SurePassID FreeRADIUS Module Guide

SurePassID Authentication Server 2025

© 2013-2025 SurePassID, Corp. All rights reserved. Protected by patents pending. SurePassID, the SurePassID logo and design, and Secure SSO are registered trademarks or trademarks of SurePassID, Corp. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

SurePassID, Corp.

360 Central Avenue

First Central Tower

Suite 800

St. Petersburg, FL 33701

USA

+1 (888) 200-8144

www.surepassid.com

Table of Contents

Table of Figures **4**

Introduction **5**

What is the SurePassID Module? **6**

Security 6

System Logging 6

Installing the SurePassID module (RLM_SUREPASSID) **7**

Red Hat Enterprise Linux (RHEL 7) 7

Configuring RLM_SUREPASSID module **10**

Configure RLM_SUREPASSID Settings 11

Verifying Installation **15**

Example 1 – Login Using Single-Factor 15

Example 2 – Login Using Code from Hard or Soft Token 16

Example 3 – Login Using SMS, Email or Voice Code 17

Example 4 – Login Using Push Authentication 19

Table of Figures

Figure 1: SurePassID Account Settings 15
Figure 2: Single-Factor VPN Login 16
Figure 3: Two-Factor Authentication with Token 17
Figure 4: Two-Factor Authentication with SMS Text Code 17
Figure 5: Two-Factor Authentication with Voice Call 18
Figure 6: Two-Factor Authentication with Email 18
Figure 7: Two-Factor Authentication with OTP 19
Figure 8: Request SMS Question 19
Figure 9: Two-Factor Authentication with SMS Question 20
Figure 10: Request Voice Call Question 20
Figure 11: Two-Factor Authentication with Voice Call Question 21
Figure 12: Request Push App Question 21
Figure 13: Two-Factor Authentication with Push App Question 22
Figure 14: Request Push OTP 22
Figure 15: Two-Factor Authentication with Push OTP 23

Introduction

This guide explains how to install and configure the SurePassID FreeRADIUS Module to meet your organization's security needs. The purpose of this guide is to provide a reference for system administrators.

This guide provides information on the following topics:

- **What is SurePassID FreeRADIUS Module?**
  - A brief introduction to the SurePassID FreeRADIUS Module and how it can help you get the most out of the SurePassID authentication system.
- **Installing and Configuring SurePassID FreeRADIUS Module**
  - Detailed explanations for installing, configuring and maintaining the SurePassID FreeRADIUS module.

Other SurePassID Guides

The SurePassID FreeRADIUS Module Guide has the following companion guides that provide additional detail on specific topics for SurePassID:

- Developer API Guide
- Fido U2F Mobile API Guide
- System Administration Guide
- Local Agent Guide
  - High performance Radius Server
  - Windows Event Log Synchronization
  - Active Directory Synchronization
- Desktop Authenticator Guide
- Google Authenticator Guide
- SurePassID Mobile Authenticator Guide
- Mobile API Connector
- Windows Credential Provider Guide
- Self-Service Portal

What is the SurePassID Module?

The SurePassID FreeRADIUS Module adds SurePassID authentication to your existing FreeRADIUS radius server installation using the standard modules that FreeRADIUS supports that is available in existing FreeRADIUS distribution. The SurePassID FreeRADIUS Module is a system service that allows SurePassID to authenticate users from any Radius-compliant system such as Microsoft Universal Access Gateway, VPN devices (Cisco, Sonic Wall, etc.), Wi-Fi Access points, etc.

Security

SurePassID FreeRADIUS Module uses transport level security (SSL) at a minimum. Optionally SurePassID FreeRADIUS Module can be configured to use message level security for a higher-level security.

System Logging

SurePassID FreeRADIUS module maintain its own system log files and write critical information to the system log. In tandem these two different event logs help you troubleshoot and repair any issues that SurePassID FreeRADIUS module might encounter during daily operations.

