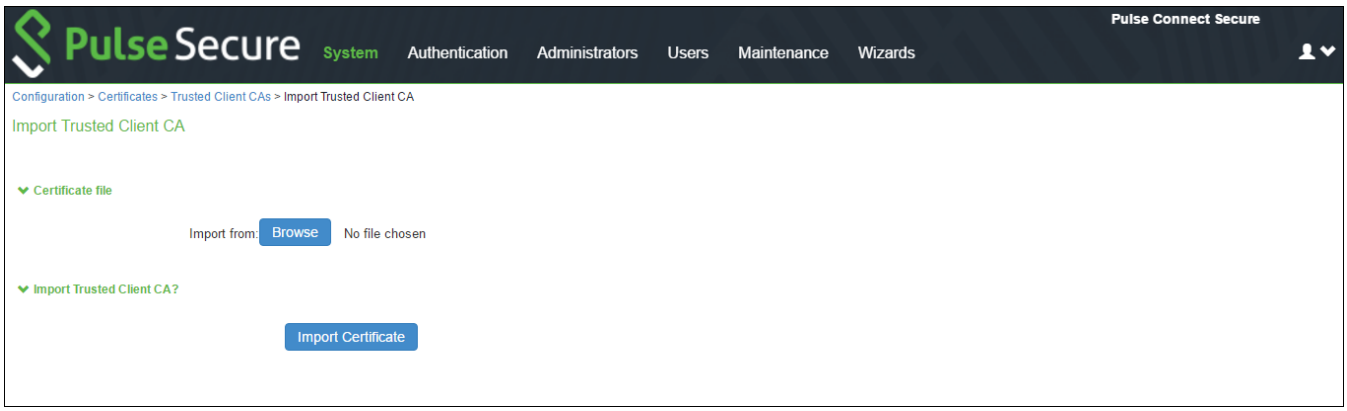
1. Select System > Configuration > Certificates > Trusted Client CAs.

