Alibaba Cloud
aliyun.com

Best Practices for
Alibaba Cloud

Setting up a VPN using OpenVPN

01 Introduction

The focus of this post is to explain in detail, the process of setting up an OpenVPN on a CentOS 7 server. We will also explore the process to establish the basic connectivity from client to server on Windows, OS X, and Linux.

OpenVPN is an open-source VPN application that lets you create and join a secure private network over the public Internet. There are several best practices documents published that look at other user cases and software involved in setting up VPNs.

02 Prerequisites

A Domain or subdomain that resolves to your server that you can use for the certificates is preferred, however, it is not mandatory for this exercise.

Note: Since OpenVPN is unavailable in the default CentOS repositories, you can setup the EPEL repository managed by the Fedora Project. It contains non-standard but popular packages.

```
yum install epel-release
```

03 Setup

Let us go through the steps below to setup and configure VPN:

OpenVPN Installation

Setup OpenVPN along with Easy RSA to generate SSL key pairs. This will secure the VPN connections.

```
yum install openvpn easy-rsa –y
```

Configuring OpenVPN

OpenVPN has example configuration files in its documentation directory. Copy the sample server.conf file as a starting point for your own configuration file.

```
cp /usr/share/doc/openvpn-*/sample/sample-config-files/server.conf /etc/openvpn
```

Open the file for editing.

```
vi /etc/openvpn/server.conf
```

You would need to change a few commands in this file. Most of the lines just need to be uncommented (remove the ;). The details of the other changes are marked in red.

When we generate our keys at a later stage, the default Diffie-Hellman encryption length for Easy RSA will be 2048 bytes, therefore, change the dh filename to dh2048.pem.

```
dh dh2048.pem
```

We need to uncomment the push "redirect-gateway def1 bypass-dhcp" command, which tells the client to redirect all traffic through our OpenVPN.

push "redirect-gateway def1 bypass-dhcp"

Next, we need to provide DNS servers to the client since it will not be able to use the default DNS servers provided by your internet service provider. In this case, we will be using Google's public DNS servers, 8.8.8.8 and 8.8.4.4.
We can do this by uncommenting the push "dhcp-option" DNS commands and updating the IP addresses.

push "dhcp-option DNS 8.8.8.8"
push "dhcp-option DNS 8.8.4.4"

You need OpenVPN to run with no privileges after starting, therefore, command it to run with a user and group of nobody. To enable this, you will need to uncomment the following commands:

user nobody
group nobody

Save and exit the OpenVPN server configuration file.

Generating Keys and Certificates

Now that we have configured the server, we need to generate our keys and certificates. Easy RSA installations of some scripts generate keys and certificates.

Let's create a directory for the keys to go in.

mkdir -p /etc/openvpn/easy-rsa/keys

Now, copy the key and certificate generation scripts into the directory.

cp -rf /usr/share/easy-rsa/2.0/* /etc/openvpn/easy-rsa

Edit the default values that the script uses so we do not have to type in our information each time. The system stores this information in the “vars" file. Pen this for editing.

vi /etc/openvpn/easy-rsa/vars

Change the values that start with “KEY_". Update the following values to be accurate as per your organization's criteria.

The ones that matter the most are:

- **KEY_NAME:** You should enter server here. You would also have to update the configuration files referenced server.key and server.crt

- **KEY_CN:** Enter the domain or subdomain that resolves to your server.
  For the other values, you can enter information for your organization based on the variable name.

```
. . .

# These are the default values for fields
# which will be placed in the certificate.
# Don't leave any of these fields blank.
export KEY_COUNTRY="US"
export KEY_PROVINCE="NY"
export KEY_CITY="New York"
export KEY_ORG="DigitalOcean"
export KEY_EMAIL="sammy@example.com"
export KEY_OU="Community"
```

```
# X509 Subject Field
export KEY_NAME="server"

. . .

export KEY_CN=openvpn.example.com

. . .
```

We also have to eliminate chances of our OpenSSL configuration not loading due to an undetectable version. For this, we will copy the required configuration file and remove the version number.

```
cp /etc/openvpn/easy-rsa/openssl-1.0.0.cnf /etc/openvpn/easy-rsa/openssl.cnf
```

To start generating our keys and certificates, we need to move into our easy-rsa directory and source in our new variables.

```
cd /etc/openvpn/easy-rsa
source ./vars
```

We will then clean up any keys and certificates that may already be in this folder and generate our certificate authority.

```
./clean-all
```

When you build the certificate authority, you will be asked to enter all the information into the "varsfile". However, you will notice that your options are already set as default. Just press ENTER in each case.

```
./build-ca
```

The next step involves generating the key and certificate for the server. Again, you can just go through the questions and press ENTER for each one to use the default options. Answer Y (yes) to commit the changes.

```
./build-key-server server
```

We also need to generate a Diffie-Hellman key exchange file. This command will take a short while to complete:

```
./build-dh
```

That completes the steps related to the server keys and certificates. Now, copy them all into your OpenVPN directory.

```
cd /etc/openvpn/easy-rsa/keys
cp dh2048.pem ca.crt server.crt server.key /etc/openvpn
```

Your clients will need the certificate to be able to authenticate. Therefore, you need to share these keys and certificates with your clients. As a best practice, generate separate keys and certificates for each client you intend to connect.

