

Lab 2: IPSec VPN

Introduction

The cloud paradigm changes the way applications are developed, architected, and operated. Improved agility, speed of development, and increased automation in operations allow enterprises to truly benefit from cloud infrastructure.

Usually the enterprise would want to setup encrypted connection from on-prem to Oracle cloud for performing POC's and do some small migrations. Later enterprises can later setup Fastconnect, out dedicated private connection from on-premises data center to Oracle cloud.

Objective

This lab walks you through the steps needed to create a VPN connection from a Linux VM to an OCI VPN Connect. On the public VM from the previous lab, we will install an open-source VPN software. In another region, we will create a VPN connection to this VM.

Pre-requisites

You need the setup from Lab1 to continue working on this Lab

Process Overview

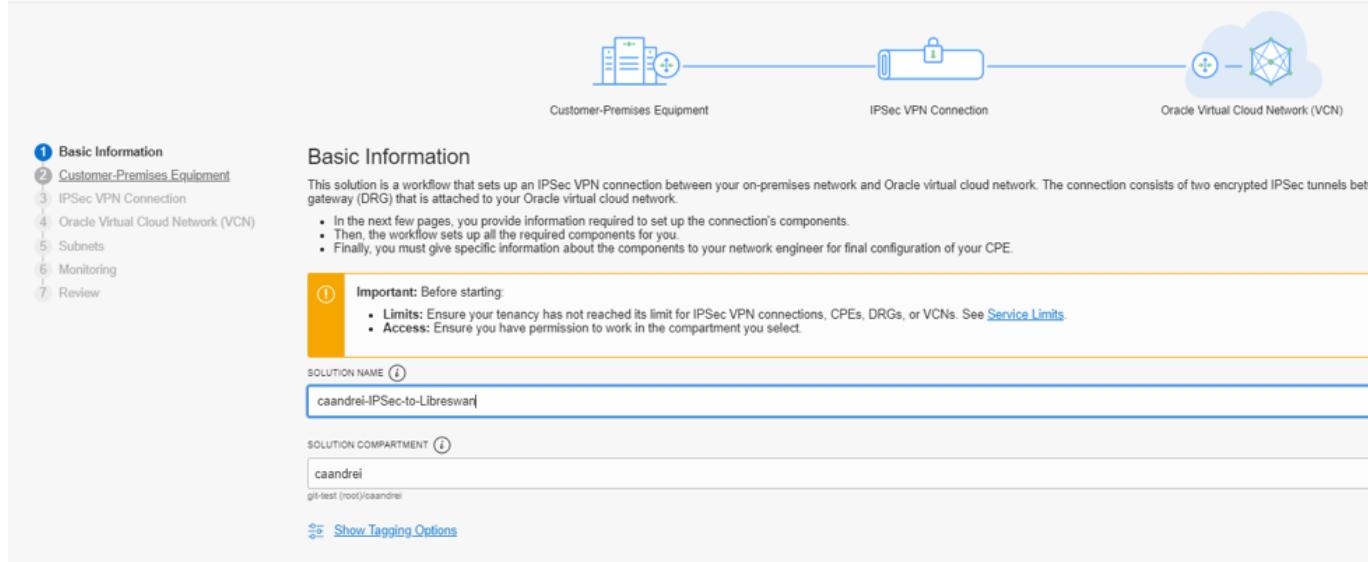
- [Create an OCI IPSec Connection](#)
- [Install and Configure Libreswan](#)
- [Setup Firewall Rules](#)
- [Install Quagga](#)
- [Tests](#)

Create an OCI IPSec Connection

Navigate to Networking > Overview and click the "Start VPN Wizard".

Fill in the Name of the IPSec Connection

Create IPSec VPN Connection



Fill in the public IP address of the public VM created in Lab1 and select Libreswan from the Vendor dropdown. Select the Platform/Version

Create IPSec VPN Connection

The screenshot shows the configuration for the Customer-Premises Equipment (CPE) part of the IPSec VPN connection. It includes sections for basic information, vendor selection, and platform/version selection.

Customer-Premises Equipment

The customer-premises equipment (CPE) is a router or other device on the edge of your on-premises network. The IPSec VPN connects to this device, which your network engineer must later

- Here you create a **virtual representation** in Oracle Cloud Infrastructure of your CPE (or select an existing one if you already have one).
- You must provide your CPE's **public IP address**, which you get from your network engineer. You cannot change this value later.

Customer-Premises Equipment (CPE)

CREATE NEW SELECT EXISTING

CPE NAME: CPE-caandrei-IPSec-to-Libreswan

CPE PUBLIC IP ADDRESS: 152.67.130.140

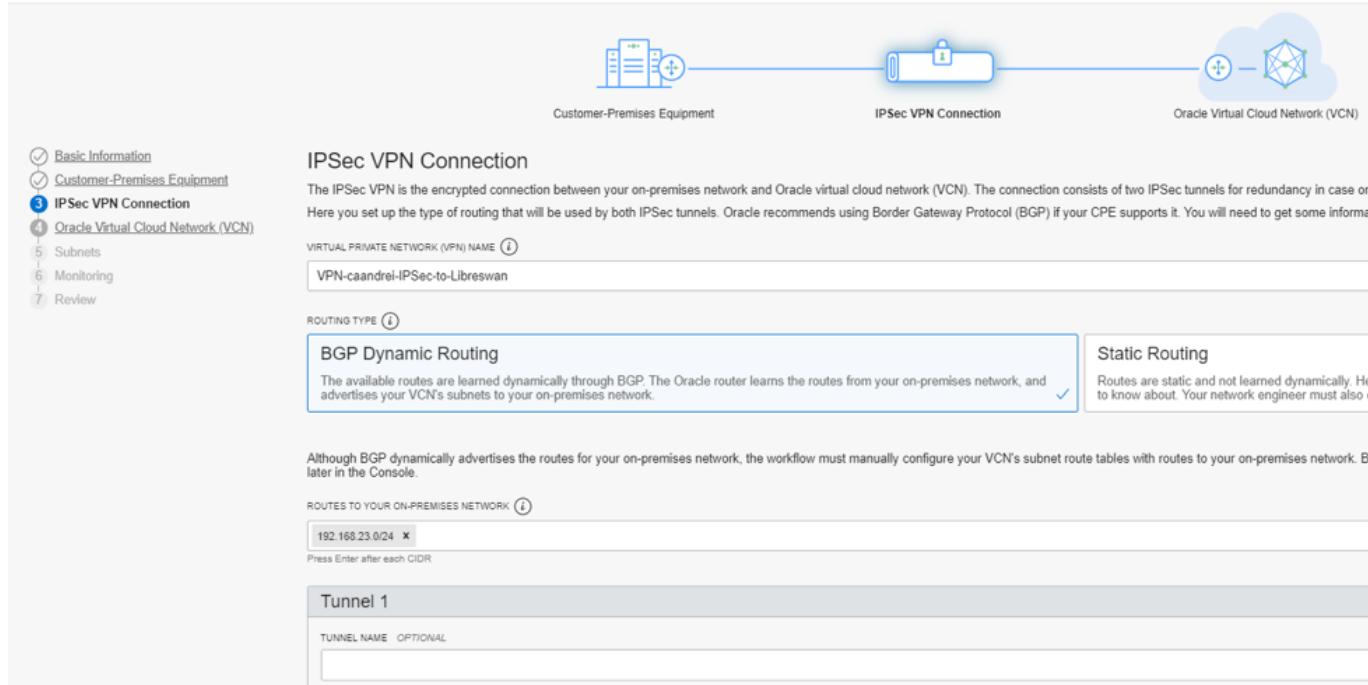
CPE Vendor Information

VENDOR: Libreswan

PLATFORM/VERSION: 3.18 or later

Fill in the CIDR space of the networks that are behind the Libreswan (192.168.23.0/24)

Create IPSec VPN Connection



Fill in the name of the Tunnel1, and the BGP information.

We will use ASN65000 and a /30 for connectivity between BGP peers: 10.10.10.0/30.

The CPE will use 10.10.10.1/30 and the oracle side will use 10.10.10.2/30

Create IPSec VPN Connection

The diagram shows the connection between Customer-Premises Equipment (CPE), IPSec VPN Connection, and Oracle Virtual Cloud Network (VCN).

