

TECHNOLOGY



Automation Testing

SSL Authentication



A Day in the Life of an Automation Test Engineer

Thomas has successfully authenticated his API project on Rest Assured. Now, however, he wants to double-check the authentication process. So, he wants to use SSL authentication, which will also increase legitimacy of the website.

To achieve the above goal, he must learn about the SSL authentication and its types and the process to deliver a two-way authentication.



Learning Objectives

By the end of this lesson, you will be able to:

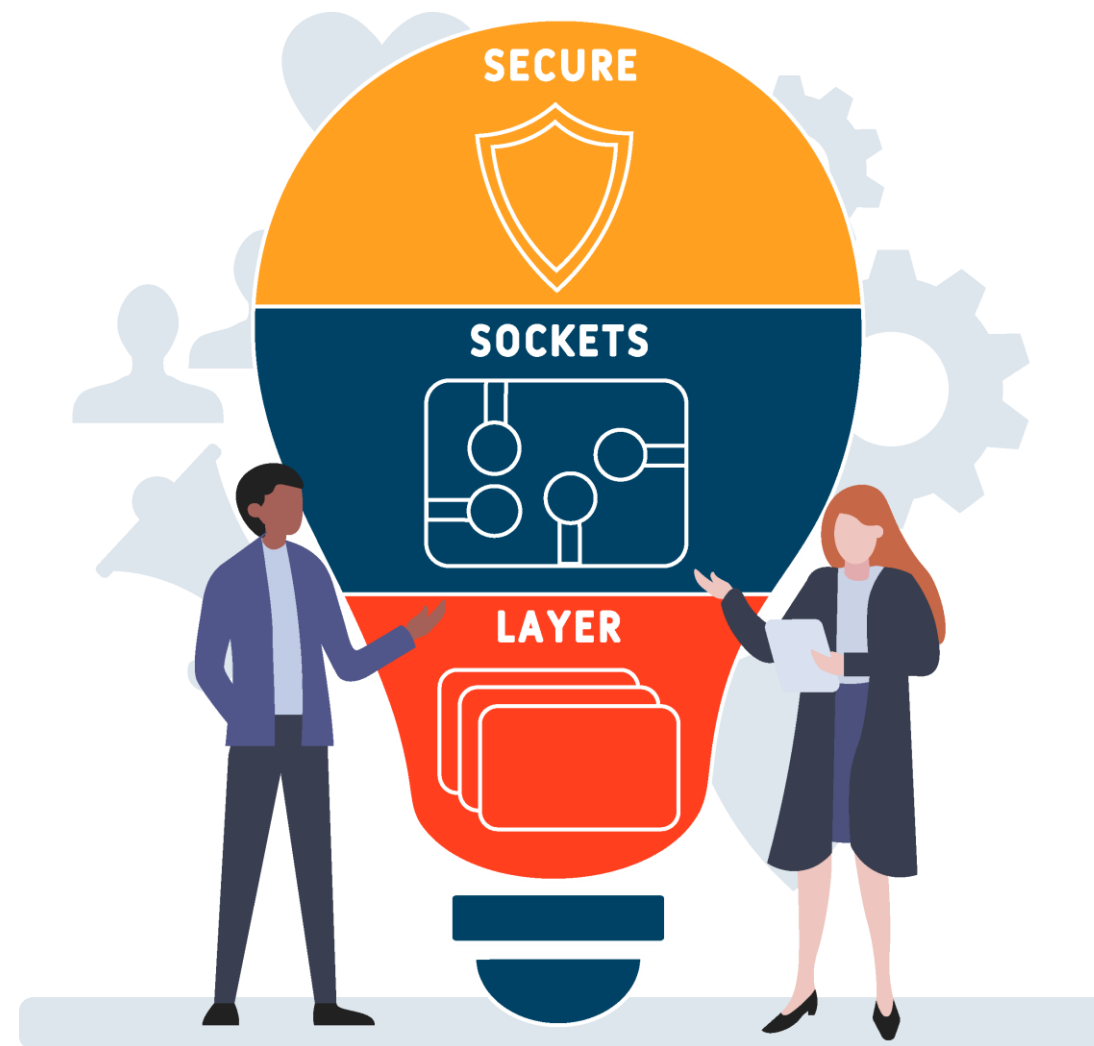
- 👁 Understand the working of SSL authentication
- 👁 Explain the types of SSL authentication
- 👁 Explain the importance of SSL authentication
- 👁 Perform two-way authentication process using SSL



What Is SSL Authentication?

