T7 Cloud Simulation

Connectivity Guide

Table of contents

1. Introduction

This document explains the methods of connecting to your T7 Cloud Simulation instance via the Internet.

Internet access is the only option offered right now.

Connecting to your T7 instance requires that you establish a VPN, using either SSL or GRE and IPsec.

You can choose which option to use when you create your instance, and we will automatically generate a basic configuration file that allows connection via either OpenVPN or strongSwan and compatible packages.

If your network infrastructure does not allow direct software tunnelling, these configuration files will at least serve as a template to allow you to configuration your infrastructure correctly.

Although we provide basic instructions on installing and configuring these packages, we do not directly support either package, and due to the diverse nature of our customers' networks, we are generally unable to assist with network configuration.

2. OpenVPN

OpenVPN is a software VPN solution that uses the industry standard SSL/TLS protocol for data encryption. The OpenVPN tunnel connectivity to T7 Cloud Simulation is strictly a host-to-host solution, meaning that the tunnel to your instance should terminate on a host on your client side.

OpenVPN tunnel connectivity to T7 Cloud Simulation can be terminated on Windows, Linux, or Mac OSX.

2.1 Possible Endpoints

For official documentation and more detailed installation instructions about OpenVPN please visit https://openvpn.net/community-resources/how-to/#install

The following instructions assume that you have downloaded the OpenVPN configuration files associated with your T7 Cloud Simulation instance, available after logging in at the service launchpad or after creating an instance via the REST API.

After the creation of your T7 Cloud Simulation instance, you will be able to download your pre-configured OpenVPN configuration file, as well as associated keys and certificates, to create a VPN tunnel for access to your instance.

NOTE: The OpenVPN endpoint on your instance is designed for use with the OpenVPN client version 2.4.8. Older versions of the client may not be compatible with your instance's server.

2.1.1 Windows

2.1.1.1 Installation

OpenVPN for Windows can be installed from the installation packages on the OpenVPN download page.

OpenVPN will only run on Windows XP or later and must be installed and run by a user who has administrative privileges.

Official OpenVPN Windows installers include OpenVPN-GUI, which allows managing OpenVPN connections from a system tray applet.

During the install process, when confronted with the "Choose Components" state, leave the default options selected.

IMPORTANT: When the installation prompts you to confirm the installation of the "TAP-Windows Adapter V9", select "Continue Anyway". This installs the necessary virtual network interface that the tunnel will be established on.

After you've completed the Windows installer, OpenVPN is ready for use and will associate itself with files having the `.ovpn` extension.

2.1.1.2 Connection initiation

Download the configuration and keys from your instances, then extract the contents of the downloaded .zip file into any directory on your client machine (i.e. the machine from which you want to access T7 on your instance).

Right click on the `client.ovpn` file that you had previously extracted and select Start OpenVPN on this configuration file. OpenVPN will then establish a secure connection to your running instance.

OpenVPN can also be run from the command prompt with a command such as:

```
openvpn client.ovpn
```

It's also possible to run OpenVPN as a service by extracting all files downloaded from your instance to \Program Files\OpenVPN\config and starting the OpenVPN Service, which can be controlled from Start Menu -> Control Panel -> Administrative Tools -> Services.

For more details and trouble-shooting tips for OpenVPN on Windows, check out the official documentation.

2.1.2 Linux

These instructions assume you have root privileges and have access to the appropriate commands (apt-get/yum, unzip, service).

If you are using Debian, Ubuntu, Fedora or RHEL/CentOS/Scientific Linux, OpenVPN is available through your distribution's repositories.

If using *Debian/Ubuntu* run:

```
apt-get install openvpn
```

If using *RHEL/Fedora* run:

```
yum install openvpn
```

Once OpenVPN has been successfully installed, unzip the contents of the "openvpn" directory from the .zip archive for your instance into /etc/openvpn/ and run restart OpenVPN:

```
unzip -j {name of the downloaded zip file} 'openvpn/*' -d /etc/openvpn
systemctl restart openvpn
```

In case you have older system, you might need to use `service` command instead of `systemctl`

```
service openvpn restart
```

If everything has been done correctly and no errors have occurred, you should be able to see the successfully established VPN tunnel to your instance.

Note: If OpenVPN is not available in your repositories, the OpenVPN source can be downloaded here: https://openvpn.net/community-downloads-2

2.1.3 Mac OSX

OpenVPN does not provide an official GUI for Mac OSX, but we have found that the application Tunnelblick works well.

You can find the Tunnelblick download link and official instructions at https://tunnelblick.net/

Once the Tunnelblick .dmg file has been downloaded, double click it. In the new window, double click the "Tunnelblick" icon and confirm the installation. When prompted for configuration files, click "Quit".

Once Tunnelblick has been successfully installed, simply double click XX_XX_XX_XX.tblk provided with the .zip archive downloaded from your instance and your VPN tunnel should automatically be configured.