Tunnel 1

TUNNEL NAME: OPTIONAL

Tunnel1

YOUR BGP ASN: 65000

Oracle's BGP ASN is 31898

INSIDE TUNNEL INTERFACE - CPE: 10.10.10.1/30

INSIDE TUNNEL INTERFACE - ORACLE: 10.10.10.2/30

SHARED SECRET: AUTO GENERATE SHARED SECRET PROVIDE CUSTOM SHARED SECRET

Fill in the name of the Tunnel2, and the BGP information.

We will use ASN65000 and a /30 for connectivity between BGP peers: 10.10.10.4/30.

The CPE will use 10.10.10.5/30 and the oracle side will use 10.10.10.6/30

Create IPSec VPN Connection

The diagram illustrates the setup of an IPSec VPN connection. It shows a 'Customer-Premises Equipment' icon connected to a central 'IPSec VPN Connection' icon, which is then connected to an 'Oracle Virtual Cloud Network (VCN)' icon.

Basic Information

Customer-Premises Equipment

IPSec VPN Connection

Oracle Virtual Cloud Network (VCN)

Tunnel 2

TUNNEL NAME (OPTIONAL): Tunnel2

YOUR BGP ASN: 65000
Oracle's BGP ASN is 31898

INSIDE TUNNEL INTERFACE - CPE: 10.10.10.2/30

INSIDE TUNNEL INTERFACE - ORACLE: 10.10.10.6/30

SHARED SECRET: AUTO GENERATE SHARED SECRET PROVIDE CUSTOM SHARED SECRET

Specify the CIDR space for the VCN from OCI: 192.168.24.0/24

Create IPSec VPN Connection

The screenshot shows the configuration steps for creating an IPSec VPN connection:

- Virtual Cloud Network (VCN)**: CREATE NEW SELECT EXISTING. VCN NAME: VCN-caandrei-IPSec-to-Libreswan
- CIDR BLOCK**: 192.168.24.0/24
- Dynamic Routing Gateway (DRG)**: CREATE NEW SELECT EXISTING. DRG NAME: DRG-caandrei-IPSec-to-Libreswan
- Internet Gateway**: An Internet gateway lets you quickly connect to an instance in a public subnet in your VCN. INTERNET GATEWAY NAME: Internet Gateway-caandrei-IPSec-to-Libreswan

On the next screen, you will have the details of the networking artifacts that will be created.

Create IPSec VPN Connection

The diagram illustrates the connection setup. On the left, 'Customer-Premises Equipment' is represented by a server icon. A blue line labeled 'IPSec VPN Connection' connects it to a central lock icon. From the lock icon, another blue line extends to the right, ending at a cloud icon representing 'Oracle Virtual Cloud Network (VCN)'.

Subnets

Here are the other components that will be automatically set up in your VCN.

Subnet ⓘ

SUBNET NAME ⓘ

Subnet-caandrei-IPSec-to-Libreswan

CIDR: No Value
Subnet Access: Public

Route Table ⓘ

Route Table Name: Default Route Table for VCN-caandrei-IPSec-to-Libreswan
Rules are automatically added for the associated internet gateway and dynamic routing gateway.

Security List ⓘ

Security List Name: Default Security List for VCN-caandrei-IPSec-to-Libreswan
Rules will be automatically added to allow:

- SSH traffic to the VCN from any location
- All types of traffic to the VCN from your on-premises network
- All types of traffic from the VCN to any location

On the next screen you will have the Monitoring options

The diagram is identical to the previous one, showing the connection between Customer-Premises Equipment and Oracle Virtual Cloud Network (VCN) via an IPSec VPN Connection.

Monitoring ⓘ Recommended

Set up alarms on your IPSec VPN tunnels.

When you set up these default alarms, you will get notified when either tunnel in your IPSec VPN Connection is down. The alarm fires when the TunnelState metric in the oci_vpn metric namespace where the notifications go. You can edit the alarms later.

ADD DEFAULT TUNNEL ALARMS ⓘ

The last screen will present the overall configuration that will be created.

Create IPSec VPN Connection

Review and Create the Solution

Basic Information

- Solution Name: caandrei-IPSec-to-Libreswan
- Solution Compartment: caandrei
- Tags: IPSec_VPN_Connection: IPSec_VPN_Connection-2020-05-21T10:08:55

Customer-Premises Equipment (CPE)

- CPE Name: CPE-caandrei-IPSec-to-Libreswan
- CPE Public IP Address: 152.67.130.140
- Vendor: Libreswan
- Platform/Version: 3.18 or later

IPSec VPN Connection

- Virtual Private Network (VPN) Name: VPN-caandrei-IPSec-to-Libreswan
- Routing Type: BGP Dynamic Routing
- Routes to Your On-Premises Network: 192.168.23.0/24

[Previous](#) [Create Solution](#) [Cancel](#)

[Terms of Use and Privacy](#) [Cookie Preferences](#)

Push the “Create Solution” button. You will have the progress of the provisioning:

Provisioning IPSec VPN Connection

Progress Bar: Provisioning 5 out of 8 Resources...

Next Steps

① Use the CPE Configuration Helper
The CPE Configuration Helper produces content that helps a network engineer configure your CPE and complete the IPSec connection.
You can open the Helper here, or from the IPSec connection page, or an individual tunnel's page.

[Open CPE Configuration Helper](#)

Create Alarms **Review Technical Documentation**

IPSec VPN Connection Solution Details

Basic Information

- Solution Name: caandrei-IPSec-to-Libreswan
- Solution Compartment: caandrei
- Tags: IPSec_VPN_Connection: IPSec_VPN_Connection-2020-05-21T10:08:55

[Copy Solution to Clipboard](#)

Wait a few minutes and click on the "Open CPE Configuration Helper".

You will see the summary of what was configured on the OCI side.

CPE Configuration Helper

The Helper creates content that helps a network engineer configure your CPE. Fill in any requested information, click **Create Content**, and give the content to your network engineer.

Customer-Premises Equipment Information

CPE Name: CPE-caandrei-IPSec-to-Libreswan CPE Public IP Address: 152.67.130.140

Info To edit the CPE vendor information, go to the [CPE page](#) and click Edit.

CPE Vendor Information

Vendor: Libreswan Platform/Version: 3.18 or later

IPSec Connection

Name: VPN-caandrei-IPSec-to-Libreswan	Static Route CIDRs:
Tunnel 1 Name: Tunnel1	Tunnel 2 Name: Tunnel2
Your BGP ASN: 65000	Your BGP ASN: 65000
Oracle BGP ASN: 31898	Oracle BGP ASN: 31898
Inside Tunnel Interface-CPE: 10.10.10.1/30	Inside Tunnel Interface-CPE: 10.10.10.5/30
Inside Tunnel Interface-Oracle: 10.10.10.2/30	Inside Tunnel Interface-Oracle: 10.10.10.6/30

Create Content [Cancel](#)

Push the "Create Content" button

CPE Configuration Helper

Configuration Output

```

#-----#
# Libreswan Configuration Template
# This template consolidates all VPN Connect information for your IPSec connection from the Oracle Console.
#
#-----#
# The following list of parameters was gathered from your Oracle Console session.
#
#-----#
# VPN PARAMETERS
# 152.67.130.140 = The public IP address for the CPE, as defined in the customer-premises equipment object. This is the IP address of your outside interface
# 152.67.130.140 = The CPE IKE identifier as configured in the Oracle Console. If you did not configure a custom IKE identifier, this value defaults to the CPE public IP address.
# 140.204.35.3 = The Oracle public IP endpoint obtained from the Oracle Console.
# 1 = The IKE version selection for this tunnel (IKEv1 or IKEv2) as configured in the Oracle Console
# zEheANYnZNCHyAyByz3MS4w3wW0cdY8wuyR8lFlLux7Dp11bXafq7Wa5VZuGE7 = The shared secret for this IPSec tunnel. By default, Oracle provides the shared secret for the tunnel.
# Bgp = Your chosen routing type for this tunnel (BGP or static).
#
# 65000 = Your BGP ASN.
# 10.10.10.1 = The CPE inside tunnel IP, also used for BGP peering. This IP address must be part of the IPSec VPN's encryption domain.
# 10.10.10.2 = The Oracle inside tunnel IP, also used for BGP peering. This IP address must be part of the IPSec VPN's encryption domain.
#
#-----#
# VPN PARAMETERS
# 152.67.130.140 = The public IP address for the CPE, as defined in the customer-premises equipment object. This is the IP address of your outside interface
# 152.67.130.140 = The CPE IKE identifier as configured in the Oracle Console. If you did not configure a custom IKE identifier, this value defaults to the CPE public IP address.
# 140.204.35.14 = The Oracle public IP endpoint obtained from the Oracle Console.
# 1 = The IKE version selection for this tunnel (IKEv1 or IKEv2) as configured in the Oracle Console.
# UOb3h0Auha6aedq9ksbSnJNP194UY10XMqs6VFDMNOmMrHRSzgb09KUSeJxaW = The shared secret for this IPSec tunnel. By default, Oracle provides the shared secret for the tunnel.
# Bgp = Your chosen routing type for this tunnel (BGP or static).
#
# 65000 = Your BGP ASN.
# 10.10.10.5 = The CPE inside tunnel IP, also used for BGP peering. This IP address must be part of the IPSec VPN's encryption domain.
# 10.10.10.6 = The Oracle inside tunnel IP, also used for BGP peering. This IP address must be part of the IPSec VPN's encryption domain.
#
#-----#
# USEFUL LINKS:
#
# Libreswan CPE Guide: https://docs.cloud.oracle.com/iaas/Content/Network/Reference/libreswanCPE.htm
#
# Supported IPSec Parameters: https://docs.cloud.oracle.com/iaas/Content/Network/Reference/supportedIPsecparams.htm