Installing the SurePassID module (RLM_SUREPASSID)

Red Hat Enterprise Linux (RHEL 7)

Prerequisites

- FreeRADIUS 3.0.7 – FreeRADIUS Getting Started Guide can be found at https://wiki.freeradius.org/guide/Getting%20Started.
- Knowledge and understanding of FreeRADIUS configuration and operational knowledge and expertise.
- RHEL 7.x - Consult the Red Hat Customer Portal for the latest information related to the most FreeRADIUS package.
- SurePassID MFA Server. Release 2019.x
- Perl 5 – You can install Perl 5 with the following command

```
yum install perl
```

- Perl 5 packages – After Perl 5 is installed install Perl 5 packages with the following commands:

```
cpan LWP
cpan JSON
cpan perl-LWP-Protocol-https
```

Distribution Media

The distribution media contains as the files required install the SurePassID FreeRADIUS modules. The distribution media is a tar archive. Contact SurePassID to get the get the link to download the tar archive.

Installation Steps

1. Create the SurePassID FreeRADIUS module installation folder **/home/surepassid**.
2. Download the SurePassID FreeRADIUS module tar archive.

3. Extract the contents of the archive using the following command:

```
tar -xzvf rlm_surepassid_current.tar.gz -C /home/surepassid
```

4. Copy RLM_SUREPASSID configuration file.

```
cp /home/surepassid/dist/etc/surepassid/rlm/radius.conf
etc/surepassid/rlm/radius.conf
```

5. Copy libperl.so to /usr/local/bin.

```
cp /home/surepassid/dist/bin/libperl.so /usr/local/bin/libperl.so
```

6.

1. Edit /etc/raddb/mods-available/perl file and edit the filename= parameter.

```
perl {
        #
        #  The Perl script to execute on authorize, authenticate,
        #  accounting, xlat, etc.  This is very similar to using
        #  'rlm_exec' module, but it is persistent, and therefore
        #  faster.
        #
        filename = /home/surepassid/rlm_surepassid/rlm_surepassid.pl

       …
       …
}
```

2. Edit etc/raddb/sites-available/default file and make the following two changes:
    - insert **perl** under the **authorize** section
    - insert **Auth-Type PERL { perl }** under the **authenticate** section

3. Edit etc/raddb/users file and insert **DEFAULT Auth-Type := perl** the authorize section.

```
            ...
            …

  DEFAULT Auth-Type := perl
            ...
            …
}
```

Configuring RLM_SUREPASSID module

The **RLM_SUREPASSID** allows you to add two-factor authentication (two-step authentication) to any system that supports Radius.

The server supports the following Radius features:

- **Challenge Response** – The user can be challenged for many different credentials. Most of the time, the challenge will be to provide a One-Time Password after successfully entering in a valid username and password. Some Radius devices (such as VPNs) only support single-factor authentication. Two-factor authentication can still be used by appending the One-Time Password to the user's password.
- **Proxy Server Chaining** – In Radius authentication, there can often be multiple Radius servers as part of the authentication process.

RLM_SUREPASSID also supports the following directories for single-factor (username and password) authentication:

- **SurePassID Directory** – For use with other cloud systems or external users that are not part of the existing enterprise Active Directory forest.
- **LDAP Directory** – For companies that use an LDAP directory such as Unix and Linux systems.

The **RLM_SUREPASSID** module supports the sending One-Time Codes to the user as well as pushing authentication requests to the SurePassID Mobile Authenticator.

Send OTP Options:

- **Send SMS OTP**– A SMS text message containing the OTP is sent to the user's phone.
- **Send OTP by Voice Call -** A call is made to the user's phone speaking the OTP. This is an invaluable option for users that do not have SMS capabilities on their phone or users sitting at their desk or for the visually impaired.
- **Email Code** – An email containing the OTP is sent to the user's email account.

Push Authentication Options:

- **Push SMS Question** – A question is sent to the user's mobile device asking the user to confirm a request to allow access to the system. If the user responds positively, the user is allowed to login with just username and password.

- **Push Question -** A question is sent to the user's mobile device asking the user to confirm a request to allow access to the system. If the user responds positively, the user is allowed to login with just username and password.
- **Push OTP -** An OTP is sent to the user's mobile device. The OTP can be used as if it were an OTP from a soft token, or hard token (fob or card).
- **Push Voice Question -** A voice call is made to the user's phone asking the user to confirm access to the system. If the user responds positively, the user is allowed to login with just username and password.