In the taskbar, click the Tunnelblick icon and connect to your instance.

If everything has been done correctly and no errors have occurred, you should be able to see the successfully established VPN tunnel to your T7 Cloud Simulation instance.

2.2 Firewall ports

Required firewall ports when using OpenVPN:

- OpenVPN tunnel TCP port 1194 (SSL)

3. IPsec/GRE

IPsec/GRE connectivity to T7 Cloud Simulation is an alternate connection method meant to accommodate customer networks that are incompatible with our OpenVPN solution.

Our IPsec/GRE connectivity leaves more of the configuration up to the customer, allowing for flexibility when it comes to larger internal networks whose security policies may not allow for direct host-to-host access.

In order to use IPsec/GRE as your tunnel method, you'll first need to have a static IP assigned to your T7 Cloud Simulation account that you can attach to your instances. You should already have one, but if not please contact our customer technical support team on cts@deutsche-boerse.com.

This solution consists of two components.

The first component is IPsec, which provides encryption of all traffic between your T7 Cloud Simulation instance and your internet-facing entry point in your network.

Because IPsec alone is incapable of transporting multicast traffic (required for T7 Cloud Simulation's market data), a separate GRE tunnel is required. The GRE tunnel is also between the same endpoints as the IPsec tunnel, as the IPsec tunnel is simply meant to just encrypt the traffic sent via the GRE tunnel.

All traffic is sent via the GRE tunnel interface. The T7 software running in your instance is available directly through this GRE tunnel, and all exchange interfaces (market data interfaces, trading interfaces) need to be addressed via this interface.

Because of the flexibility of our IPsec/GRE connection solution, we are unable to provide an officially supported configuration, but this document will list several example network configurations that have been tested and work for us.

To clarify basic terminology, we will refer to the following IP addresses throughout our documentation.

Cloud Simulation instances are using Libreswan for IPsec connections. They are running within a cloud environment, behind a transparent NAT. To configure the connection on your end correctly, you shouldn't need to mention anything else but the instance's public IP address in the left= configuration parameter.

In some cases, however there are configuration options that might need to be tweaked as per your environment. You can select these when generating a new set of keys, to alter libreswan's configuration on our end.

Term	Refers to
<CLOUDSIM_END-POINT_IP>	Publicly addressable IP address assigned to your T7 Cloud Simulation instance.
<CUSTOMER_END-POINT_IP>	Publicly addressable IP address assigned to the interface on which the customer chooses to terminate the IPsec tunnel.

3.1 IPsec Settings

3.1.1 Phase 1 (IKE-AES-256-SHA-DH5) Parameters

Setting	Parameter
Encryption Algorithm	AES256
Hashing Algorithm	SHA-1
Diffie-Hellman Group	Group 5 (1536-bit)
Authentication Mode	Pre-Shared Key
IKE Negotiation Mode	Main
IKE Timeout (ISAKMP)	3600 Seconds (1 Hour)

3.1.2 Phase 2 (ESP-AES-128-SHA-HMAC) Parameters

Setting	Parameter
Encryption Algorithm	AES128
Hashing Algorithm	SHA-1
PFS Diffie-Hellman Group	Group 5 (1536-bit)
IPsec Timeout (SA)	3360 Seconds (56 Minutes)

Please be sure to set your SA lifetime is set to a value higher than 3360. Failure to do so may cause your tunnel to reset every 56 minutes, disconnecting you from your instance.

3.1.3 Pre-Shared Key

The Pre-Shared Key (PSK) for your T7 Cloud Simulation instance is available for download on the instance details from the T7 Cloud Simulation after you have configured your initial set of keys for your instance's IP address. It is also available via the T7 Cloud Simulation REST API.

3.2 GRE Settings

10.8.0.1 is the T7 Cloud Simulation instance's IP address on the GRE tunnel interface. This is the peer on the other side of the GRE tunnel from the customer endpoint. This 10.8.0.1 address is the T7 "backend", being the address of multicast sources as well as TCP/IP connections. Explain how GRE endpoint is the "next router hop"

3.3 IKEv1 Settings

NAT Traversal in IKEv1 is negotiated via Vendor ID options as specified in RFC 3947. However, many implementations only support the draft version of the RFC. Libreswan sends both the RFC and the

most common draft versions (02, 02_n and 03) to maximize interoperability. Unfortunately, there are known broken implementations of RFC 3947, notably Cisco routers that have not been updated to the latest firmware. As the NAT-T payload is sent in the very first packet of the initiator, there is no method to auto-detect this problem and initiate a workaround.

This option allows fine tuning which of the NAT-T payloads to consider for sending and processing. Currently the accepted values are drafts, rfc, both (the default) and none. To interoperate with known broken devices, use nat-ikev1-method=drafts. To prevent the other end from triggering IKEv1 NAT-T encapsulation, set this to none. This will omit the NAT-T payloads used to determine NAT, forcing the other end not to use encapsulation.