```

Copy Configuration To Clipboard

[Close](#)

Download the configuration

Install and Configure Libreswan

Follow the official documentation for Libreswan [here](#).

Scroll to the “Configuration Process” and start with Task 1. This will enable the routing on the Linux instance.

Login to the public VM from Lab1. Switch user to root and edit the /etc/sysctl.conf file

```
nano /etc/sysctl.conf
```

```
root@caandrei-vpn-01:~  
[root@caandrei-vpn-01 ~]# login as: opc  
[root@caandrei-vpn-01 ~]# Authenticating with public key "training"  
Last login: Thu May 21 10:36:37 2020 from 188.27.166.79  
[opc@caandrei-vpn-01 ~]$ sudo su -  
Last login: Thu May 21 10:36:41 GMT 2020 on pts/0  
[root@caandrei-vpn-01 ~]# nano /etc/sysctl.conf
```

Add the routing information

```
root@caandrei-vpn-01:~  
GNU nano 2.3.1          File: /etc/sysctl.conf          Modified ^  
  
# sysctl settings are defined through files in  
# /usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.  
#  
# Vendors settings live in /usr/lib/sysctl.d/.  
# To override a whole file, create a new file with the same in  
# /etc/sysctl.d/ and put new settings there. To override  
# only specific settings, add a file with a lexically later  
# name in /etc/sysctl.d/ and put new settings there.  
#  
# For more information, see sysctl.conf(5) and sysctl.d(5).  
net.ipv4.ip_forward=1  
net.ipv4.conf.all.accept_redirects = 0  
net.ipv4.conf.all.send_redirects = 0  
net.ipv4.conf.default.send_redirects = 0  
net.ipv4.conf.ens3.send_redirects = 0  
net.ipv4.conf.default.accept_redirects = 0  
net.ipv4.conf.ens3.accept_redirects = 0  
  
^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text  ^C Cur Pos  
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

Please be aware that in the documentation, the interface name of the VM is eth0. You can check the name of your interface with ifconfig command.

In my screenshot, the interface name is ens3.

Save the config with ctrl+X and type "Y" to save

```
root@caandrei-vpn-01:~
```

GNU nano 2.3.1 File: /etc/sysctl.conf Modified

```
# sysctl settings are defined through files in
# /usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.
#
# Vendors settings live in /usr/lib/sysctl.d/.
# To override a whole file, create a new file with the same in
# /etc/sysctl.d/ and put new settings there. To override
# only specific settings, add a file with a lexically later
# name in /etc/sysctl.d/ and put new settings there.
#
# For more information, see sysctl.conf(5) and sysctl.d(5).
net.ipv4.ip_forward=1
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0
net.ipv4.conf.ens3.send_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.ens3.accept_redirects = 0
```

Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?

Y Yes
N No ^C Cancel

Hit "Enter" and confirm the filename.

```
root@caandrei-vpn-01:~
```

GNU nano 2.3.1 File: /etc/sysctl.conf Modified

```
# sysctl settings are defined through files in
# /usr/lib/sysctl.d/, /run/sysctl.d/, and /etc/sysctl.d/.
#
# Vendors settings live in /usr/lib/sysctl.d/.
# To override a whole file, create a new file with the same in
# /etc/sysctl.d/ and put new settings there. To override
# only specific settings, add a file with a lexically later
# name in /etc/sysctl.d/ and put new settings there.
#
# For more information, see sysctl.conf(5) and sysctl.d(5).
net.ipv4.ip_forward=1
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0
net.ipv4.conf.ens3.send_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.ens3.accept_redirects = 0
```

File Name to Write: /etc/sysctl.conf

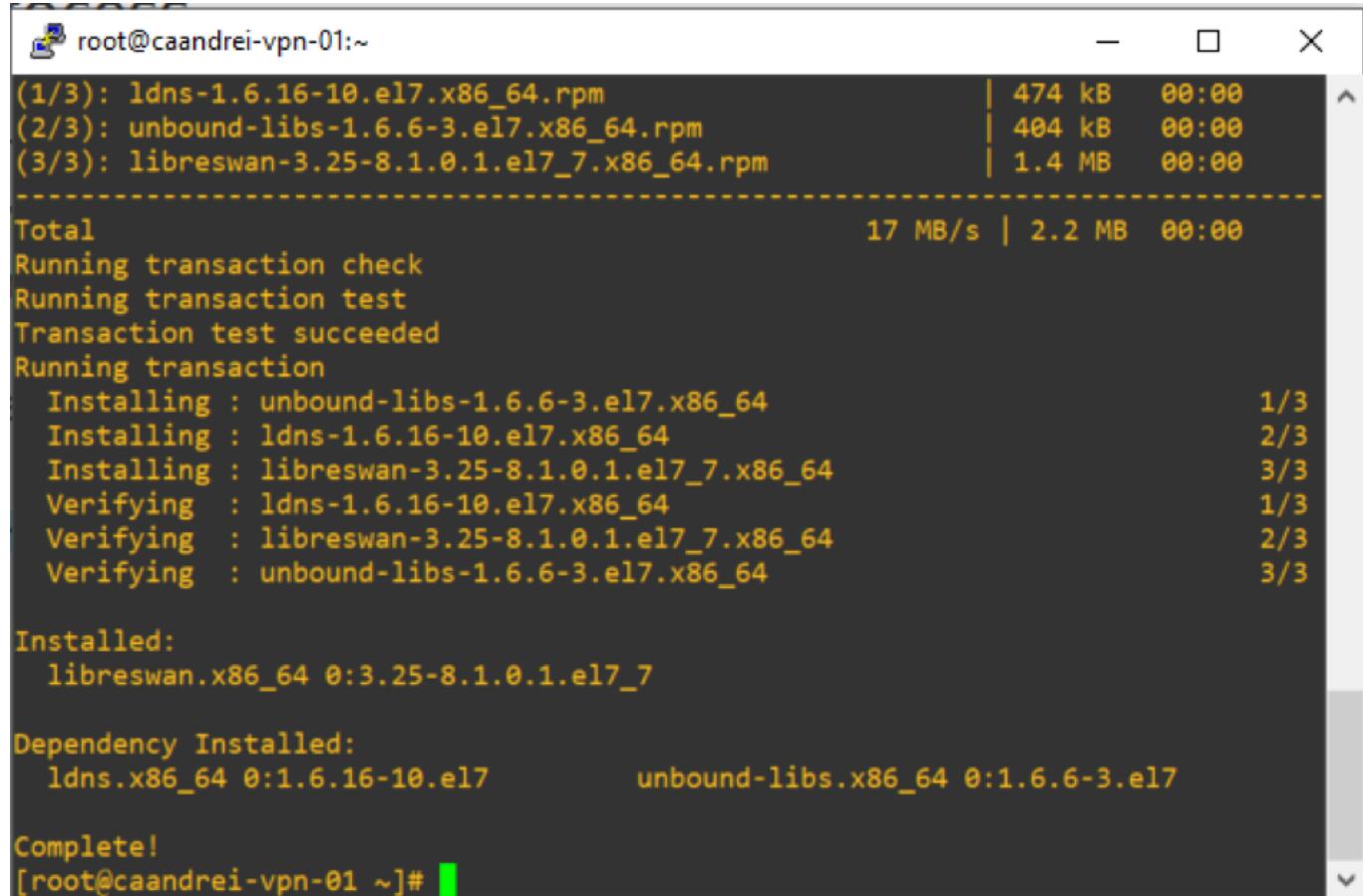
^G Get Help M-D DOS Format M-A Append M-B Backup File
^C Cancel M-M Mac Format M-P Prepend

Reload the routing configuration with

