

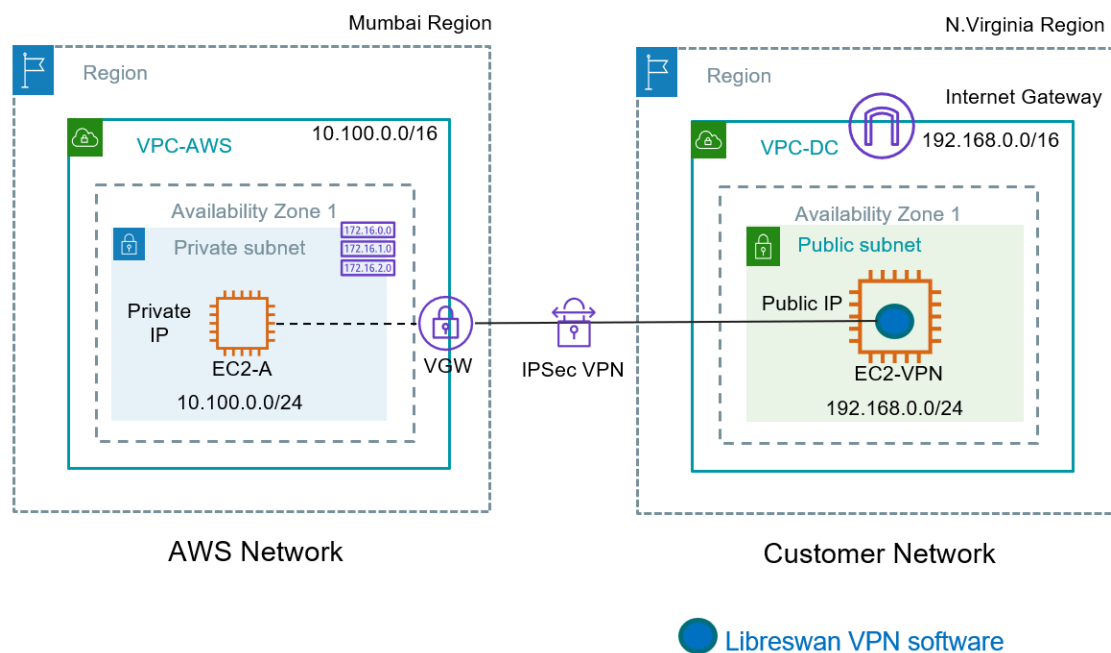


# AWS Site-to-Site VPN setup guide

By Chetan Agrawal

<https://www.awswithchetan.com>

## Architecture



## Steps

1. Create VPC-AWS (10.0.0.0/16) in Mumbai region and VPC-DC (192.168.0.0/16) in N. Virginia region
  - a. Create one Private subnet in VPC-AWS, corresponding route table and associate with the subnet
  - b. Create IGW for VPC-DC, associate it with the VPC-DC. Create one Public subnet in VPC-DC, corresponding route table and associate route table with the subnet
2. Launch EC2 instances in both the VPCs
  - a. In VPC-AWS, instance will have only Private IP. Security group to allow All ICMP IPv4 from VPC-DC CIDR (192.168.0.0/16)

- b. In VPC-DC, launch EC2 instance (EC2-VPN) with **Amazon Linux 2023 AMI**.
  - c. This instance should have Public IP as well. Security group to allow All ICMP – IPv4 for VPC-AWS CIDR and SSH from MyIP or 0.0.0.0/0
  - d. For EC2-VPN, go to Actions -> Networking -> Source/Destination Check -> Stop
3. Create virtual private gateway and associate with VPC-AWS
  - a. In Mumbai region, go to VPC console -> Left panel -> Virtual Private Gateways -> Create virtual private gateway (Use default ASN)
  - b. Name: VPC-AWS-VGW -> Create virtual private gateway
  - c. Select VGW -> Actions -> Attach to VPC -> Select VPC-AWS from the dropdown -> Attach to VPC-AWS
  - d. Wait for the attachment to complete
  - e. Modify VPC-AWS Private subnet route table and add route for 0.0.0.0/0 with target as Virtual Private Gateway (vgw-xxxxx)
  - f. Note down or copy the Public IP of EC2-VPN instance in N. Virginia region.
4. Create VPN connection
  - a. VPC console -> Left panel -> Site-to-Site VPN Connections -> Create VPN connection
  - b. Name: AWS-DC-VPN, Target gateway type: Virtual Private Gateway, Select VGW from the dropdown
  - c. Customer Gateway: New, Enter Public IP address of EC2-VPN EC2 instance
  - d. Routing options: Static, Static IP prefixes: 192.168.0.0/16
  - e. Local IPv4 network CIDR – 192.168.0.0/16
  - f. Remote IPv4 network CIDR – 10.0.0.0/16
  - g. Create VPN connection and wait for the connection to complete
5. Download the configuration file
  - a. Select VPN connection you created above -> Download configuration -> Select vendor as **Openswan** -> Download
  - b. Save the configuration file on your local machine and open in Notepad
6. Install and configure DC VPN server
  - a. SSH into EC2-VPN from your workstation using PuTTY or any SSH client
  - b. Add Libreswan repository. For this create /etc/yum.repos.d/fedora.repo and add fedora repository.