HINT: All messages sent to the user can be tailored to your company's needs in the SurePass portal using the Customize SMS Messages and Customize Email Messages menus.

Configure RLM_SUREPASSID Settings

The RLM_SUREPASSID configuration settings are stored in the /etc/surepassid/rlm/radius.conf file. The format of this file is the same format as the settings file for the Windows Credential Provider and ADFS plug-ins. The default radius.conf is shown below:

```
AuthServerURL=https://sandbox.surepassid.com/AuthServer/REST/OATH/OATHServer.aspx
AuthServerToken=<your account token>
AuthServerKey=<your account key>
AllowSMS=0
AllowEmail=0
AllowCall=0
AllowPushApp=0
AllowPushSMS=0
AllowPushOtp=0
AllowPushAppU2F=0
AllowPushVoice=0
PushAppName=Remote Access
PushAuthnReason=Login
RelyingPartyUrl=
TraceOn=0
```

When FreeRADIUS (radiusd) initializes it loads the RLM_SUREPASSID module. The RLM_SUREPASSID reads the radius.conf file and caches the settings for processing future RADIUS requests.

Note: When you make changes to the radius.conf file the settings will not take effect until the FreeRADIUS server is restarted.

The format of the file is one option per line, each option is a combination if an option name and value are separated by an equal (=) sign. Option names are described below:

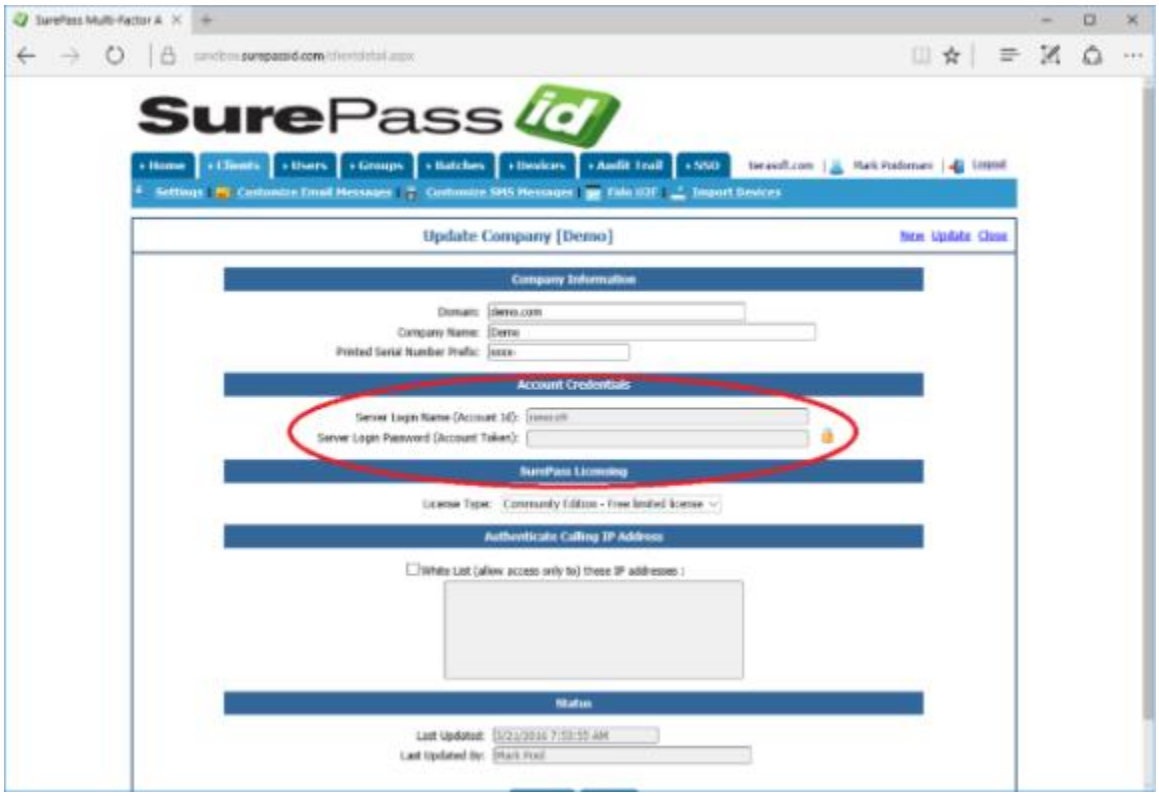
- **AllowSMS** - Allow the user to request an OTP be sent by SMS to their mobile device. 0=no 1=yes
- **AllowEmail** - Allow the user to request an OTP be sent to their email. 0=no 1=yes
- **AllowCall** - Allow the user to request an OTP be sent by voice call. 0=no 1=yes
- **AllowPushApp** - Allow the user to request that a push authentication be sent (pushed) to their mobile device to confirm their identity. Requires the user to have SurePassID Mobile Authenticator installed on their mobile device. 0=no 1=yes
- **AllowPushSMS** - Allow the user to request that an SMS push question can be sent to their mobile device to confirm their identity. 0=no 1=yes
- **AllowPushOtp** - Allow the user to request an OTP be sent to their mobile device. Requires the user to have SurePassID Mobile Authenticator installed on their mobile device. 0=no 1=yes
- **AllowPushAppU2F** - Allow the user to request that they be authenticated on their mobile device with a Fido U2F token. Requires the user to have SurePassID Mobile Authenticator installed on their mobile device. 0=no 1=yes
- **AllowPushVoice** - Allow users to request a voice call that will allow them to confirm their identity. 0=no 1=yes
- **PushAppName** - For push authentications, the name of the application requesting access. The PushAppName is displayed to the user when they receive the in a push notification. The default is Remote Access.
- **PushAuthnReason** – For push authentications, the reason why the application is requesting access. The default is Login.
- **RelyingPartyUrl** - The URL of the application that is requesting access in push notifications. Push notifications response will be sent to this URL.
- **TraceOn** - Turn on tracing for RLM_SUREPASSID. The trace file is stored in /etc/surepassid/rlm. 0=no 1=yes
- **AuthServerURL** – The SurePassID authentication endpoint URL. In most cases, you will not need to change this unless you are using a custom SurePassID installation. The values are:
    - **sandbox** - The SurePassID sandbox system.
    - **prod** – The SurePassID production cloud system.
    - **SurePassID MFA System** (Full Name)- On-premises or custom install of the SurePassID MFA server. The format of this parameter is usually: **https://<surepassid_server>/api/mfa/v1**