Make sure that when you do this, each of them has a descriptive name, but for now, we are going to have one client, and for demonstration purposes, we will just call it client.

```
cd /etc/openvpn/easy-rsa
./build-key client
```

With that, we complete our keys and certificates step.

Routing

To keep things simple, we are going to do our routing directly with “iptables” rather than the new “firewalld”. First, make sure you have installed and enabled the iptables service.

```
yum install iptables-services -y
systemctl mask firewalld
systemctl enable iptables
systemctl stop firewalld
systemctl start iptables
iptables --flush
```

Next, we will add a rule to iptables to forward our routing to OpenVPN subnet and save this rule.

```
iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o eth0 -j MASQUERADE
iptables-save > /etc/sysconfig/iptables
```

Enable IP forwarding in sysctl. Open sysctl.conf for editing.

```
vi /etc/sysctl.conf
```

Add the following command at the top of the file:

```
net.ipv4.ip_forward = 1
```

Then restart the network service and the IP forwarding will take effect.

```
systemctl restart network.service
```

Starting OpenVPN

Now, we are ready to run our OpenVPN service. dd it to systemctl:

```
systemctl -f enable openvpn@server.service
```

Start OpenVPN:

```
systemctl start openvpn@server.service
```

This completes the server-side configuration for OpenVPN.

Configuring a Client

Regardless of your client's operating system, you will definitely need a copy of the CA certificate from the server, along with the client key and certificate.
Locate the following files on the **server**. If you generated multiple client keys with unique descriptive names, then the key and certificate names will be different. We have used client in this case for reference.

```
/etc/openvpn/easy-rsa/keys/ca.crt
/etc/openvpn/easy-rsa/keys/client.crt
/etc/openvpn/easy-rsa/keys/client.key
```

Copy these three files to your **client machine**. You can use SFTP or your preferred method. You also have an option to open the files in your text editor and copy and paste the contents into new files on your client machine.

Make sure you note where you have saved them. We will create a file called client.ovpn in this case. This is a configuration file for an OpenVPN client commanding it to connect to the server.

- You will need to change the first command to reflect the name you gave the client in your key and certificate; in our case, it was just client
- You also need to update the IP address from your_server_ip to the IP address of your server; port1194 remains the same
- Ensure the paths to your key and certificate files are correct

```
client
dev tun
proto udp
remote your_server_ip 1194
resolv-retry infinite
nobind
persist-key
persist-tun
comp-lzo
verb 3
ca /path/to/ca.crt
cert /path/to/client.crt
key /path/to/client.key
```

Now, this file can now be used by any OpenVPN client to connect to your server.

Windows:
On Windows, you will need the official OpenVPN Community Edition binaries which come with a GUI. Then, place your .ovpn configuration file into a proper directory, C:\Program Files\OpenVPN\config, and click Connect in the GUI. OpenVPN GUI on Windows must be executed with administrative privileges.

OS X:
On Mac OS X, the open source application Tunnelblick provides an interface similar to the OpenVPN GUI on Windows and comes with OpenVPN and the required TUN/TAP drivers. As with Windows, the only step required is to place your .ovpn configuration file into the ~/Library/ApplicationSupport/Tunnelblick/Configurations directory. Alternatively, you can double-click on your .ovpn file.

Linux:
On Linux, you should install OpenVPN from your distribution's official repositories. You can then invoke OpenVPN by executing:

```
sudo openvpn --config ~/path/to/client.ovpn
```

04 Conclusion

After you establish a successful client connection, you can verify that your traffic is being routed through the VPN by using Google to reveal your public IP or try https://www.whatismyip.com/.

This document attempts to introduce the concept of setting up a VPN using OpenVPN. While this architecture is not highly available, it can be easily extended to be fail-safe by relying on HA features available with Alibaba Cloud. Please refer to http://intl.aliyun.com for more reference documents.

Further Information

VPC
https://intl.aliyun.com/docs#/pub/vpc_en_us

ECS
https://intl.aliyun.com/docs#/pub/ecs_en_us

intl.aliyun.com