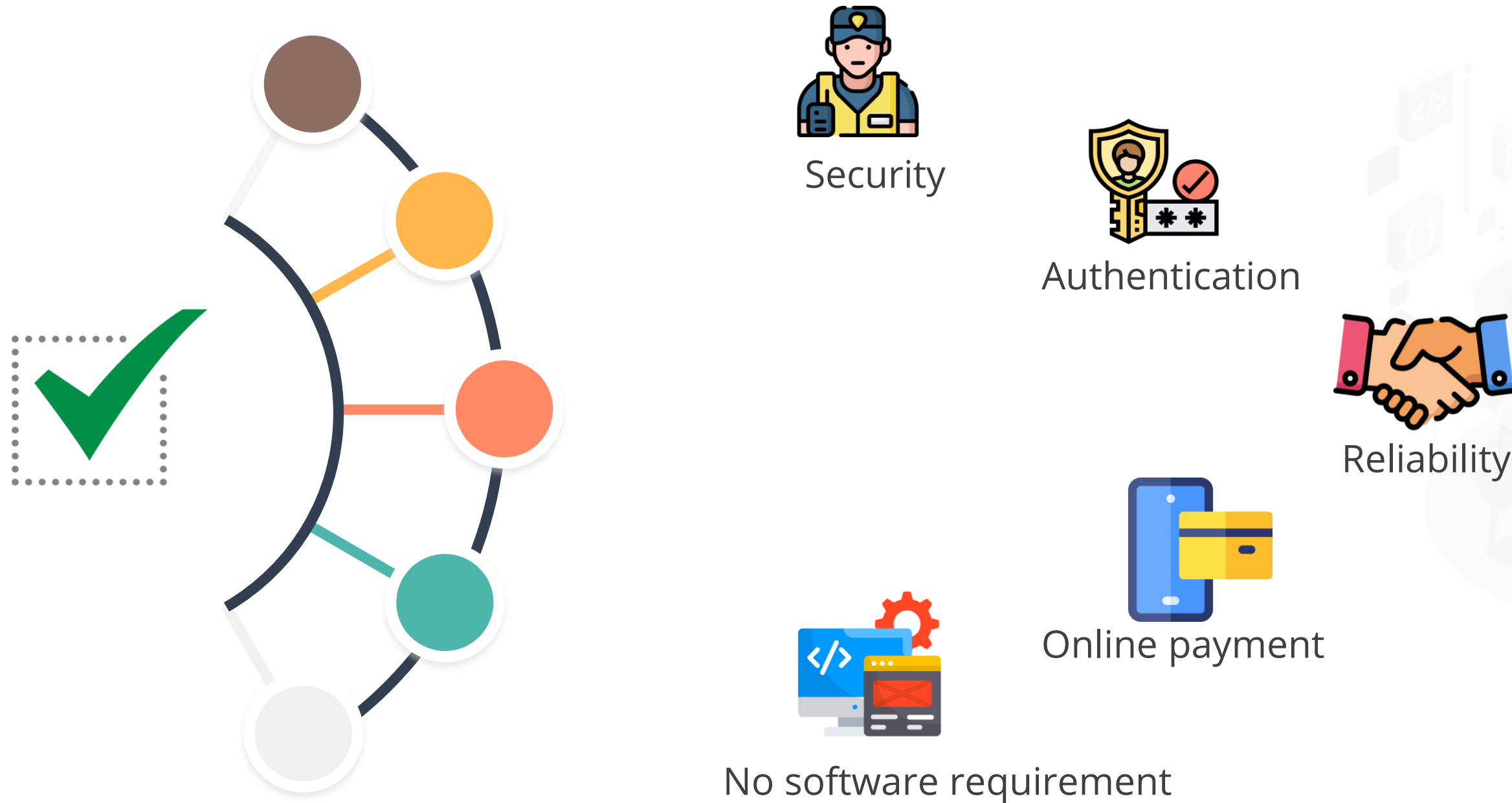
What Is SSL Authentication?

SSL stands for Secure Socket Layer. It helps to create a secure connection between client and server.



Why SSL Authentication?

The various reasons to use SSL authentication are:



Why SSL Authentication?

The various reasons to use SSL authentication are following:



Prevention of phishing



Better search engine ranking



Types of SSL Authentication for REST

There are two types of SSL authentication:

One-way SSL authentication

Two-way SSL authentication

It is also known as server certificate authentication.



Single way check

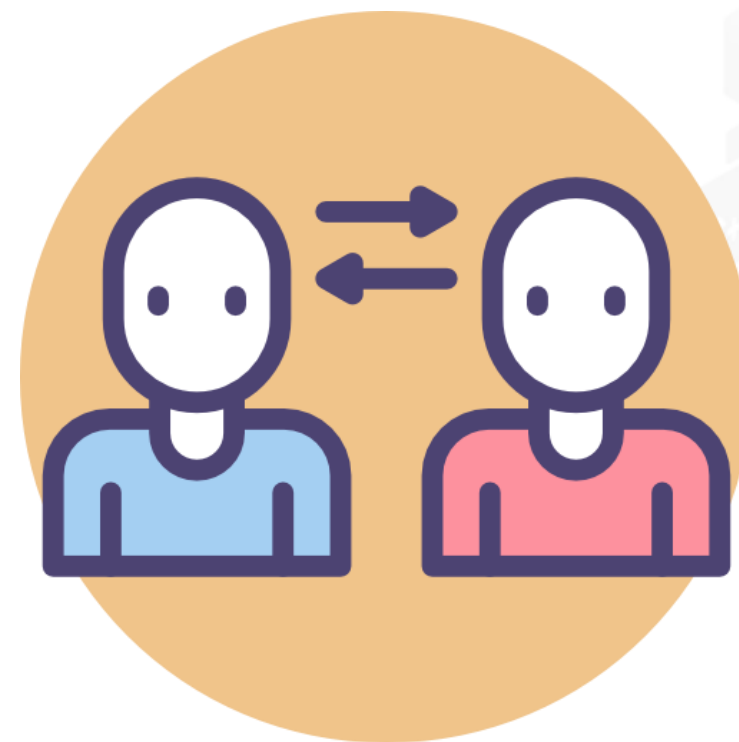
Types of SSL Authentication for REST

There are two types of SSL authentication:

One-way SSL authentication

Two-way SSL authentication

It is also known as client authentication.