- **AuthServerToken** - Refers to the **API Key Identifier**, which forms one part of a custom API Key used to specify ADFS access rights to an MFA

server. For earlier versions of SurePassID, use **Server Login Name** for your SurePassID account.

- **AuthServerKey** - Refers to the **API Key**, which forms a second part of a custom API Key used to specify ADFS access rights to an MFA server. For earlier versions of SurePassID, use **Server Login Password** for your SurePassID account.

If you are using an older version of SurePassID MFA server, you will see the following info for your account as shown below.



SurePassID Legacy Account Settings

If you are using the latest version of SurePassID you will see the following info for your account:

SurePassID API Key Settings

All application requests to the MFA Server require an **Application Key**. To create a new **Application Key** just for **RADIUS Server** (recommended) click the **New Application Key** link and see the form below:

The **Key Name** is the friendly name to identify this key. It is not used for any security purposes. For instance, you could name it **ADFS Key**.

Select the **RADIUS** and **FreeRADIUS** the **Permissions Templates** drop-down list.

The **Key Identifier** parameter is the **AuthServerToken** and the **Key** parameter is the **AuthServerKey**

Copy **Key Identifier** and **Key** for later use when configuring **ADFS MFA Adapter**. One you click the **Add** button the **Key** will no longer be viewable. If you forget the **Key**, you can delete this **Application Key** and add a new one.

Press the **Add** button to save the Api Key.

```
radtest <sp_user_name> <sp_user_password><otp> localhost 0  testing123
```

Caution: Deleting an **Application Key** prevents applications using it from accessing the MFA Server. It cannot be restored; a new key must be generated and updated in the application.