```
sudo sysctl -p
```

Now install the Libreswan software

```
sudo yum install libreswan -y
```



```
root@caandrei-vpn-01:~ (1/3): ldns-1.6.16-10.el7.x86_64.rpm | 474 kB 00:00  
(2/3): unbound-libs-1.6.6-3.el7.x86_64.rpm | 404 kB 00:00  
(3/3): libreswan-3.25-8.1.0.1.el7_7.x86_64.rpm | 1.4 MB 00:00  
-----  
Total 17 MB/s | 2.2 MB 00:00  
Running transaction check  
Running transaction test  
Transaction test succeeded  
Running transaction  
  Installing : unbound-libs-1.6.6-3.el7.x86_64 1/3  
  Installing : ldns-1.6.16-10.el7.x86_64 2/3  
  Installing : libreswan-3.25-8.1.0.1.el7_7.x86_64 3/3  
  Verifying : ldns-1.6.16-10.el7.x86_64 1/3  
  Verifying : libreswan-3.25-8.1.0.1.el7_7.x86_64 2/3  
  Verifying : unbound-libs-1.6.6-3.el7.x86_64 3/3  
  
Installed:  
  libreswan.x86_64 0:3.25-8.1.0.1.el7_7  
  
Dependency Installed:  
  ldns.x86_64 0:1.6.16-10.el7           unbound-libs.x86_64 0:1.6.6-3.el7  
  
Complete!  
[root@caandrei-vpn-01 ~]#
```

Proceed to Task 3 from the documentation and create the config file. Copy-paste the template in an editor (Notepad) and replace the variables with the information that we already have.

```
conn oracle-tunnel-1
    left=152.67.130.140
    # leftid=${cpePublicIpAddress} # See preceding note about 1-1 NAT device
    right=140.204.35.3
    authby=secret
    leftsubnet=0.0.0.0/0
    rightsubnet=0.0.0.0/0
    auto=start
    mark=5/0xffffffff # Needs to be unique across all tunnels
    vti-interface=vti1
    leftvti=10.10.10.1/30
    vti-routing=no
    ikev2=no # To use IKEv2, change to ikev2=insist
    ike=aes_cbc256-sha2_384;modp1536
    phase2alg=aes_gcm256;modp1536
    encapsulation=yes
    ikelifetime=28800s
    salifetime=3600s
conn oracle-tunnel-2
    left=152.67.130.140
    # leftid=${cpePublicIpAddress} # See preceding note about 1-1 NAT device
    right=140.204.35.14
    authby=secret
    leftsubnet=0.0.0.0/0
    rightsubnet=0.0.0.0/0
    auto=start
    mark=6/0xffffffff # Needs to be unique across all tunnels
    vti-interface=vti2
    leftvti=10.10.10.5/30
    vti-routing=no
    ikev2=no # To use IKEv2, change to ikev2=insist
    ike=aes_cbc256-sha2_384;modp1536
    phase2alg=aes_gcm256;modp1536
    encapsulation=yes
    ikelifetime=28800s
    salifetime=3600s
```

Notice that two things are modified from the template: the *vti* interface (I used *vti1* and *vti2*) and the ip addresses assigned to the *vti* interface ("*leftvti*=")

Take the config from the editor and create the config file (*/etc/ipsec.d/oci-ipsec.conf*)

```

root@caandrei-vpn-01:~ 
(1/3): ldns-1.6.16-10.el7.x86_64.rpm | 474 kB 00:00
(2/3): unbound-libs-1.6.6-3.el7.x86_64.rpm | 404 kB 00:00
(3/3): libreswan-3.25-8.1.0.1.el7_7.x86_64.rpm | 1.4 MB 00:00
Total 17 MB/s | 2.2 MB 00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : unbound-libs-1.6.6-3.el7.x86_64 1/3
  Installing : ldns-1.6.16-10.el7.x86_64 2/3
  Installing : libreswan-3.25-8.1.0.1.el7_7.x86_64 3/3
  Verifying : ldns-1.6.16-10.el7.x86_64 1/3
  Verifying : libreswan-3.25-8.1.0.1.el7_7.x86_64 2/3
  Verifying : unbound-libs-1.6.6-3.el7.x86_64 3/3

Installed:
  libreswan.x86_64 0:3.25-8.1.0.1.el7_7

Dependency Installed:
  ldns.x86_64 0:1.6.16-10.el7           unbound-libs.x86_64 0:1.6.6-3.el7

Complete!
[root@caandrei-vpn-01 ~]# nano /etc/ipsec.d/oci-ipsec.conf

```

Follow Task 4 and create the secrets file.

The file will look like this:

```

root@caandrei-vpn-01:~ 
  Installing : unbound-libs-1.6.6-3.el7.x86_64 1/3
  Installing : ldns-1.6.16-10.el7.x86_64 2/3
  Installing : libreswan-3.25-8.1.0.1.el7_7.x86_64 3/3
  Verifying : ldns-1.6.16-10.el7.x86_64 1/3
  Verifying : libreswan-3.25-8.1.0.1.el7_7.x86_64 2/3
  Verifying : unbound-libs-1.6.6-3.el7.x86_64 3/3

Installed:
  libreswan.x86_64 0:3.25-8.1.0.1.el7_7

Dependency Installed:
  ldns.x86_64 0:1.6.16-10.el7           unbound-libs.x86_64 0:1.6.6-3.el7

Complete!
[root@caandrei-vpn-01 ~]# nano /etc/ipsec.d/oci-ipsec.conf
[root@caandrei-vpn-01 ~]# nano /etc/ipsec.d/oci-ipsec.secrets
[root@caandrei-vpn-01 ~]#
[root@caandrei-vpn-01 ~]#
[root@caandrei-vpn-01 ~]# cat /etc/ipsec.d/oci-ipsec.secrets
152.67.130.140 140.204.35.3: PSK "zEheANYIvZNCHyAyByz3M54w3wW0c0Ya8wuyR8IFrLux7D
p11bXafq7Wa5VZuGE7"
152.67.130.140 140.204.35.14: PSK "UOb3h0Auha6aedq9ksbSnJINP194UYl0XMQs6VFDMNOnM
IrHiRSzgbO9KUSeJxaW"
[root@caandrei-vpn-01 ~]#

```

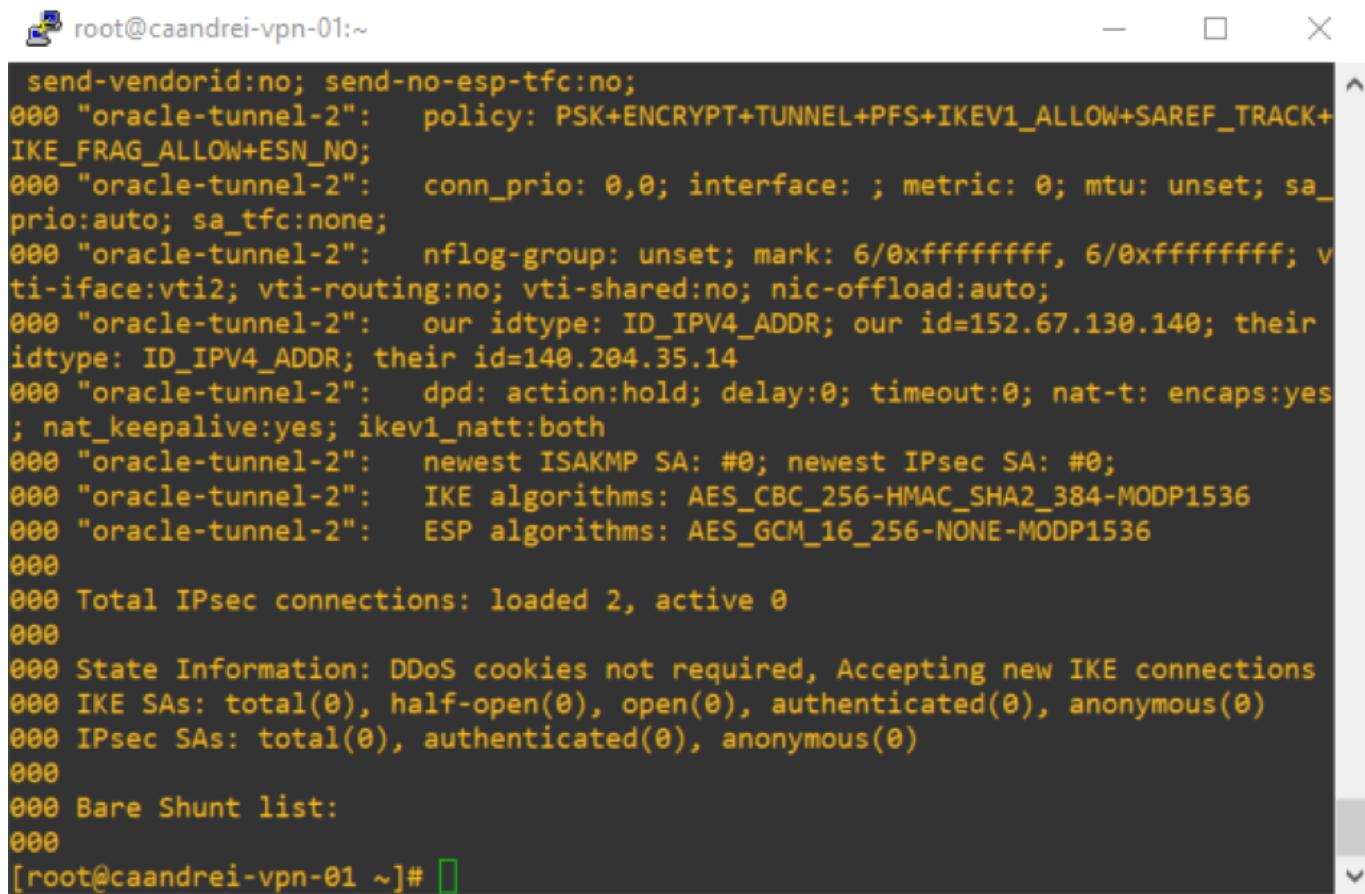
Start the ipsec service