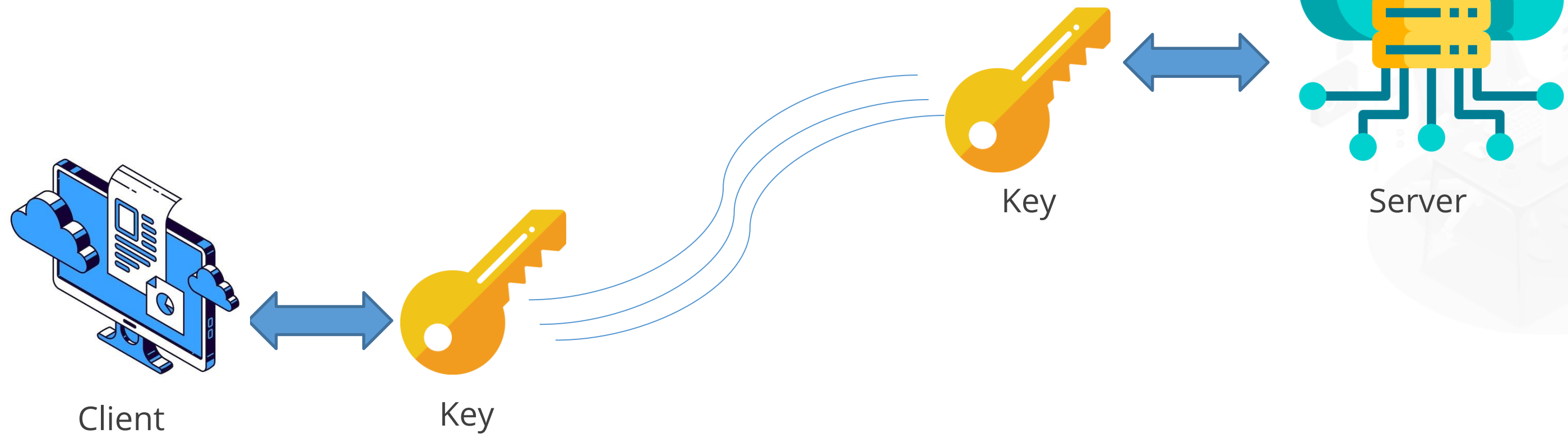


Mutual check

Four Phases of SSL Handshake

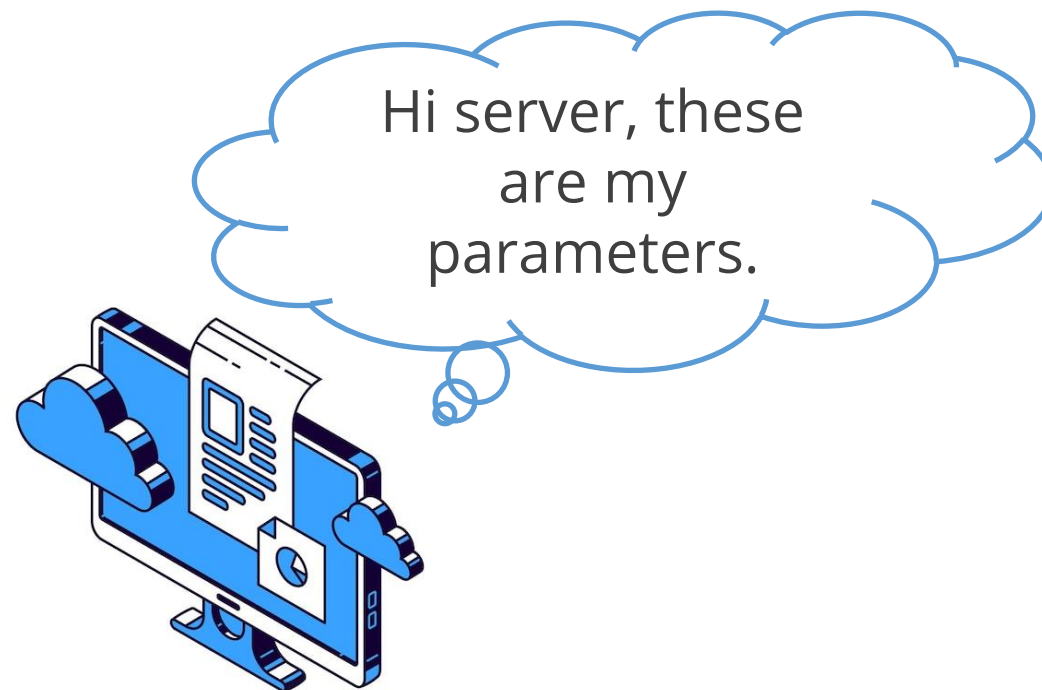
Four Phases of SSL Handshake

Handshake is a process by which a client and a server establish a key, which is used for communication.



Four Phases of SSL Handshake

Phase 1: The first phase involves the establishment of security capabilities.



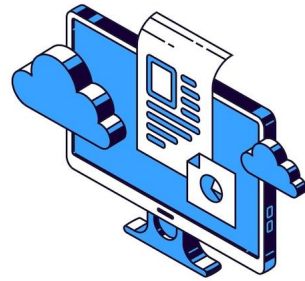
Client_hello



Server_hello

Four Phases of SSL Handshake

Phase 1: The first phase involves the establishment of security capabilities.



Client_hello parameters

- Version
- Client random
- Session ID
- Cipher suite
- Compression method

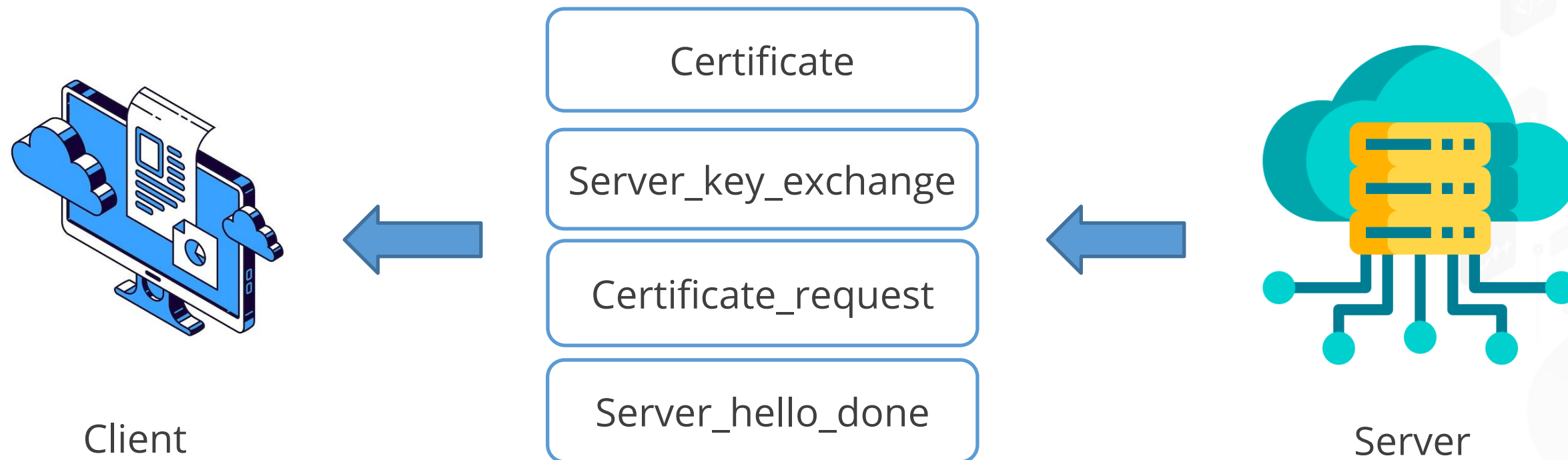


Server_hello parameters

- Version
- Server random
- Session ID
- Selected cipher suite
- Selected compression method

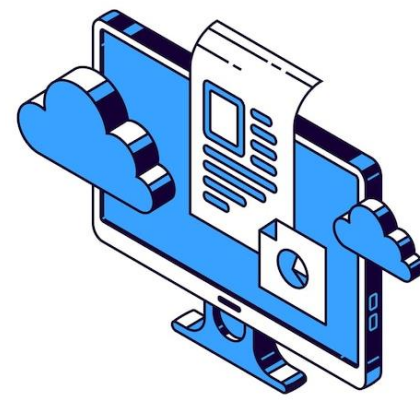
Four Phases of SSL Handshake

Phase 2: The third phase involves the authentication of the server and the exchange of keys.