Figure 1: SurePassID Account Settings

Verifying Installation

You can verify the installation operation by starting FreeRADIUS and using the FreeRADIUS **radtest** tool on the FreeRADIUS server.

Where:
<sp_user_name> - SurePassID username
<sp_user_password> - SurePassID password for user <sp_user_name>
<otp> - OTP for <sp_user_name>

Information on the syntax can be found here:

https://freeradius.org/radiusd/man/radtest.html

VPN End-User Login Overview & Examples

Example 1 – Login Using Single-Factor

Logging into your VPN with single-factor authentication is a fairly straight forward and legacy process that requires only the username and password. Typically, you follow these steps:

1. Start VPN client software
2. Enter username
3. Enter password

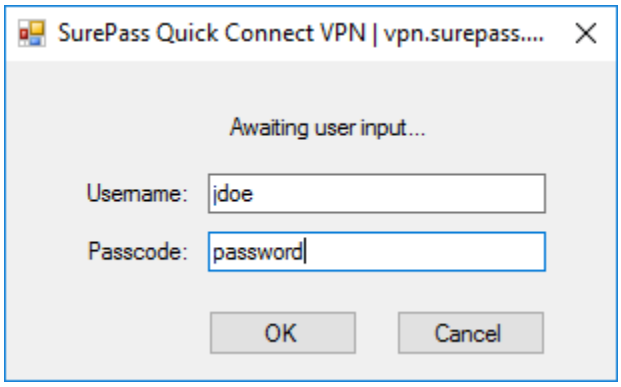


Figure 2: Single-Factor VPN Login

4. Press **OK** button to authenticate

When you use two-factor authentication, the process changes slightly because you will need to enter the second factor code in addition to the username and password.

Example 2 – Login Using Code from Hard or Soft Token

You will follow these steps if you have a device that displays a second factor code such as a hard token (OTP display smart card, key fob, etc.) or a soft token (SurePassID desktop token, mobile OTP apps such as SurePassID Mobile Authenticator, Google Authenticator, Authy, etc.)

1. Start VPN client software
2. Enter username
3. Enter password and concatenate the second factor code that is displayed from the device. In this example, the number displayed on the device is 254865 as show below:

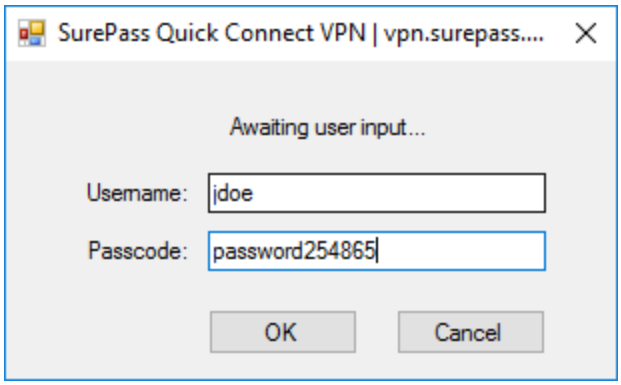


Figure 3: Two-Factor Authentication with Token

4. Press **OK** button to authenticate

Example 3 – Login Using SMS, Email or Voice Code

You will follow these steps if you want to have a second factor code sent to you via SMS Text, Voice Call, or Email:

1. Start VPN client software.
2. Enter username.
3. Enter a command in the password field. You can enter these values:

   - **#** (or **SENDSMS**) to have a code sent via text to your cell phone.

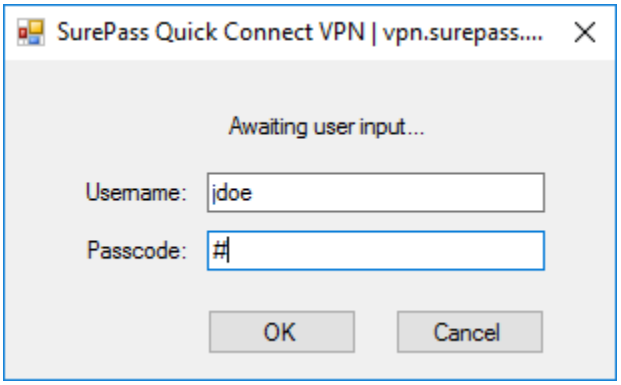


Figure 4: Two-Factor Authentication with SMS Text Code

   - **v** (or **SENDVOICE**) to have an automated voice call made to your cell phone that will tell you a code.