Figure 17 Trusted Client CA Management



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

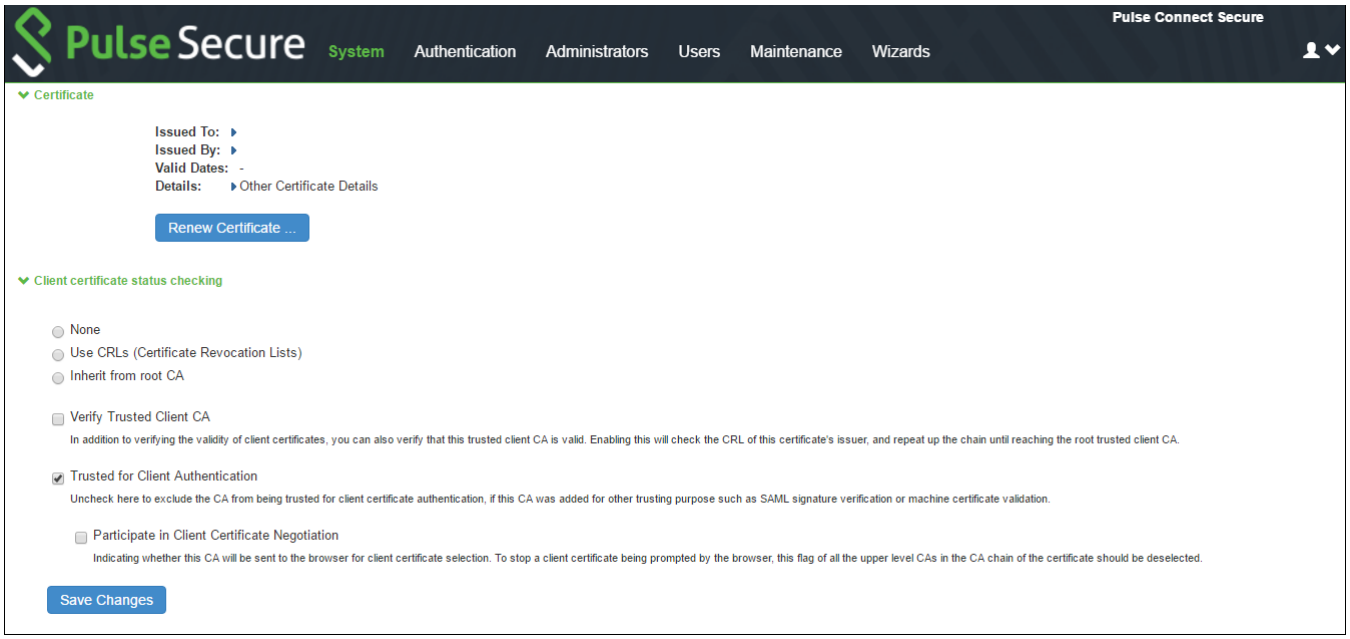
Figure 18 Import Trusted Client CA



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.

© 2018 by Pulse Secure, LLC. All rights reserved

Figure 19 Trusted Client CA Configuration

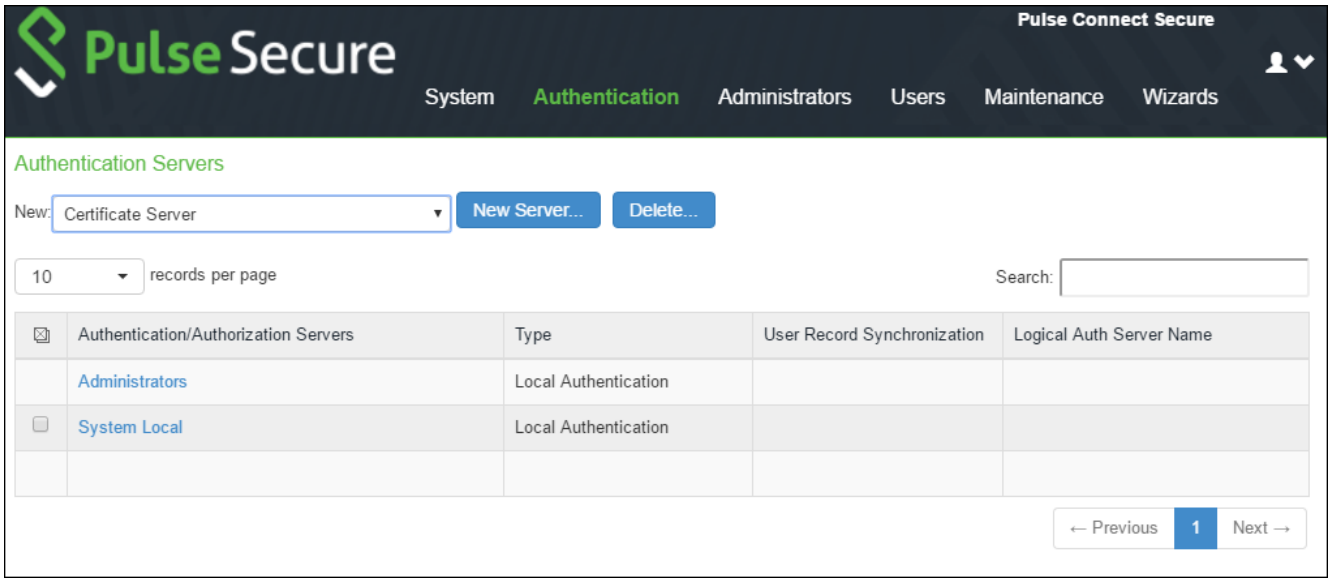


Configuring Authentication with the Certificate Server

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.