Four Phases of SSL Handshake

Phase 3: The third phase involves the authentication of the client and the exchange of keys.



Client



Certificate

Client_key_exchange

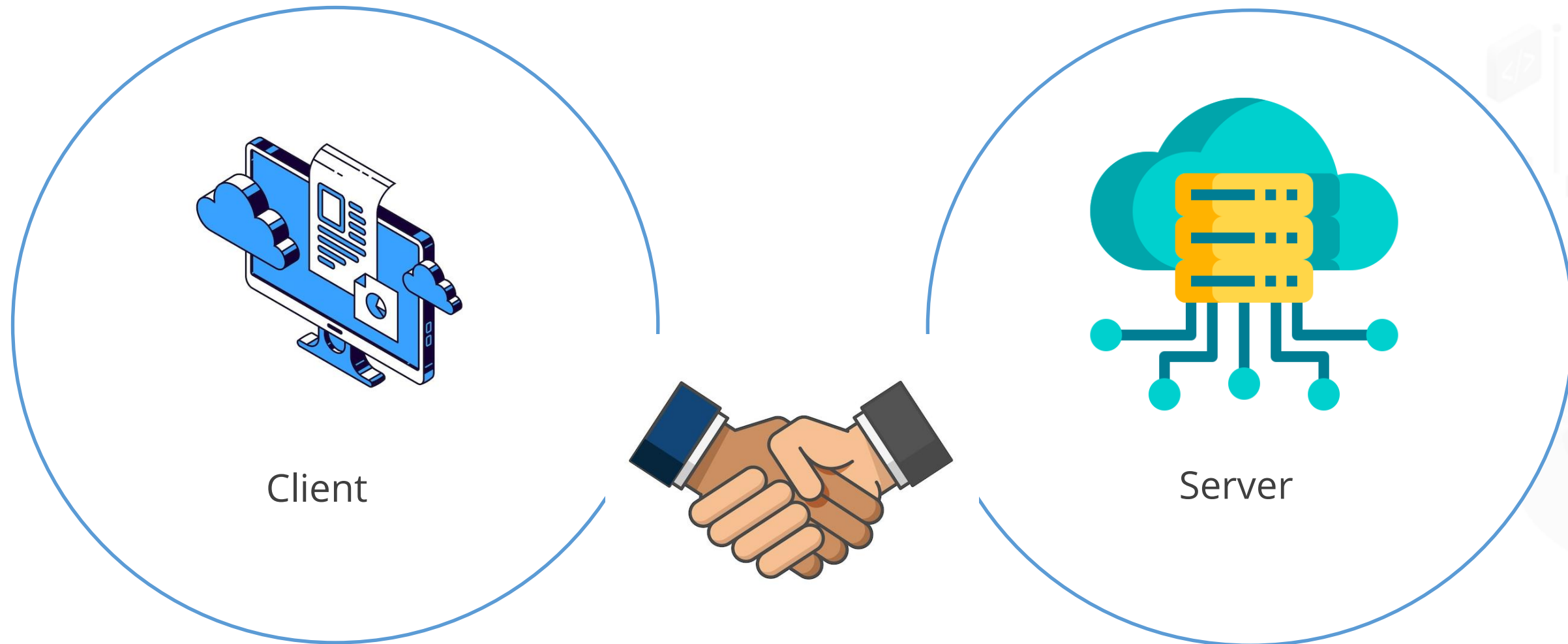
Certificate_verify



Server

Four Phases of SSL Handshake

Phase 4: The fourth phase involves the finalizing of the handshake protocol.

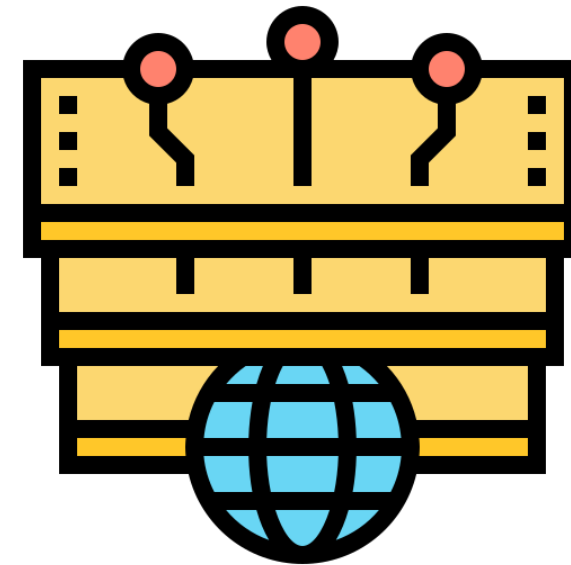


Post-Handshake Protocol

After the completion of four phases, the session starts, and data can be exchanged between the client and the server.



One session



Multiple connections



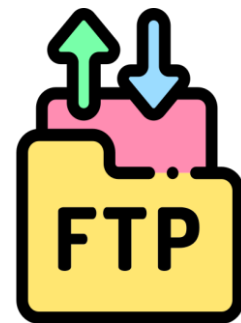
Pre-requisites for SSL Authentication in REST

Pre-requisites for SSL Authentication in REST

Pre-requirements for configuring SSL authentication in REST:



OpenSSL



FTP user



Postman



Client username

What Is OpenSSL?

OpenSSL is an open-source cryptography library tool. It is used for:



Steps to Install OpenSSL

Following are the steps to install OpenSSL in the computer:



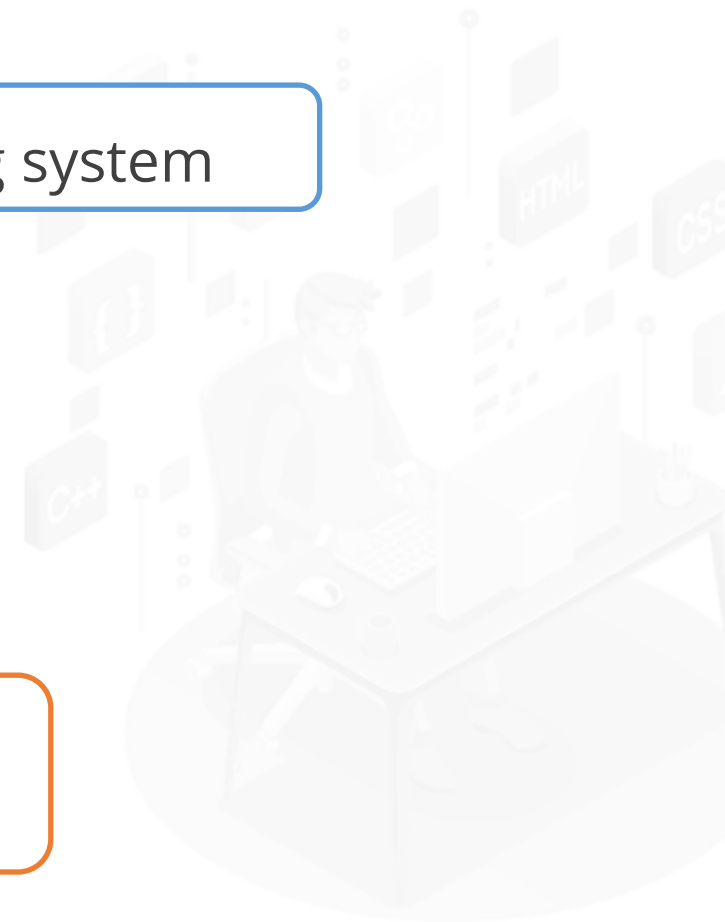
Step 1

Download the open SSL version according to your operating system



Step 2

Run the downloaded setup to begin the installation process



Steps to Install OpenSSL

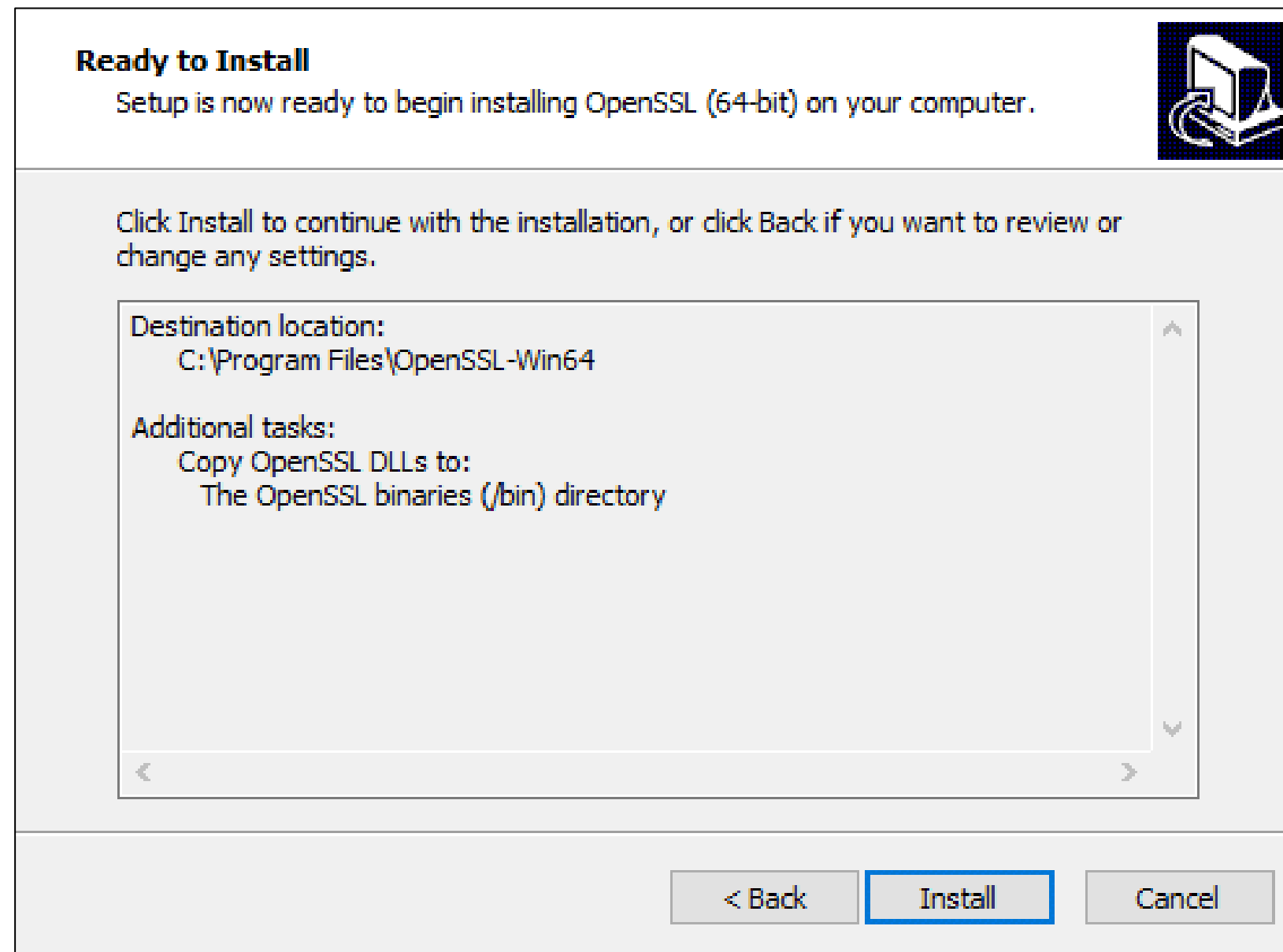
Select the recommended version according to the systems specifications to download:

Download Win32/Win64 OpenSSL		
Download Win32/Win64 OpenSSL today using the links below!		
File	Type	Description
Win64 OpenSSL v3.0.5 Light EXE MSI	5MB Installer	Installs the most commonly used essentials of Win64 OpenSSL v3.0.5 (Recommended for users by the creators of OpenSSL). Only installs on 64-bit versions of Windows. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.0.5 EXE MSI	140MB Installer	Installs Win64 OpenSSL v3.0.5 (Recommended for software developers by the creators of OpenSSL). Only installs on 64-bit versions of Windows. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win32 OpenSSL v3.0.5 Light EXE MSI	4MB Installer	Installs the most commonly used essentials of Win32 OpenSSL v3.0.5 (Only install this if you need 32-bit OpenSSL for Windows. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win32 OpenSSL v3.0.5 EXE MSI	116MB Installer	Installs Win32 OpenSSL v3.0.5 (Only install this if you need 32-bit OpenSSL for Windows. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.0.5 Light for ARM (EXPERIMENTAL) EXE MSI	5MB Installer	Installs the most commonly used essentials of Win64 OpenSSL v3.0.5 for ARM64 devices (Only install this VERY EXPERIMENTAL build if you want to try 64-bit OpenSSL for Windows on ARM processors. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.0.5 for ARM (EXPERIMENTAL) EXE MSI	113MB Installer	Installs Win64 OpenSSL v3.0.5 for ARM64 devices (Only install this VERY EXPERIMENTAL build if you want to try 64-bit OpenSSL for Windows on ARM processors. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.