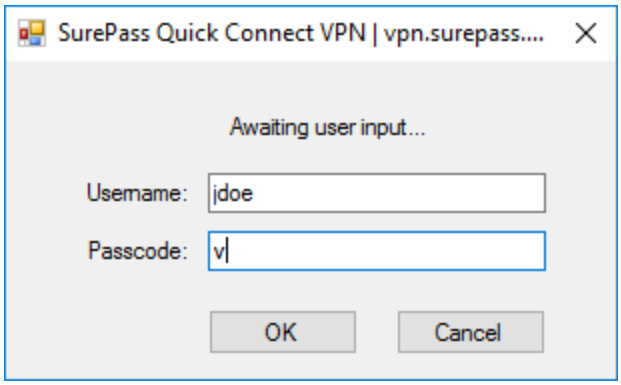


Figure 5: Two-Factor Authentication with Voice Call

- **m** (or **SENDEMAIL**) to have a code sent via email.

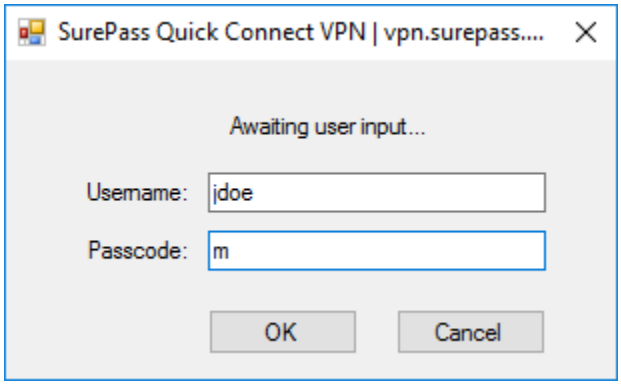


Figure 6: Two-Factor Authentication with Email

1. Press **OK** button to send code.
2. Wait for the OTP to be delivered to you.
3. Enter the password and concatenate (append) the OTP you have just received (no spaces after the password). Assuming the OTP received is 254865.
4. Press **OK** button to authenticate.

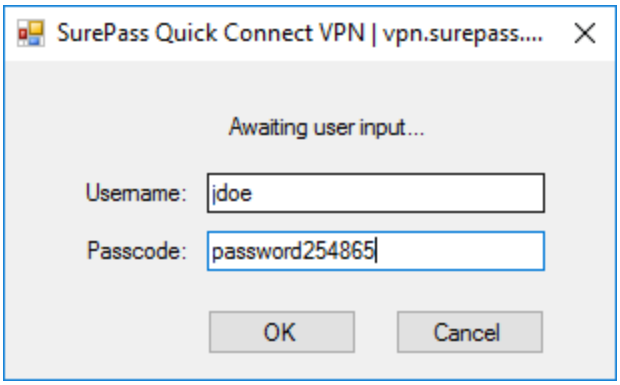


Figure 7: Two-Factor Authentication with OTP

Example 4 – Login Using Push Authentication

1. Start VPN client software.
2. Enter username.
3. Enter a command in the password field. You can enter these values:

You will follow one of these steps to secure yourself via a push notification.

- Enter **?** (or **PUSHSMS**) to have a question sent via text to your phone. You only need to reply **y** or **yes** (not case sensitive) to the text message to have the second factor authenticated. If you did not request to be authenticated, you can respond with any other answer (such as **n** or **no**) and access will be denied.