Note: If you want, you can avoid using sudo in every command by changing current ec2-user to root using *sudo su* command (optional)

```
sudo vi /etc/yum.repos.d/fedora.repo
```

- Copy following content to clipboard using CTRL+C
- Press i
- Right click to paste
- Press ESC
- :wq

```
[fedora]
name=Fedora 36 - $basearch
#baseurl=http://download.example/pub/fedora/linux/releases/36/Everything/$basearch/os/
metalink=https://mirrors.fedoraproject.org/metalink?repo=fedora-36&arch=$basearch
enabled=0
countme=1
metadata_expire=7d
repo_gpgcheck=0
type=rpm
gpgcheck=1
gpgkey=https://getfedora.org/static/fedora.gpg
skip_if_unavailable=False
```

c. Install Libreswan

```
sudo dnf --enablerepo=fedora install libreswan -y
```

d. Open the downloaded VPN server configuration file and follow the instructions. Instructions in this file should be like the following steps e. through step i.:

e. Open `/etc/sysctl.conf` and ensure that its values match the following:

```
net.ipv4.ip_forward = 1
net.ipv4.conf.default.rp_filter = 0
net.ipv4.conf.default.accept_source_route = 0
```

- f. Apply the changes in step 1 by executing the command `'sysctl -p'`
- g. Open `/etc/ipsec.conf` and look for the line below. Ensure that the `#` in front of the line has been removed, then save and exit the file.
- i. `#include /etc/ipsec.d/*.conf`
- h. Create a new file at `/etc/ipsec.d/aws.conf` if doesn't already exist, and then open it. Append the following configuration to the end in the file:

- `#leftsubnet=` is the local network behind your openswan server, and you will need to replace the `<LOCAL NETWORK>` below with this value (don't include the brackets). If you have multiple subnets, you can use `0.0.0.0/0` instead.
- `#rightsubnet=` is the remote network on the other side of your VPN tunnel that you wish to have connectivity with, and you will need to replace `<REMOTE NETWORK>` with this value (don't include brackets).

▪ **Remove `auth=esp`**

▪ **`phase2alg=aes_gcm`**

▪ **`ike=aes256-sha1`**

Text in red will be different values as per your environment. Make sure to check and replace as necessary.

```
conn Tunnel1
  authby=secret
  auto=start
  left=%defaultroute
  leftid=184.73.51.101
  right=13.232.7.88
  type=tunnel
  ikelifetime=8h
  keylife=1h
  phase2alg=aes_gcm
  ike=aes256-sha1
  keyingtries=%forever
  keyexchange=ike
  leftsubnet=192.168.0.0/16
  rightsubnet=10.0.0.0/16
  dpddelay=10
  dpdtimeout=30
  dpdaction=restart_by_peer
  encapsulation=yes
```

- i. Create a new file at /etc/ipsec.d/aws.secrets if it doesn't already exist, and append this line to the file (be mindful of the spacing!):

```
184.73.51.101 13.232.7.88: PSK "BbO4GaeAaQbrk3.Z_moK325TOE.ovaZk"
```

- j. Start ipsec service

```
sudo systemctl start ipsec.service
```

- k. Check status of the ipsec service

```
sudo systemctl status ipsec.service
```

7. Check the connectivity from EC2-VPN to EC2-A. Ping should be successful.

```
ping 10.0.0.x
PING 10.0.0.167 (10.0.0.167) 56(84) bytes of data.
64 bytes from 10.0.0.167: icmp_seq=1 ttl=127 time=187 ms
64 bytes from 10.0.0.167: icmp_seq=2 ttl=127 time=186 ms
```

## Cleanup:

Afterful successful VPN connectivity, delete all the resources that you created during this lab

- Delete VPN Connection in (Mumbai region)
- Delete Customer Gateway (Mumbai region)
- Delete Virtual Private Gateway (Mumbai region)
- Terminate both EC2 instances (both the regions)

- Delete both VPCs (both the regions)

### Troubleshooting:

1. Check the VPN tunnel status by Mumbai region VPN console -> VPN Connections -> Select your connection -> Tunnel details
  - a. Tunnel 1 status should be UP. Tunnel 2 will be down as we haven't configured it.

Tunnel number ▾	Outside IP address ▾	Inside IPv4 CIDR ▾	Inside IPv6 CIDR ▾	Status ▾
Tunnel 1	13.232.7.88	169.254.41.204/30	–	🟢 Up
Tunnel 2	35.154.79.214	169.254.239.84/30	–	🔴 Down

2. If Tunnel 1 isn't UP
  - a. Make sure the VPN configuration file is correct. Specially check instructions highlighted in yellow above.
  - b. Enable the VPN logs by editing `/etc/ipsec.conf`
    - i. Remove # from this line to make it: `logfile=/var/log/pluto.log`
    - ii. Remove # from this line to make it: `plutodebug="base"`
    - iii. Save the file
    - iv. Start ipsec service and check these logs
  - c. Make sure that `aws.secrets` file is exactly same as given in the VPN configuration file downloaded from AWS.
  - d. After any changes restart ipsec service: `sudo systemctl restart ipsec.service`
3. If Tunnel is UP but ping is not working:
  - a. Make sure EC2-A Security group allows ICMP IPv4 All traffic from VPC-DC CIDR
  - b. Make sure VPC-AWS Private subnet route table has route for Destination VPC-DC CIDR with target as VGW