Figure 20 Authenticating Servers



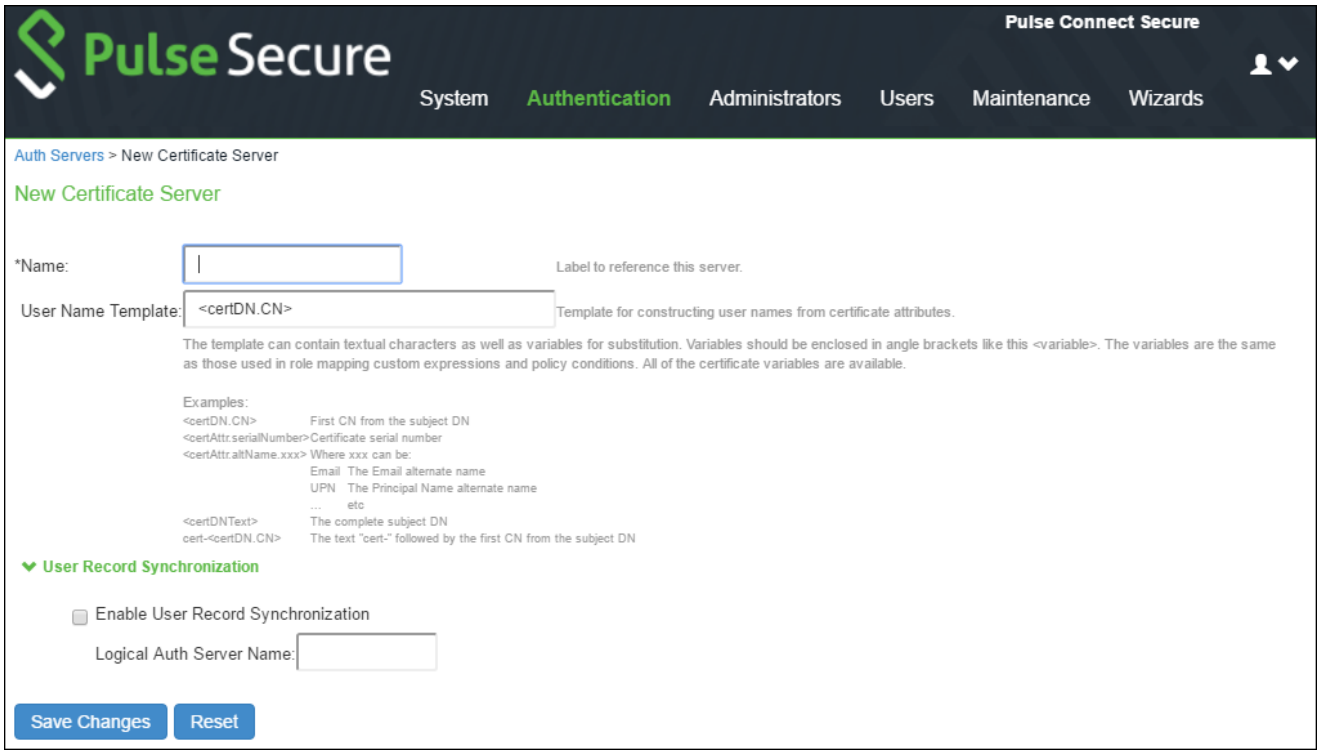
3. Complete the configuration as described in **Table 3**.

© 2018 by Pulse Secure, LLC. All rights reserved

Table 3: Certificate Server Settings

Settings	Guidelines
Name	Specify a name to identify the server within the system
User Name Template	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 <USER> and <USERNAME> session variables for use throughout the rest of the system configuration.
User Record Synchronization	This applies only to Connect Secure
Enable User Record Synchronization	Select this option to retain the bookmarks and individual preferences regardless of which system you log in to.
Logical Auth Server Name	Specify a logical authentication server name.

Figure 21 Configuring Certificate Server



4. Save the configuration.

© 2018 by Pulse Secure, LLC. All rights reserved

Client Certificate Installation

The installation of the certificates can be facilitated through a script. Client certificates can be installed using util script "**PulseClient.sh**". An example on how to install the certificate are shown below:

- To install the certificate from pfx file :

  /usr/local/pulse/PulseClient.sh install_certificates -inpfx /mnt/hgfs/shared_dir/10.30.113.196.pfx

- Using separate private and public certificate:

  /usr/local/pulse/PulseClient.sh install_certificates -inpriv /mnt/hgfs/shared_dir/certs/flowerCert.key -inpub /mnt/hgfs/shared_dir/certs/flowerCert.pem