```
sudo systemctl start ipsec
```

Check the status of the connection

```
ipsec status
```

You will notice that the tunnels are not up.



```
root@caandrei-vpn-01:~ send-vendorid:no; send-no-esp-tfc:no;
000 "oracle-tunnel-2":   policy: PSK+ENCRYPT+TUNNEL+PFS+IKEV1_ALLOW+SAREF_TRACK+
IKE_FRAG_ALLOW+ESN_NO;
000 "oracle-tunnel-2":   conn_prio: 0,0; interface: ; metric: 0; mtu: unset; sa_
prio:auto; sa_tfc:none;
000 "oracle-tunnel-2":   nflog-group: unset; mark: 6/0xffffffff, 6/0xffffffff; v
ti-iface:vti2; vti-routing:no; vti-shared:no; nic-offload:auto;
000 "oracle-tunnel-2":   our idtype: ID_IPV4_ADDR; our id=152.67.130.140; their
idtype: ID_IPV4_ADDR; their id=140.204.35.14
000 "oracle-tunnel-2":   dpd: action:hold; delay:0; timeout:0; nat-t: encaps:yes
; nat_keepalive:yes; ikev1_natt:both
000 "oracle-tunnel-2":   newest ISAKMP SA: #0; newest IPsec SA: #0;
000 "oracle-tunnel-2":   IKE algorithms: AES_CBC_256-HMAC_SHA2_384-MODP1536
000 "oracle-tunnel-2":   ESP algorithms: AES_GCM_16_256-NONE-MODP1536
000
000 Total IPsec connections: loaded 2, active 0
000
000 State Information: DDoS cookies not required, Accepting new IKE connections
000 IKE SAs: total(0), half-open(0), open(0), authenticated(0), anonymous(0)
000 IPsec SAs: total(0), authenticated(0), anonymous(0)
000
000 Bare Shunt list:
000
[root@caandrei-vpn-01 ~]#
```

Check the logs from the Libreswan

```
tail /var/log/secure | grep pluto
```

```
[root@caandrei-vpn-01 ~]# tail /var/log/secure | grep pluto
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: adding interface lo/lo 127.0.0.1:5
00
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: adding interface lo/lo 127.0.0.1:4
500
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: | setup callback for interface lo:
4500 fd 19
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: | setup callback for interface lo:
500 fd 18
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: | setup callback for interface ens
3:4500 fd 17
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: | setup callback for interface ens
3:500 fd 16
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: loading secrets from "/etc/ipsec.s
ecrets"
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: loading secrets from "/etc/ipsec.d
/oci-ipsec.secrets"
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: "oracle-tunnel-1": We cannot ident
ify ourselves with either end of this connection. 152.67.130.140 or 140.204.35.
3 are not usable
May 21 11:29:00 caandrei-vpn-01 pluto[30720]: "oracle-tunnel-2": We cannot ident
ify ourselves with either end of this connection. 152.67.130.140 or 140.204.35.
14 are not usable
[root@caandrei-vpn-01 ~]# 
```

The following line in the log will help identify the problem

```
oracle-tunnel-1": We cannot identify ourselves with either end of this
connection. 152.67.130.140 or 140.204.35.3 are not usable
```

This means that we did not fully configure the libreswan. On ens3 interface we do not have a public IP address, so we need to adjust the config and use as left the private ip address configured on the interface and for the leftid the public ip address. Please make sure you adjust both of the tunnels.

```
root@caandrei-vpn-01:~#
GNU nano 2.3.1          File: /etc/ipsec.d/oci-ipsec.conf          Modified ^

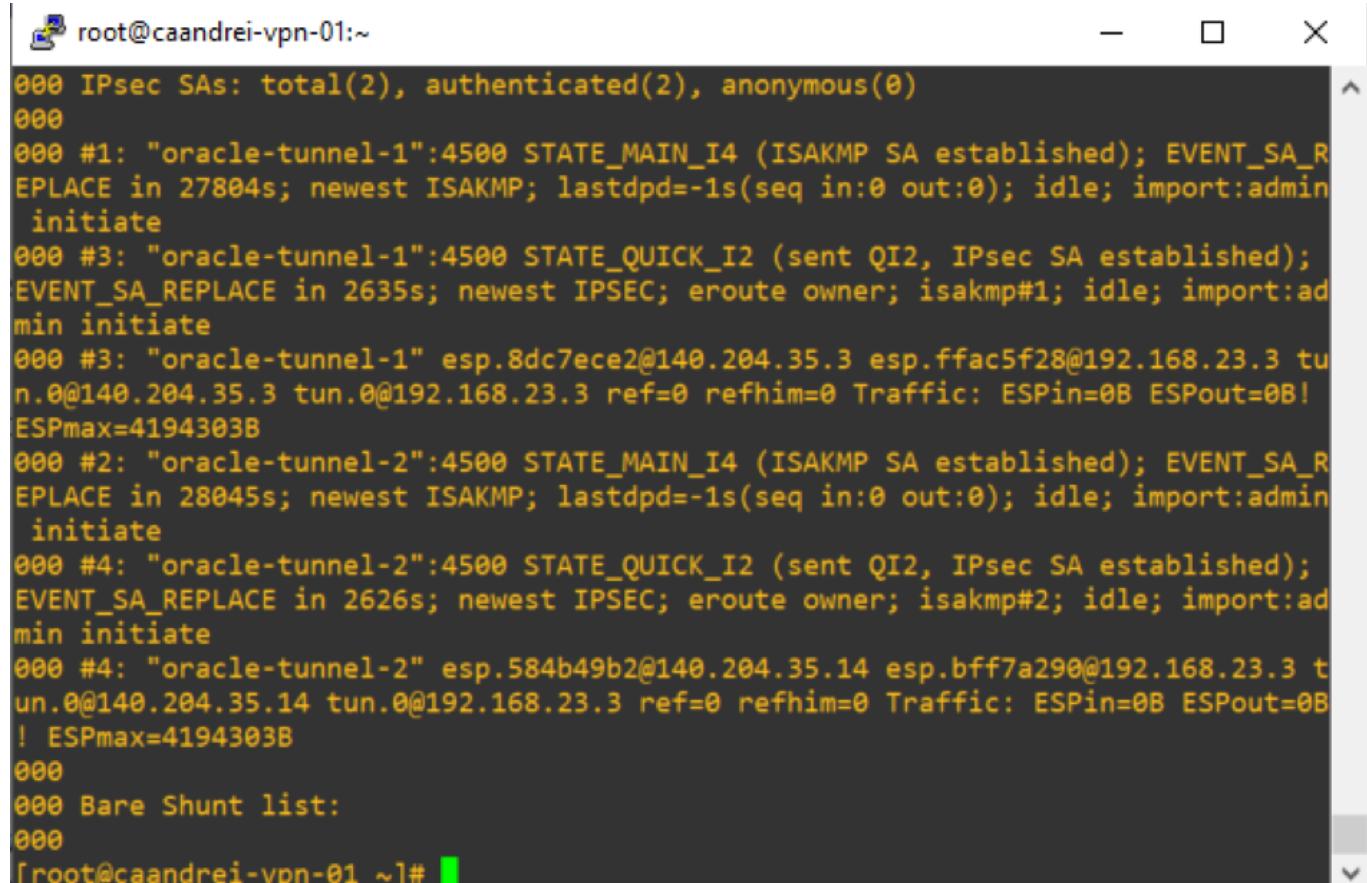
conn oracle-tunnel-1
left=192.168.23.3
leftid=152.67.130.140 # See preceding note about 1-1 NAT device
right=140.204.35.3
authby=secret
leftsubnet=0.0.0.0/0
rightsubnet=0.0.0.0/0
auto=start
mark=5/0xffffffff # Needs to be unique across all tunnels
vti-interface=vti1
    leftvti-10.10.10.1/30
```

Restart the ipsec service