Screenshot courtesy: <https://slproweb.com/products/Win32OpenSSL.html>

Steps to Install OpenSSL

Click on install to run the setup:



Screenshot courtesy: <https://thesecmaster.com/procedure-to-install-openssl-on-the-windows-platform/>

Steps to Install OpenSSL

Open the command prompt using ' Windows + r ' and type cmd. Set the OPENSSL_CONF and Path environment variables using the commands:



```
set OPENSSL_CONF=C:\OpenSSL-Win64\bin\openssl.cfg  
set Path=%Path%;C:\OpenSSL-Win64\bin
```

Steps to Install OpenSSL

Following are the steps to install OPENSSL in the computer:



Set up the environment variables

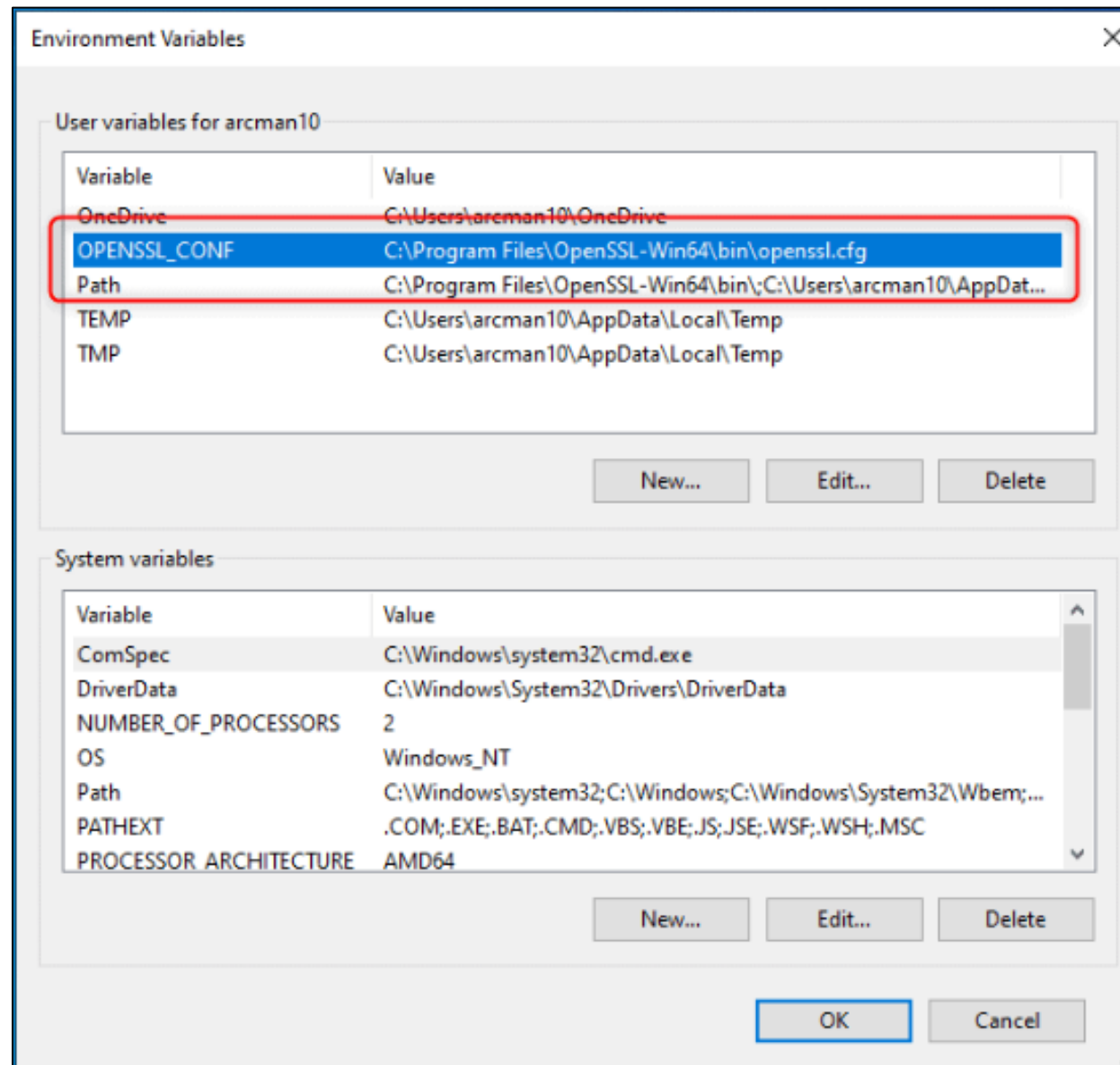


To verify completion, run the OPENSSL binary



Steps to Install OpenSSL

To set up the configurations permanently, in the command prompt enter ' sysdm.cpl '



Advance > Environment Variables

Screenshot courtesy: <https://thesecmaster.com/procedure-to-install-openssl-on-the-windows-platform/>

Steps to Install OpenSSL

Check the completion by entering the following command in the prompt:

```
openssl version
```



Solace PubSub

It is a publish-subscribe, model-based event broker.
It helps in:



Event management



Event streaming



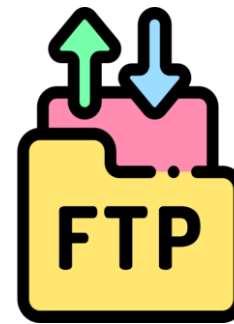
Event insight

Creating FTP User Account in Solace

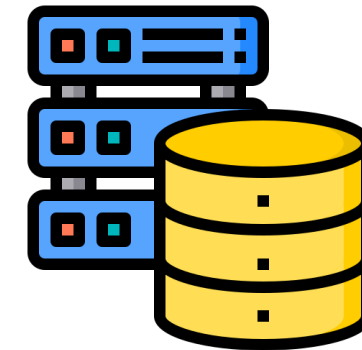
File Transfer Protocol, or FTP user, is an account that sends files to a host computer over a network using FTP services.