- To list the certificate from Certificate Store:

  /usr/local/pulse/PulseClient.sh list_installed_certificates

- To delete the Certificate from certificate store

  /usr/local/pulse/PulseClient.sh delete_certificates -certName <certificate name>

Certificate Types Supported

Public Certificates

Extensions	Certificate Formats
der, cer	DER
pem, crt, key, pub	PEM

Private Keys

Extensions	Certificate Formats
der, cer	DER
pem, crt, key	PEM

PFX file (Contains both Private Key and Public Keys)

Extensions	Certificate Formats
Pfx, p12	PFX

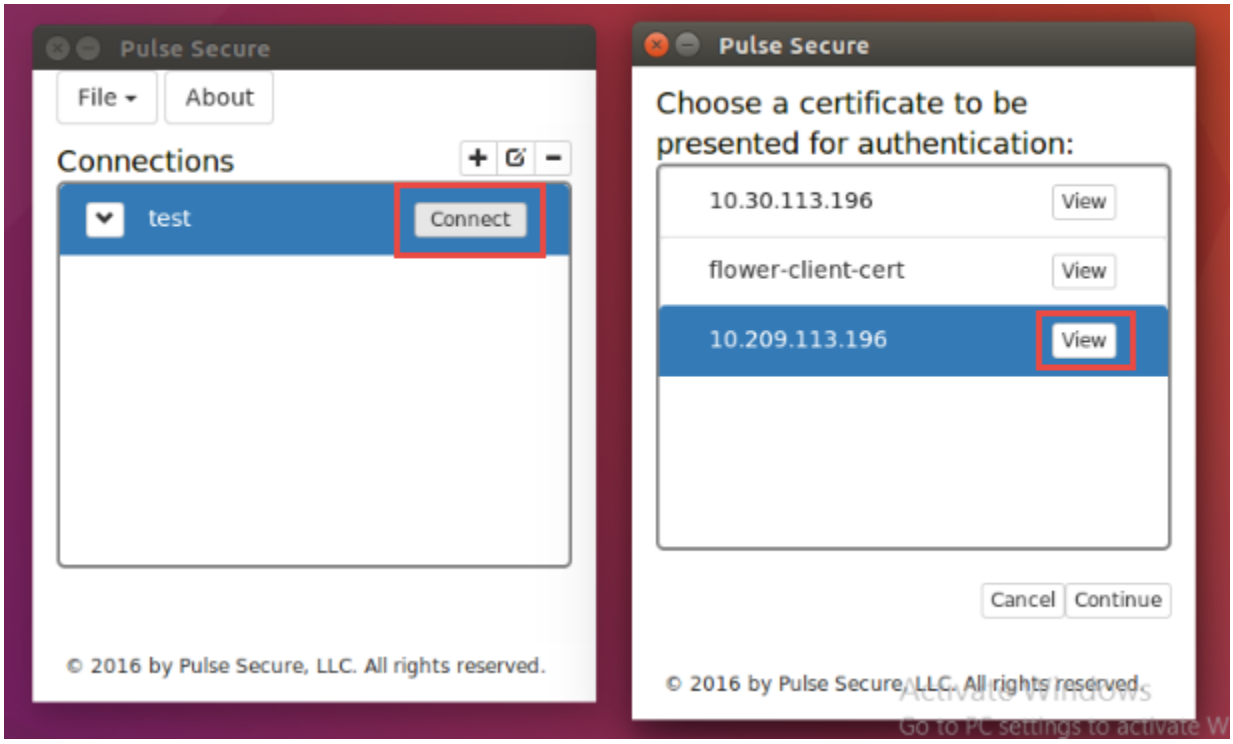
Note: Other formats like **p7b, p7c, pk8** are not currently supported. All of the above formats will be converted from PEM format to store it in GNOME-keyring.

© 2018 by Pulse Secure, LLC. All rights reserved

Default Certificate Selection

If multiple certificates are available for a connection, the certificates will be listed and user can select the certificate for establishing the connection. Once user selects the certificate, that certificate will be stored as preferred certificate for the corresponding connection and this will be used for subsequent authentications to all the corresponding connection.