```
sudo systemctl restart ipsec
```

Check the status of the service

```
ipsec status
```



A terminal window titled "root@caandrei-vpn-01:~" displaying the output of the "ipsec status" command. The output shows two IPsec SAs established: one for tunnel 1 and one for tunnel 2. Tunnel 1 is in STATE_MAIN_I4 and Tunnel 2 is also in STATE_MAIN_I4. Both tunnels have ISAKMP SAs established. The traffic for both tunnels is ESP traffic.

```
root@caandrei-vpn-01:~ 000 IPsec SAs: total(2), authenticated(2), anonymous(0)
000
000 #1: "oracle-tunnel-1":4500 STATE_MAIN_I4 (ISAKMP SA established); EVENT_SA_REPLACE in 27804s; newest ISAKMP; lastdpd=-1s(seq in:0 out:0); idle; import:admin initiate
000 #3: "oracle-tunnel-1":4500 STATE_QUICK_I2 (sent QI2, IPsec SA established); EVENT_SA_REPLACE in 2635s; newest IPSEC; eroute owner; isakmp#1; idle; import:admin initiate
000 #3: "oracle-tunnel-1" esp.8dc7ece2@140.204.35.3 esp.ffac5f28@192.168.23.3 tun.0@140.204.35.3 tun.0@192.168.23.3 ref=0 refhim=0 Traffic: ESPin=0B ESPout=0B! ESPmax=4194303B
000 #2: "oracle-tunnel-2":4500 STATE_MAIN_I4 (ISAKMP SA established); EVENT_SA_REPLACE in 28045s; newest ISAKMP; lastdpd=-1s(seq in:0 out:0); idle; import:admin initiate
000 #4: "oracle-tunnel-2":4500 STATE_QUICK_I2 (sent QI2, IPsec SA established); EVENT_SA_REPLACE in 2626s; newest IPSEC; eroute owner; isakmp#2; idle; import:admin initiate
000 #4: "oracle-tunnel-2" esp.584b49b2@140.204.35.14 esp.bff7a290@192.168.23.3 tun.0@140.204.35.14 tun.0@192.168.23.3 ref=0 refhim=0 Traffic: ESPin=0B ESPout=0B! ESPmax=4194303B
000
000 Bare Shunt list:
000
[root@caandrei-vpn-01 ~]#
```

You can observe that the tunnels are up (IPsec SA established).

Setup Firewall Rules

Adjust the NSG of the public VM:

Add Rule

[Help](#)

Add Security Rules

Optionally add one or more rules to the network security group. [Learn more about security rules.](#)

STATELESS [\(i\)](#)

DIRECTION [Ingress](#) SOURCE TYPE [CIDR](#) SOURCE CIDR [140.204.35.3/32](#)
Specified IP addresses: 140.204.35.3-140.204.35.3 (1 IP addresses)

IP PROTOCOL [UDP](#) SOURCE PORT RANGE [OPTIONAL](#) DESTINATION PORT RANGE [OPTIONAL](#)
All 500,4500

Allows: Allows UDP traffic 500,4500

DESCRIPTION [OPTIONAL](#)
Maximum 255 characters

[+ Another Rule](#)

[Add](#) [Cancel](#)

Add another rule for the second ipsec peer.

Add Rule

[Help](#)

Maximum 255 characters

STATELESS (i)

DIRECTION: Ingress

SOURCE TYPE: CIDR

SOURCE CIDR: 140.204.35.14/32
Specified IP addresses: 140.204.35.14-140.204.35.14 (1 IP addresses)

IP PROTOCOL: UDP

SOURCE PORT RANGE: OPTIONAL (i) All

DESTINATION PORT RANGE: OPTIONAL (i) 500,4500

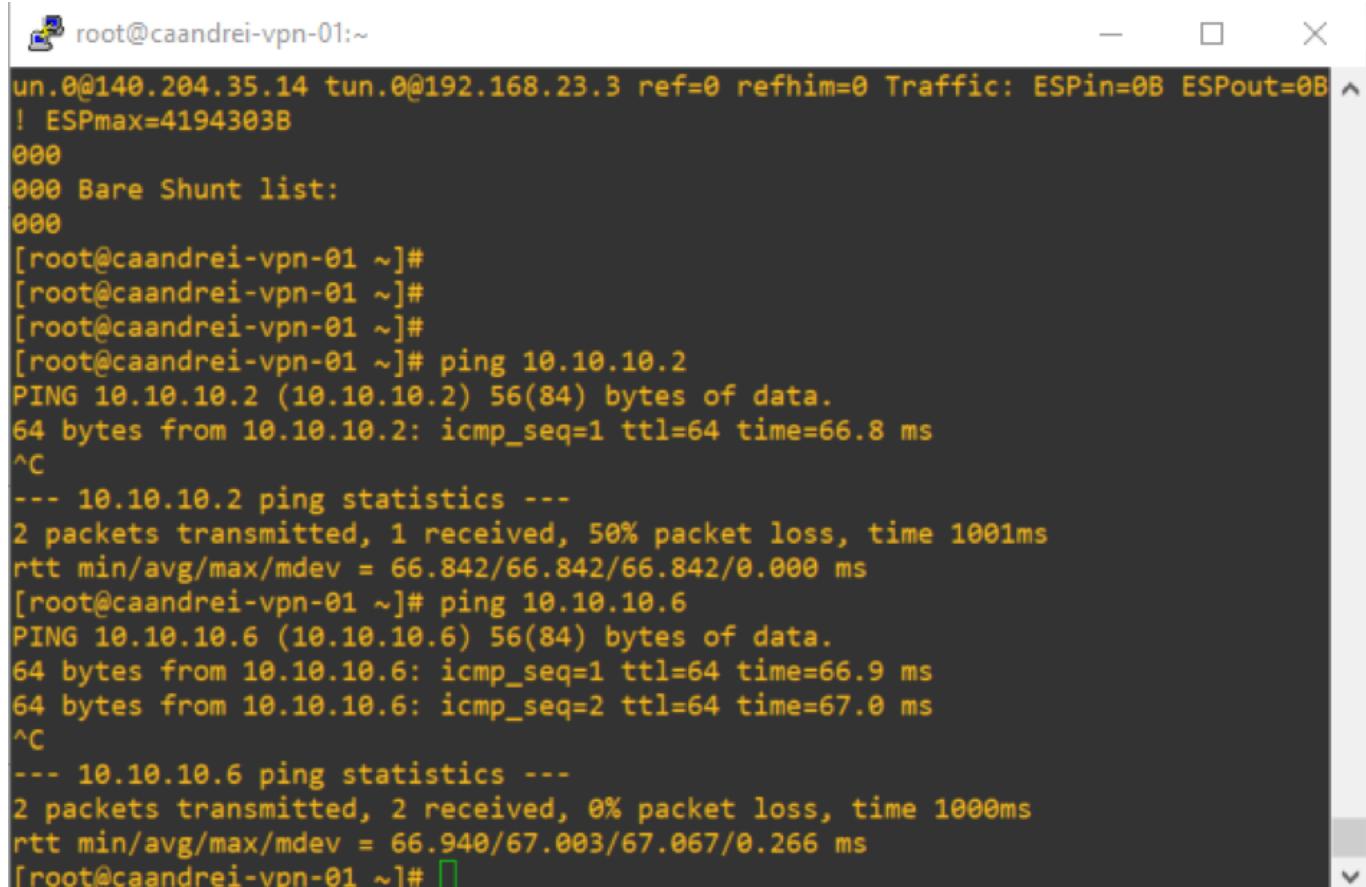
Allows: Allows UDP traffic 500,4500

DESCRIPTION: OPTIONAL
Maximum 255 characters

[+ Another Rule](#)

Add [Cancel](#)

Push the "Add" button. Test the connectivity over the ipsec tunnel by pinging the other side tunnel interface