Host



FTP Account on a local system



Server



Creating FTP User Account in Solace

Command for creating FTP User account:

```
solace(configure)# create username <name> password  
                        <password> file-transfer
```

Command for changing configurations of FTP User account:

```
solace(configure)# username <name>
```



Steps to Configure Two-Way SSL Authentication for REST

Steps to Configure Two-Way SSL Authentication for REST

There are six steps to configure two-way SSL authentication for REST.
The steps involve:

01

Generation of SSL certificate

02

Enabling the SSL on event broker

03

Verifying the REST over SSL



Steps to Configure Two-Way SSL Authentication for REST

There are six steps to configure two-way SSL authentication for REST.
The steps involve:

04

Generating the client specific certificate

05

Configuring the CAs in event blocker

06

Validating the client authentication



Steps to Configure Two-Way SSL Authentication for REST

Step 1.1

In the first step, create a key to generate SSL certificates.
Type the below command in the command prompt:



```
openssl genrsa -des3 -out root.key 4096
```



Steps to Configure Two-Way SSL Authentication for REST

Step 1.2

Generate a server certificate and signing request using the following command in the command prompt:

```
openssl req -new -x509 -days 1000 -key root.key -out server.pem -subj  
"/C=/ST=/L=/O=/OU=/CN=root"
```


Steps to Configure Two-Way SSL Authentication for REST

Step 1.3

Concatenate the server certificate and the key to create a file, named `server_certificate.pem`.



Key



Server certificate



PEM file

PEM is a file format that stands for privacy enhanced mail.

Steps to Configure Two-Way SSL Authentication for REST

Code for concatenating the server certificate and KEY to create a PEM file:

```
cat root.key > server_certificate.pem  
cat server.pem >> server_certificate.pem
```

Command for saving the certificate using SCP service, in /certificates/ folder:

```
scp -P2222 /home/<username>/server_cert.pem <Solace FTP  
User>@<Host>:/certificates/
```



Steps to Configure Two-Way SSL Authentication for REST

Step 2.1

Enable the SSL for the message VPN (Virtual private network) on the event broker (solace).



Login to the solace CLI using the FTP user



Perform the successive steps to enable SSL on event broker

Steps to Configure Two-Way SSL Authentication for REST

A command for enabling the SSL for the message VPN:

```
solace(configure)# message-vpn <vpn-name>
solace(configure/message-vpn)# service rest
solace(configure/message-vpn/service/rest)# incoming
solace(...ure/message-vpn/service/rest/incoming)# listen-port 9443 ssl
solace(...ure/message-vpn/service/rest/incoming)# no ssl shutdown
```

Steps to Configure Two-Way SSL Authentication for REST

Step 2.2

Use the certificate that was created in step 1 and set it for event broker (solace) using the command:

```
solace(configure)# message-vpn <vpn-name>
solace(configure/message-vpn)# service rest
solace(configure/message-vpn/service/rest)# incoming
solace(...ure/message-vpn/service/rest/incoming)# listen-port 9443 ssl
solace(...ure/message-vpn/service/rest/incoming)# no ssl shutdown
```

Steps to Configure Two-Way SSL Authentication for REST

Step 3.1

In step three get access to the URL to verify the REST over SSL.



Steps to Configure Two-Way SSL Authentication for REST

Command for getting access to the URL using the given https and SSL port:

```
https://<host>:<9443>/
```



Steps to Configure Two-Way SSL Authentication for REST

Step 4.1

In step four, first create a client key to generate the client-specific certificate. The code to create the client key and CSR file is:

```
openssl req -nodes -new -newkey rsa:4096 -keyout client.key -out client.csr -  
subj "/C=/ST=/L=/O=/OU=/CN=Client_User"
```

Steps to Configure Two-Way SSL Authentication for REST

Step 4.2

Client PEM file and signing request for REST client is generated. The command to do so is:

```
openssl x509 -req -in client.csr -CA server.pem -CAkey root.key -CAcreateserial  
-out client.pem -days 1825 -sha256
```

Steps to Configure Two-Way SSL Authentication for REST

Step 4.3

Now, create a client certificate by concatenating root.key and server.pem. The command for this is as follows:

```
cat root.key > server_cert.pem  
cat server.pem >> server_cert.pem
```



Steps to Configure Two-Way SSL Authentication for REST

Step 5.1

In step five, create a certificate authority (CA) and assign the certificates.

```
solace> enable
solace# configure
solace(configure)# authentication
solace(configure/authentication)# create certificate-authority <ca-name>
solace(.../authentication/certificate-authority)# certificate file <ca-certificate>
```