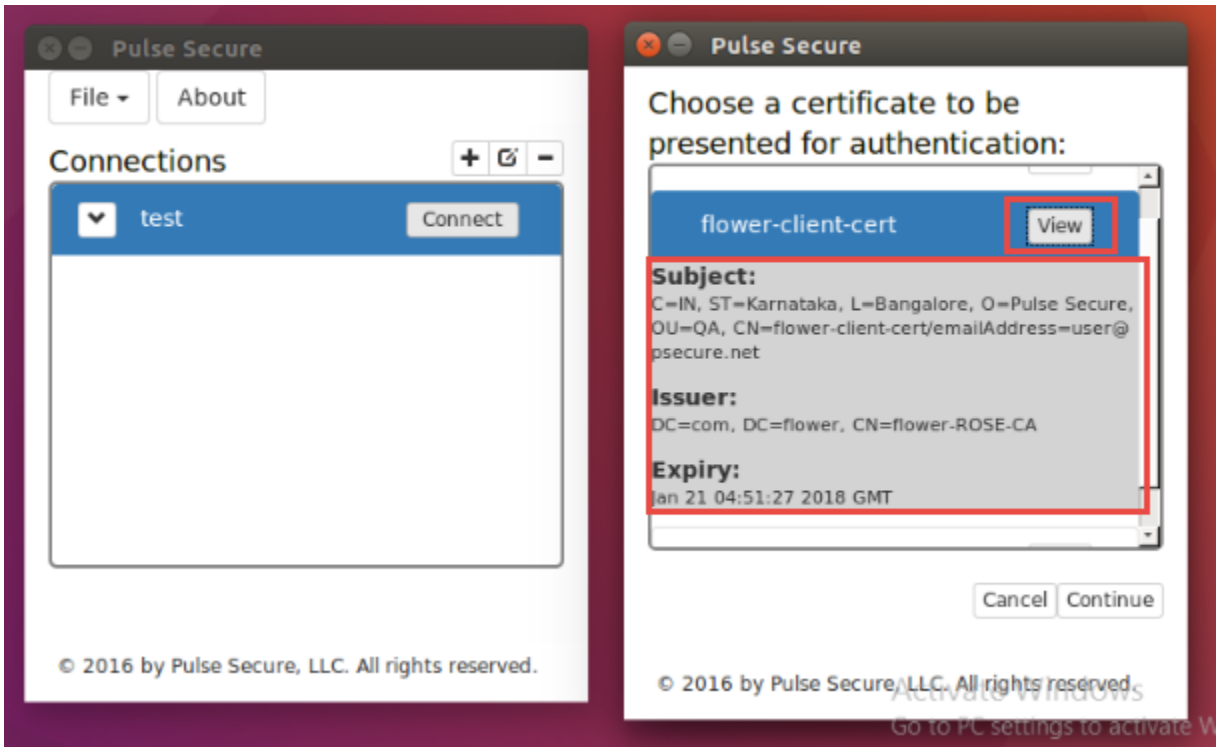
Figure 22 Cert_list



The above Figure 22 shows that when user clicks on connect the multiple client certificate will be displayed to continue authenticating for connection. On clicking the view, the certificate details can be viewed by the user (refer Figure 23).

© 2018 by Pulse Secure, LLC. All rights reserved

Figure 23 Certs_details_onclicking view



Note:

- The certificate authentication through UI will be supported only on the machines using libsoup 2.48 and above. Centos 6.4 and Ubuntu 14.04 doesn't meet the above requirements.
- The user cannot save the certificate selection. Save credentials will not be supported.
- Client certificate authentication through smart cards is not supported.

© 2018 by Pulse Secure, LLC. All rights reserved

Revision History

The following table lists the revision history for this document.

Revision	Date	Description
1.1	April, 2018	9.0R1
1.0	February, 2018	9.0R1 Beta
1.0	December, 2017	5.3R4 Update
4.0	October, 2017	5.3R3 Update

© 2018 by Pulse Secure, LLC. All rights reserved