OpenVPN Client on Mikrotik Router

Con Ubuntu 18.04 64bits Para Conectar SmartISP by API



Prepared By

Rodrigo Anrrango

Network Specialist and Consultant

<https://SmartISP.us/install>

Contents

[Introduction 3](#_Toc14251833)

[Network Topology 3](#_Toc14251834)

[Devices 3](#_Toc14251835)

[OpenVPN Server 3](#_Toc14251836)

[Mikrotik 3](#_Toc14251837)

[Server Configurations 3](#_Toc14251838)

[Mikrotik Configurations 8](#_Toc14251839)

[Accessing Client LAN 11](#_Toc14251840)

[Reserve Static IPs for clients 13](#_Toc14251841)

[Troubleshooting 14](#_Toc14251842)

[Useful Links 15](#_Toc14251843)

#

# Introduction

The goal of this project is to configure an Ubuntu OpenVPN server and a Mikrotik OpenVPN client.

# Network Topology

In this project we are using the following topology:



# Devices

## OpenVPN Server

A VPS server with Ubuntu 18.04.2 LTS.

## Mikrotik

A Mikrotik RB951Ui-2HnD router as OpenVPN client.

# Server Configurations

Run the following commands as root:

*Step 1: Install OpenVPN*

apt-get update

apt-get install openvpn easy-rsa

*Step 2: Setup CA*

make-cadir certificates && cd certificates

*Step 3: Configure the CA Variables*

nano vars

Set the following according to your requirement:

export KEY\_COUNTRY="US"

export KEY\_PROVINCE="NY"

export KEY\_CITY="New York City"

export KEY\_ORG="DigitalOcean"

export KEY\_EMAIL="admin@example.com"

export KEY\_OU="Community"

(Only edit the values in red)

Change:

export KEY\_CONFIG=`$EASY\_RSA/whichopensslcnf $EASY\_RSA`

To:

export KEY\_CONFIG="$EASY\_RSA/openssl-1.0.0.cnf"

Save and close the file.

source vars

*Step 4: Build the Certificate Authority*

./clean-all && ./build-ca

Press Enter through all the prompts.

*Step 5: Create the Server Certificate, Key, and Encryption Files*

./build-key-server server

Accept the default values by pressing **ENTER**. Do *not* enter a challenge password for this setup. Towards the end, you will have to enter **y** to two questions to sign and commit the certificate.

./build-dh

openvpn --genkey --secret keys/ta.key

*Step 6: Client Setup*

source vars && ./build-key client

*Step 7: Configure the OpenVPN Service*

cp keys/{server.crt,server.key,ca.crt,dh2048.pem,ta.key} /etc/openvpn

gunzip -c /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz | tee /etc/openvpn/server.conf

Adjust the OpenVPN Configuration:

nano /etc/openvpn/server.conf

Set the following:

cipher AES-128-CBC

Change protocol from UDP to TCP:

proto tcp

;proto udp

LZO Compression should be disabled:

;comp-lzo

Uncomment duplicate common name setting (this allows multiple clients to connect to the server using the same certificate):

duplicate-cn

Comment this setting (otherwise it produces HMAC errors):

;tls-auth ta.key 0

Comment this setting (not compatible with TCP):

;explicit-exit-notify 1

Save and close the file

*Step 8: Adjust the Server Networking Configuration*

nano /etc/sysctl.conf

Allow IP Forwarding:

# Uncomment:

net.ipv4.ip\_forward=1

Save and close the file.

Aplicar:

sysctl -w net.ipv4.ip\_forward=1

Adjust the UFW Rules to Masquerade Client Connections:

nano /etc/ufw/before.rules

Add the part in red (replace eth0 with your actual interface):

#

# rules.before

#

# Rules that should be run before the ufw command line added rules. Custom

# rules should be added to one of these chains:

# ufw-before-input

# ufw-before-output

# ufw-before-forward

#

# START OPENVPN RULES

# NAT table rules

\*nat

:POSTROUTING ACCEPT [0:0]

# Allow traffic from OpenVPN client to eth0

-A POSTROUTING -s 10.8.0.0/8 -o eth0 -j MASQUERADE

COMMIT

# END OPENVPN RULES

# Don't delete these required lines, otherwise there will be errors

\*filter

. . .

Save and close the file.

nano /etc/default/ufw

Change the value from DROP to ACCEPT:

DEFAULT\_FORWARD\_POLICY="ACCEPT"

Save and close the file.

Open the OpenVPN Port and Enable the Changes:

ufw allow 1194

ufw allow OpenSSH

ufw disable

ufw enable

*Step 9: Start and Enable the OpenVPN Service*

systemctl start openvpn@server

systemctl status openvpn@server

The status should be active (running).

systemctl enable openvpn@server

*Step 10: Generate Additional Client Configurations*

To generate configuration files for additional clients, enter the following commands:

cd certificates

source vars && ./build-key client1

RBs 2

source vars && ./build-key client2

This will generate all configuration files for client1 which may be downloaded from /root/certificates/keys/ directory.

# *Step 11:* Mikrotik Configurations

*Step 1: Copy Certificates and Keys*

Go to /etc/openvpn/ directory on OpenVPN server and download this file to your local computer:

ca.crt

Go to /root/certificates/keys/ directory and download these file as well:

client1.crt client1.key

Open Mikrtok router using Winbox and drag and drop these files:



*Step 2: Installing Certificates*

Go to System -> Certificates and import ca.crt:

Similarly import client1.crt and client1.key.



*Step 3: Create a PPP Profile*

Go to PPP -> Profiles and create a new profile. Give it a name of your choice. Go to Protocols and disable MPLS and IPv6. “Use Encryption” option should be set to “required”. Go to Limits tab and set Only One to “yes”. Save this profile.

*Step 4: Create an OVPN Client Connection*

Go to PPP -> Interface and add a new “OVPN Client” interface. Give it a name of your choice. Go to “Dial Out” tab and set the following properties:



The username and password can be anything. However, it is mandatory and cannot be blank. If everything went well your VPN should be connected.

# Reserve Static IPs for clients OpenVPN

First allow client-config option in OpenVPN config file:

nano /etc/openvpn/server.conf

Go to the following line:

;client-config-dir ccd

Uncomment it and change it to:

client-config-dir ccd

Save and exit the file.

Enter the following commands:

mkdir /etc/openvpn/ccd

touch /etc/openvpn/ccd/client1

Create a CCD file for the client. The filename should be the same as the client name:

nano /etc/openvpn/ccd/client1

ifconfig-push 10.8.0.254 10.8.0.1

Save and close the file. These settings will make sure that client1 is always assigned the IP address 10.8.0.254 upon connection with gateway IP 10.8.0.1.

Next step is to reserve this IP in the ifconfig-pool-persist file. Enter the following commands:

nano /etc/openvpn/ipp.txt

client1,10.8.0.254

Save and close the file and restart the OpenVPN service:

systemctl restart openvpn.service

# Troubleshooting

To troubleshoot you can go to OpenVPN server and run the following command to see Syslogs related to OpenVPN:

tailf /var/log/syslog | grep vpn

Any errors encountered during connection will be displayed here.

# Useful Links

<https://linuxconfig.org/openvpn-setup-on-ubuntu-18-04-bionic-beaver-linux>

<https://github.com/missinglink/mikrotik-openvpn-client>