```
root@caandrei-vpn-01:~  
un.0@140.204.35.14 tun.0@192.168.23.3 ref=0 refhim=0 Traffic: ESPin=0B ESPout=0B ^  
! ESPmax=4194303B  
000  
000 Bare Shunt list:  
000  
[root@caandrei-vpn-01 ~]#  
[root@caandrei-vpn-01 ~]#  
[root@caandrei-vpn-01 ~]#  
[root@caandrei-vpn-01 ~]# ping 10.10.10.2  
PING 10.10.10.2 (10.10.10.2) 56(84) bytes of data.  
64 bytes from 10.10.10.2: icmp_seq=1 ttl=64 time=66.8 ms  
^C  
--- 10.10.10.2 ping statistics ---  
2 packets transmitted, 1 received, 50% packet loss, time 1001ms  
rtt min/avg/max/mdev = 66.842/66.842/66.842/0.000 ms  
[root@caandrei-vpn-01 ~]# ping 10.10.10.6  
PING 10.10.10.6 (10.10.10.6) 56(84) bytes of data.  
64 bytes from 10.10.10.6: icmp_seq=1 ttl=64 time=66.9 ms  
64 bytes from 10.10.10.6: icmp_seq=2 ttl=64 time=67.0 ms  
^C  
--- 10.10.10.6 ping statistics ---  
2 packets transmitted, 2 received, 0% packet loss, time 1000ms  
rtt min/avg/max/mdev = 66.940/67.003/67.067/0.266 ms  
[root@caandrei-vpn-01 ~]#
```

Install Quagga

Check the Ipsec connection the on DRG. You will notice that the IPsec tunnel is up but the bgp connection is down.

We need to install a bgp service on the public VM.

```
sudo yum install quagga -y
```

Create the bgpd config file:

```
sudo touch /etc/quagga/bgpd.conf
```

Enable quagga daemons

```
systemctl start zebra  
systemctl enable zebra  
systemctl start bgpd  
systemctl enable bgpd
```

root@caandrei-vpn-01:~

Verifying : 1:net-snmp-5.7.2-48.el7_8.x86_64 5/5

Installed:

quagga.x86_64 0:0.99.22.4-5.el7_4

Dependency Installed:

net-snmp.x86_64 1:5.7.2-48.el7_8
net-snmp-agent-libs.x86_64 1:5.7.2-48.el7_8
net-snmp-libs.x86_64 1:5.7.2-48.el7_8
perl-Data-Dumper.x86_64 0:2.145-3.el7

Complete!

[root@caandrei-vpn-01 ~]# ls /etc/quagga/
vtysh.conf zebra.conf

[root@caandrei-vpn-01 ~]# sudo touch /etc/quagga/bgpd.conf

[root@caandrei-vpn-01 ~]# systemctl start zebra

[root@caandrei-vpn-01 ~]# systemctl enable zebra

Created symlink from /etc/systemd/system/multi-user.target.wants/zebra.service to /usr/lib/systemd/system/zebra.service.

[root@caandrei-vpn-01 ~]# systemctl start bgpd

[root@caandrei-vpn-01 ~]# systemctl enable bgpd

Created symlink from /etc/systemd/system/multi-user.target.wants/bgpd.service to /usr/lib/systemd/system/bgpd.service.

[root@caandrei-vpn-01 ~]#

Enter in the quagga console

```
vtysh
```

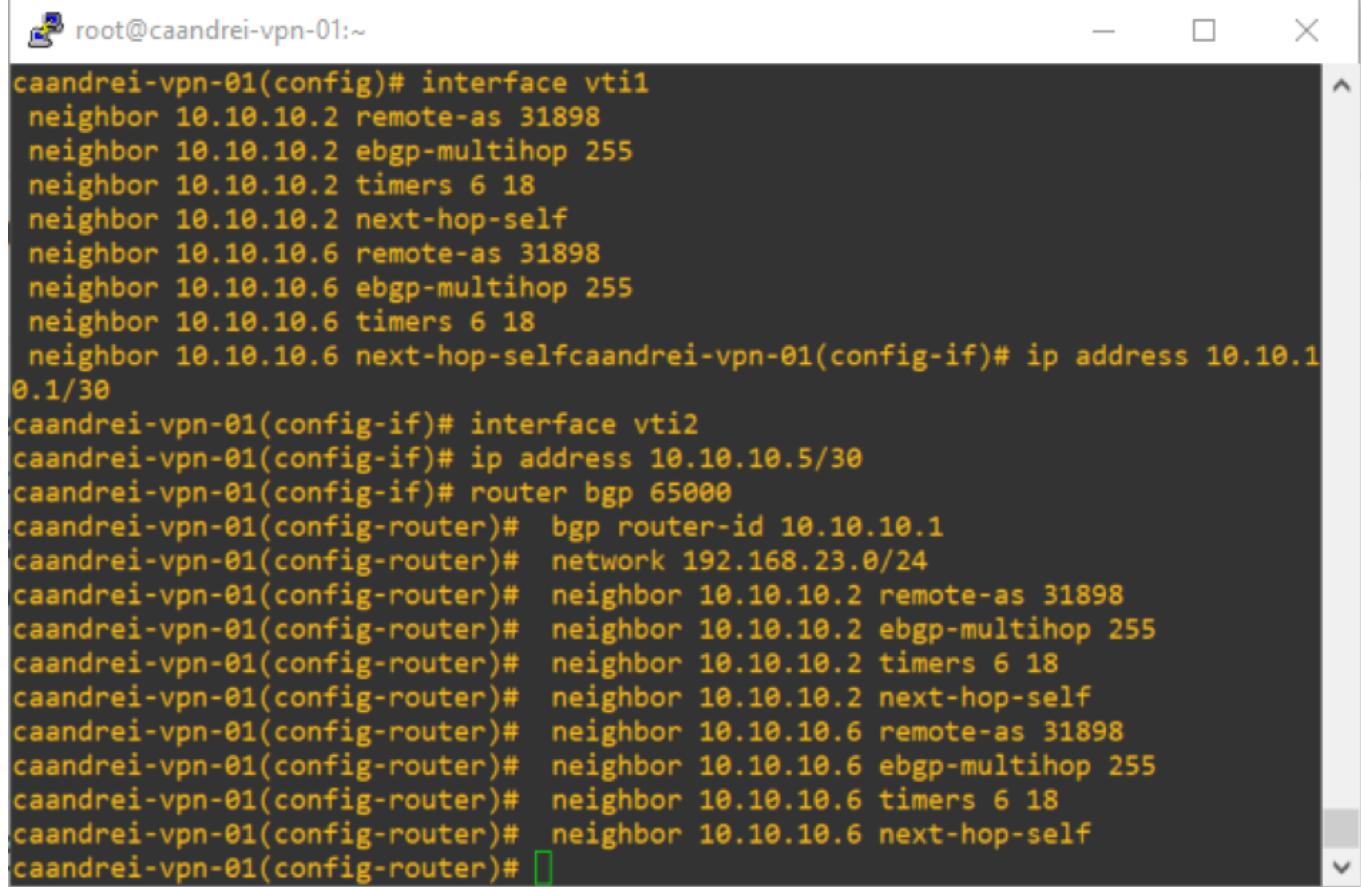
Enter in the configuration mode

```
conf t
```

Paste the following config:

```
interface vti1
ip address 10.10.10.1/30
interface vti2
ip address 10.10.10.5/30
router bgp 65000
bgp router-id 10.10.10.1
network 192.168.23.0/24
neighbor 10.10.10.2 remote-as 31898
neighbor 10.10.10.2 ebgp-multipath 255
neighbor 10.10.10.2 timers 6 18
neighbor 10.10.10.2 next-hop-self
neighbor 10.10.10.6 remote-as 31898
neighbor 10.10.10.6 ebgp-multipath 255
```

```
neighbor 10.10.10.6 timers 6 18  
neighbor 10.10.10.6 next-hop-self
```



A terminal window titled 'root@caandrei-vpn-01:~' showing the configuration of a Cisco router. The configuration includes multiple neighbor statements for BGP connections, interface configurations for vti1 and vti2, and a global BGP configuration with a router ID of 10.10.10.1.