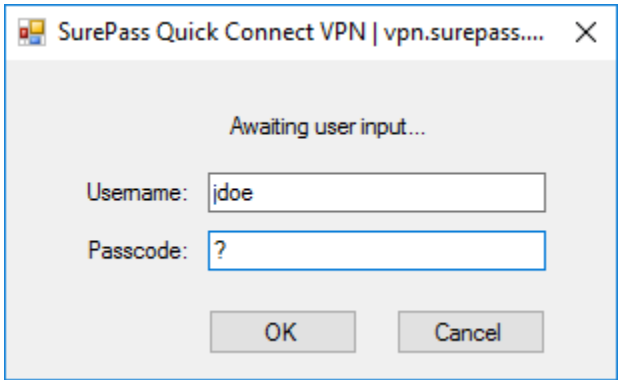


Figure 8: Request SMS Question

1. Press button **OK** to send your mobile phone an SMS question.
2. Wait for the question to be delivered via SMS to your mobile device.

3. Reply **Yes** (or **Y**) to the SMS question to allow approve access.
4. Wait for SMS confirmation that you have been authenticated.
5. Enter your account password.
6. Press **OK** button to authenticate

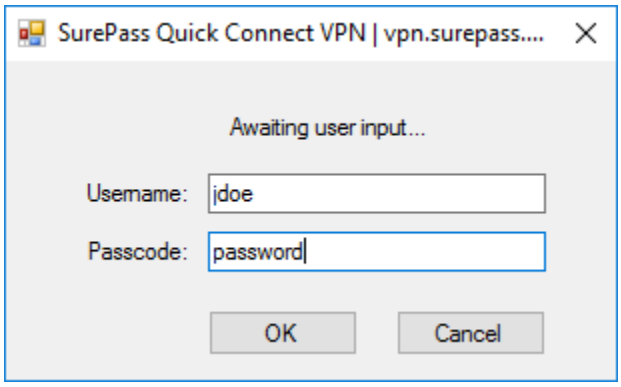


Figure 9: Two-Factor Authentication with SMS Question

- Enter **c** (or **PUSHVOICE**) to have a call sent to your phone. After you receive the call, you only need to reply by press **#** or **Y** on your phones keypad to verify your access request. If you did not request to be authenticated you can respond with any other answer (such as **n** or **no**) or hang up and access will be denied.

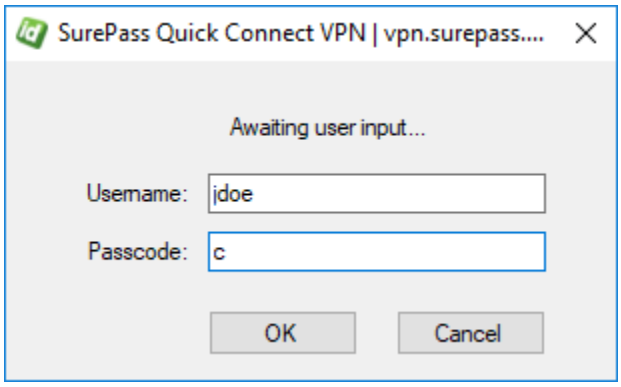


Figure 10: Request Voice Call Question

1. Press button **OK** to send your mobile a question.
2. Wait for the question to be delivered via SMS to your mobile device.
3. Reply **#** (or **Y**) to the voice call question to approve access.
4. Wait for voice response that you have been authenticated.
5. Enter your account password.
7. Press **OK** button to authenticate

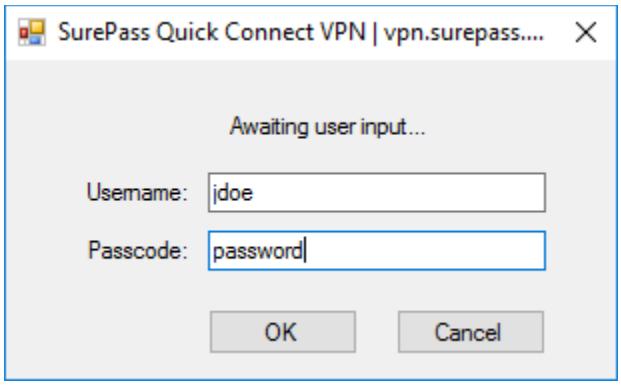


Figure 11: Two-Factor Authentication with Voice Call Question

- Enter **??** (or **PUSHAPPQUESTION**) to have a question sent to the SurePassID Mobile Authenticator app installed on your phone for approval. You only need to click the **Authenticate** alert on your phone to authenticate yourself.