Steps to Configure Two-Way SSL Authentication for REST

Step 5.2

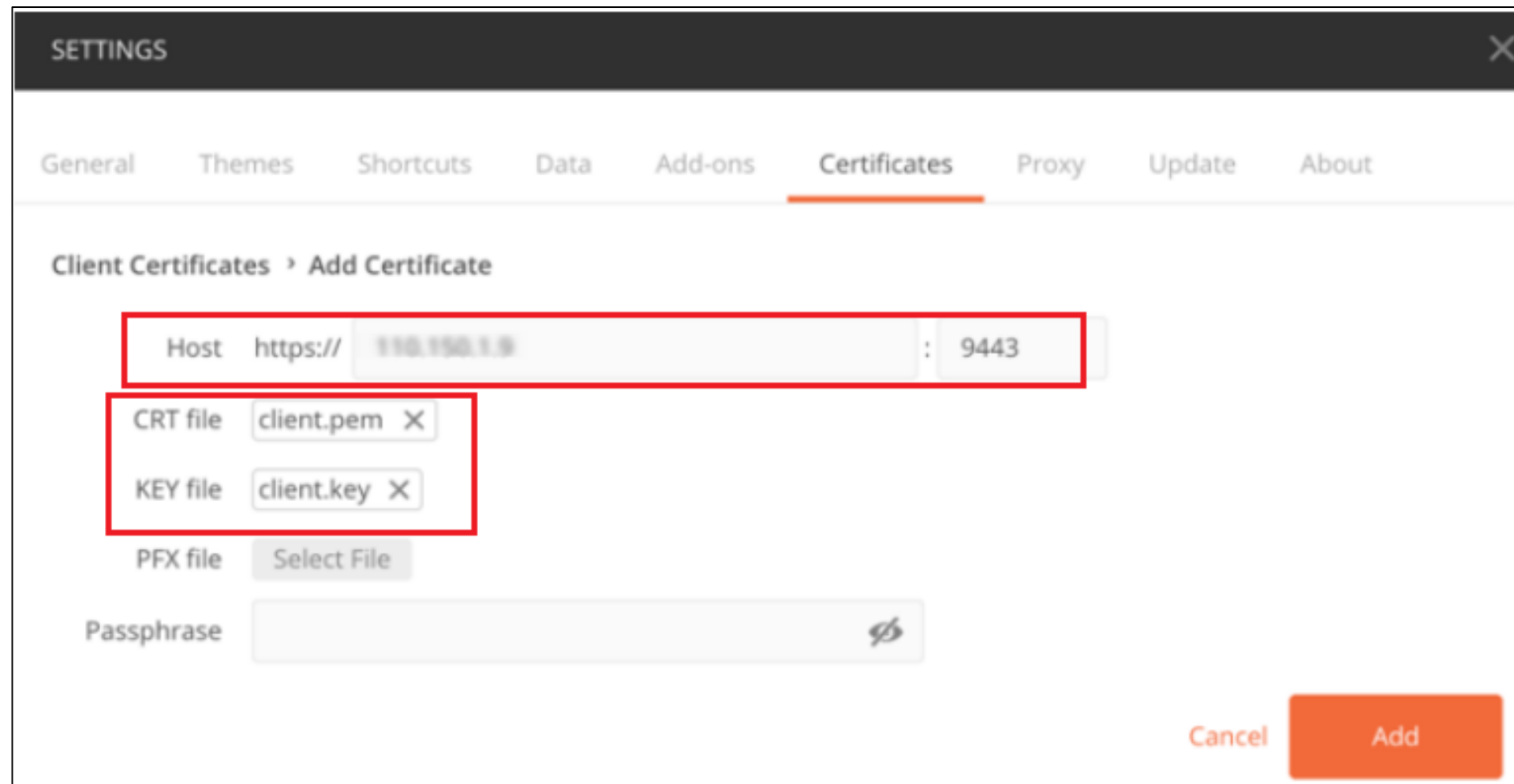
Assign a username source to the message VPN. Use the command given below:

```
solace(configure)# message-vpn <vpn-name>
solace(configure/message-vpn)# authentication
solace(configure/message-vpn/authentication)# client-certificate
solace(...vpn/authentication/client-certificate)# username-source common-name
solace(...vpn/authentication/client-certificate)# no shutdown
```


Steps to Configure Two-Way SSL Authentication for REST

Step 6.1

Finally, in the sixth step, client authentication is done using Postman. First, add a certificate in the Postman by navigating to preference and clicking on the certificates.

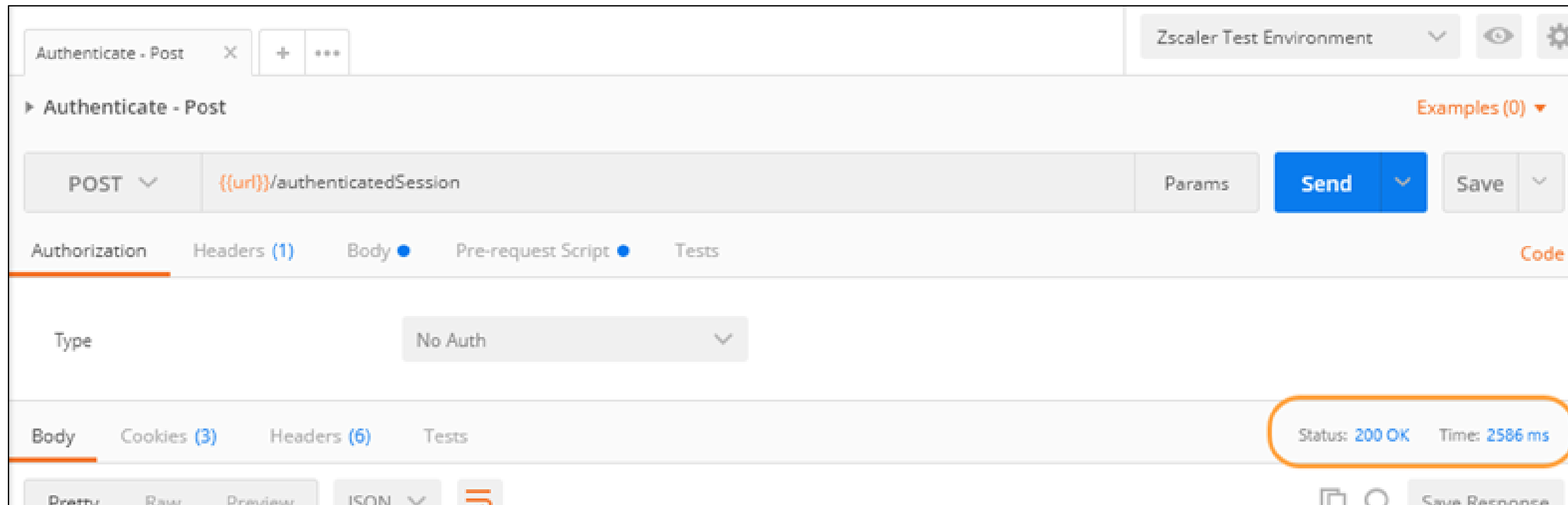


The screenshot shows the 'SETTINGS' window in Postman, specifically the 'Certificates' tab. The 'Client Certificates' section is active, and the 'Add Certificate' button is clicked. The 'Host' field is set to 'https:// 110.150.1.9 : 9443'. The 'CRT file' field is set to 'client.pem' and the 'KEY file' field is set to 'client.key'. The 'PFX file' field is set to 'Select File'. The 'Passphrase' field is empty. The 'Add' button is highlighted in orange.

Screenshot courtesy: <https://docs.solace.com/Security/Two-Way-SSL-Authentication.htm?Highlight=two%20way>

Steps to Configure Two-Way SSL Authentication for REST

In Postman, a 200 OK status indicates successful authentication.



Screenshot courtesy: <https://docs.solace.com/Security/Two-Way-SSL-Authentication.htm?Highlight=two%20way>

Key Takeaways

- Secure Socket Layer (SSL) provides a secure connection by authenticating the key and certificates of the user and server.
- An event broker is a middleware (SaaS) that transfers events.
- Two-way authentication involves the mutual sharing of certificates and credentials.
- Solace PubSub is an event broker, and FTP users can be created with it.