```
root@caandrei-vpn-01:~  
caandrei-vpn-01(config)# interface vti1  
neighbor 10.10.10.2 remote-as 31898  
neighbor 10.10.10.2 ebgp-multipath 255  
neighbor 10.10.10.2 timers 6 18  
neighbor 10.10.10.2 next-hop-self  
neighbor 10.10.10.6 remote-as 31898  
neighbor 10.10.10.6 ebgp-multipath 255  
neighbor 10.10.10.6 timers 6 18  
neighbor 10.10.10.6 next-hop-self  
caandrei-vpn-01(config-if)# ip address 10.10.10.1 0.1/30  
caandrei-vpn-01(config-if)# interface vti2  
caandrei-vpn-01(config-if)# ip address 10.10.10.5/30  
caandrei-vpn-01(config-if)# router bgp 65000  
caandrei-vpn-01(config-router)# bgp router-id 10.10.10.1  
caandrei-vpn-01(config-router)# network 192.168.23.0/24  
caandrei-vpn-01(config-router)# neighbor 10.10.10.2 remote-as 31898  
caandrei-vpn-01(config-router)# neighbor 10.10.10.2 ebgp-multipath 255  
caandrei-vpn-01(config-router)# neighbor 10.10.10.2 timers 6 18  
caandrei-vpn-01(config-router)# neighbor 10.10.10.2 next-hop-self  
caandrei-vpn-01(config-router)# neighbor 10.10.10.6 remote-as 31898  
caandrei-vpn-01(config-router)# neighbor 10.10.10.6 ebgp-multipath 255  
caandrei-vpn-01(config-router)# neighbor 10.10.10.6 timers 6 18  
caandrei-vpn-01(config-router)# neighbor 10.10.10.6 next-hop-self  
caandrei-vpn-01(config-router)#[ ]
```

Issue CTRL+Z to exit the config mode. Check the configuration that you just pasted with

```
sh run
```

```
root@caandrei-vpn-01:~  
 ipv6 nd suppress-ra  
!  
interface vti2  
 ip address 10.10.10.5/30  
 ipv6 nd suppress-ra  
!  
router bgp 65000  
 bgp router-id 10.10.10.1  
 network 192.168.23.0/24  
 neighbor 10.10.10.2 remote-as 31898  
 neighbor 10.10.10.2 ebgp-multipath 255  
 neighbor 10.10.10.2 timers 6 18  
 neighbor 10.10.10.2 next-hop-self  
 neighbor 10.10.10.6 remote-as 31898  
 neighbor 10.10.10.6 ebgp-multipath 255  
 neighbor 10.10.10.6 timers 6 18  
 neighbor 10.10.10.6 next-hop-self  
!  
ip forwarding  
!  
line vty  
!  
end  
caandrei-vpn-01#
```

Issue `wr` command to save the config

```
root@caandrei-vpn-01:~  
!  
router bgp 65000  
 bgp router-id 10.10.10.1  
 network 192.168.23.0/24  
 neighbor 10.10.10.2 remote-as 31898  
 neighbor 10.10.10.2 ebgp-multipath 255  
 neighbor 10.10.10.2 timers 6 18  
 neighbor 10.10.10.2 next-hop-self  
 neighbor 10.10.10.6 remote-as 31898  
 neighbor 10.10.10.6 ebgp-multipath 255  
 neighbor 10.10.10.6 timers 6 18  
 neighbor 10.10.10.6 next-hop-self  
!  
ip forwarding  
!  
line vty  
!  
end  
caandrei-vpn-01# wr  
Building Configuration...  
Can't open configuration file /etc/quagga/zebra.conf.MGDT22.  
Can't open configuration file /etc/quagga/bgpd.conf.9fJM12.  
[OK]  
caandrei-vpn-01#
```

You will notice that the config file can't be written. This can be fix by following the instruction [here](#)

```
root@caandrei-vpn-01:~  
!  
ip forwarding  
!  
line vty  
!  
end  
caandrei-vpn-01# wr  
Building Configuration...  
Can't open configuration file /etc/quagga/zebra.conf.MGDT22.  
Can't open configuration file /etc/quagga/bgpd.conf.9fJM12.  
[OK]  
caandrei-vpn-01# getsebool zebra_write_config  
% Unknown command.  
caandrei-vpn-01# exit  
[root@caandrei-vpn-01 ~]# getsebool zebra_write_config  
zebra_write_config --> off  
[root@caandrei-vpn-01 ~]# setsebool zebra_write_config  
  
Usage: setsebool [ -NPV ] boolean value | bool1=val1 bool2=val2...  
  
[root@caandrei-vpn-01 ~]# setsebool zebra_write_config=on  
[root@caandrei-vpn-01 ~]# getsebool zebra_write_config  
zebra_write_config --> on  
[root@caandrei-vpn-01 ~]#
```

Now return to *vtysh* and save the config

```
root@caandrei-vpn-01:~  
zebra_write_config --> off  
[root@caandrei-vpn-01 ~]# setsebool zebra_write_config  
  
Usage: setsebool [ -NPV ] boolean value | bool1=val1 bool2=val2...  
  
[root@caandrei-vpn-01 ~]# setsebool zebra_write_config=on  
[root@caandrei-vpn-01 ~]# getsebool zebra_write_config  
zebra_write_config --> on  
[root@caandrei-vpn-01 ~]# vtysh  
  
Hello, this is Quagga (version 0.99.22.4).  
Copyright 1996-2005 Kunihiro Ishiguro, et al.  
  
caandrei-vpn-01# wr  
Building Configuration...  
Configuration saved to /etc/quagga/zebra.conf  
Can't backup old configuration file /etc/quagga/bgpd.conf.sav.  
[OK]
```

Tests

Check the BGP status

```
show ip bgp sum
```

```
caandrei-vpn-01# sh ip bgp summ
BGP router identifier 10.10.10.1, local AS number 65000
RIB entries 3, using 336 bytes of memory
Peers 2, using 9120 bytes of memory

Neighbor      V      AS MsgRcvd MsgSent    TblVer  InQ OutQ Up/Down  State/PfxRcd
10.10.10.2    4 31898       72       72        0     0     0 00:06:38          1
10.10.10.6    4 31898       73       71        0     0     0 00:06:39          1

Total number of neighbors 2
```

Display the routes learn by bgp

```
sh ip route bgp
```

```
caandrei-vpn-01# sh ip route bgp
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

B>* 192.168.24.0/24 [20/0] via 10.10.10.6, vti2, 00:07:46
```

Check the status of the IPsec connection on the DRG

Networking » [IPSec Connections](#) » VPN-caandrei-IPSec-to-Libreswan



VPN-caandrei-IPSec-to-Libreswan

[Edit](#) [Add Tags](#) [Open CPE Configuration Helper](#) [Terminate](#)

IPSec Connection Information	Tags
Static Route CIDR: — i Created: Thu, May 21, 2020, 10:24:53 UTC OCID: ...dm2yjq Show Copy	DRG: DRG-caandrei-IPSec-to-Libreswan CPE: CPE-caandrei-IPSec-to-Libreswan CPE IKE Identifier Type: IP Address CPE IKE Identifier: 152.67.130.140

Resources

Tunnels in caandrei Compartment

Name	Lifecycle State i	IPSec Status i	BGP Status i	Oracle VPN IP Address	Routing Type
Tunnel1	● Available	● Up	● Up	140.204.35.3	BGP Dynamic Routing
Tunnel2	● Available	● Up	● Up	140.204.35.14	BGP Dynamic Routing

Showing 2 Items

Both bgp sessions are UP!