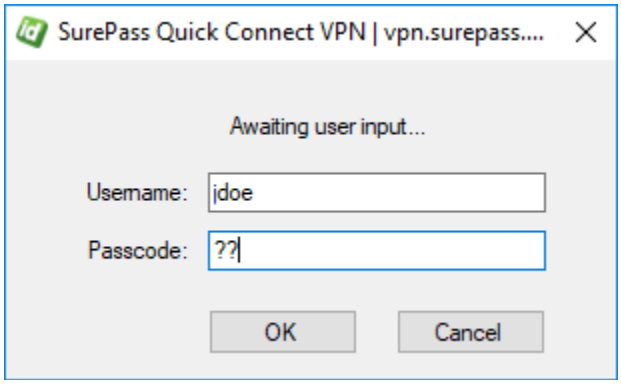


Figure 12: Request Push App Question

1. Press button **OK** to send you a question.
2. Wait for the authentication request to be delivered your mobile device as a mobile alert.
3. Press the **Authenticate** option in the mobile alert.
4. Wait for confirmation from the server that you have been authenticated.
5. Enter password.
6. Press **OK** button to authenticate.

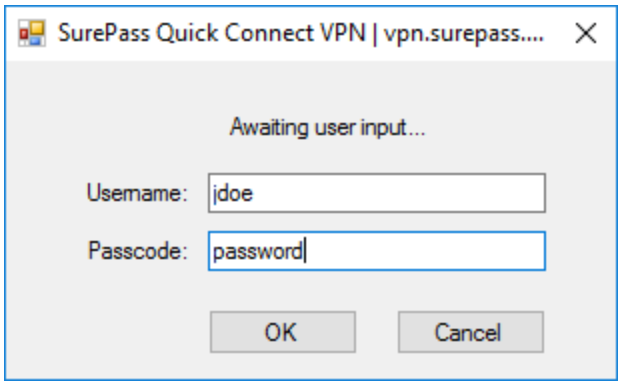


Figure 13: Two-Factor Authentication with Push App Question

- Enter **o** (or **PUSHOTP**) to have an OTP sent to the SurePassID Mobile Authenticator app on your phone. Once you receive the OTP you will enter it with your password.

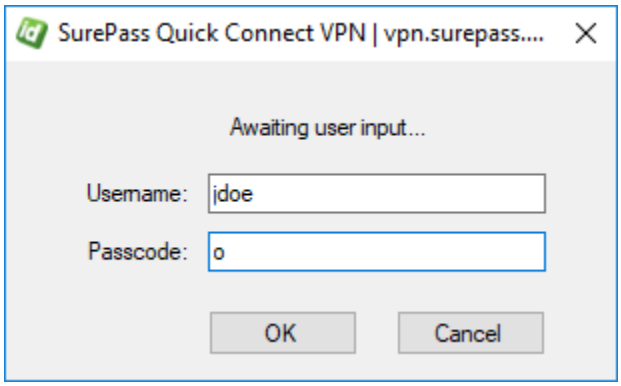


Figure 14: Request Push OTP

1. Press button **OK** to send you a question.
2. Wait for the code to be delivered to you.
3. Enter your password and concatenate (append) the OTP code you have just received (no spaces after the password).
4. Press **OK** button to authenticate.

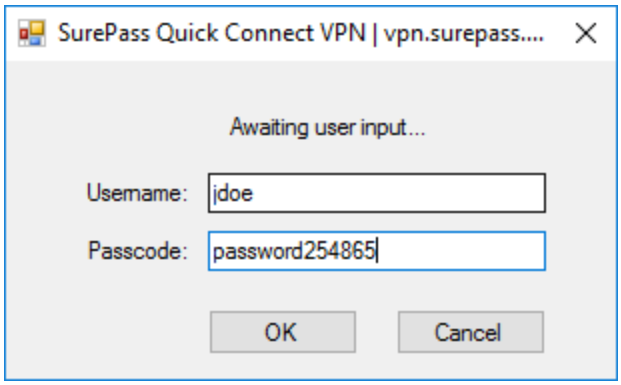


Figure 15: Two-Factor Authentication with Push OTP