AWS site-to-site VPN using VTI and BGP to update dynamic routing

Srividya Anantapatnaikuni - 2021-08-20 - 0 Comments - in IPSec

Introduction:

Our goal is to create a VPN tunnel between private networks in AWS service providers using VyOS routers.

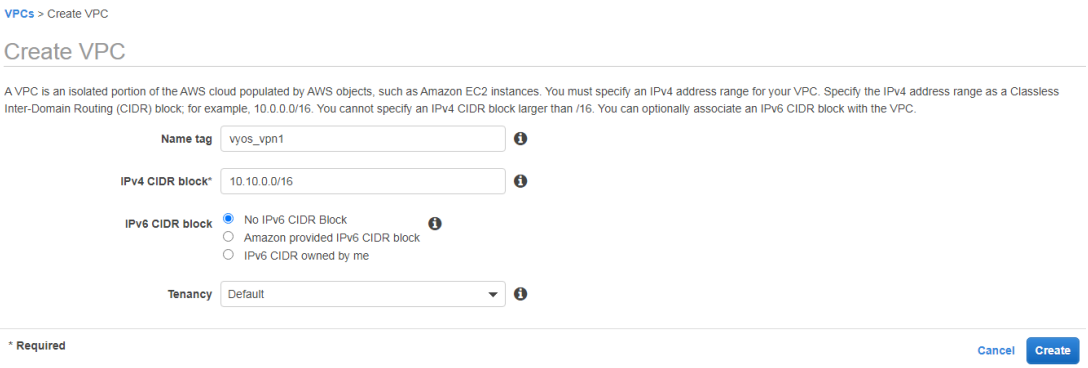
We will use VyOS routers on both sites with VTI interfaces, IPSec encryption and BGP for dynamic routing. In the AWS, we have a private subnet with `10.10.0.0/24` and another in ESXi host with `192.168.254.0/24`. Public IP addresses of AWS is `18.222.200.181` and ESXi host is `185.144.208.250`.

AWS:

.

First of all, we have to perform a network part in AWS, which is called Virtual Private Cloud(VPC).

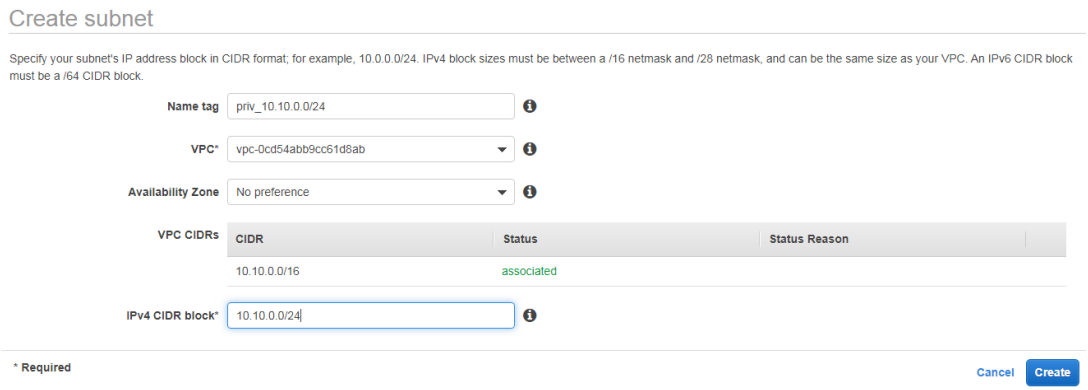
Go to `Your VPCs`, select `Create VPC` and fill out the required fields and select `Create`.



We will use a `10.10.0.0/16` network in our VPC and `vyos-vpn1` as name.

When network creation is finished, we are going to create two subnets. First subnet `10.10.0.0/24` will be used for private networks and second subnet `10.10.1.0/24` will be used for public networks. We will attach interfaces of VyOS router to these subnets.

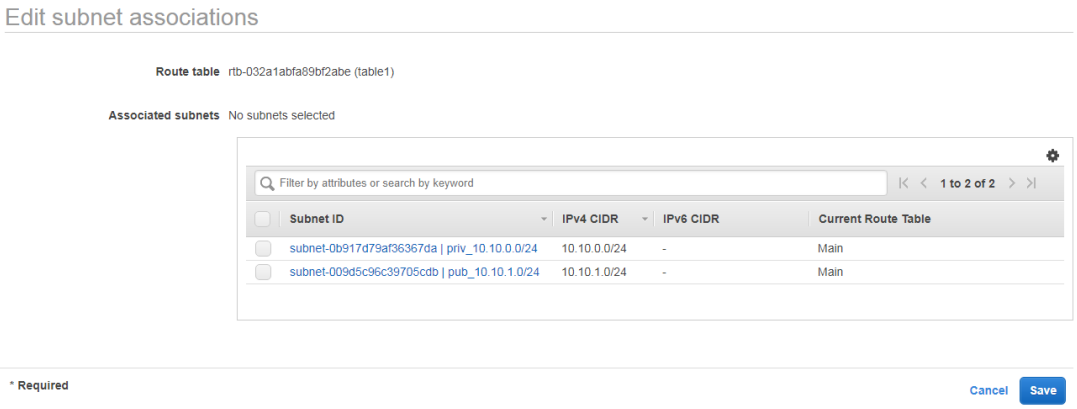
To create a Private subnet on the VPC page, go to the subnets link and click on the `Create Subnet` button. Fill Name tag field and choose subnet from configured IPv4 CIDR block before. We will call the subnet `priv_10.10.0.0/24` and choose the appropriate prefix `10.10.0.0/24`. From the drop-down list select VPC `vyos_vpn1` and click, `Create`.



Follow the same steps to create the public subnet.

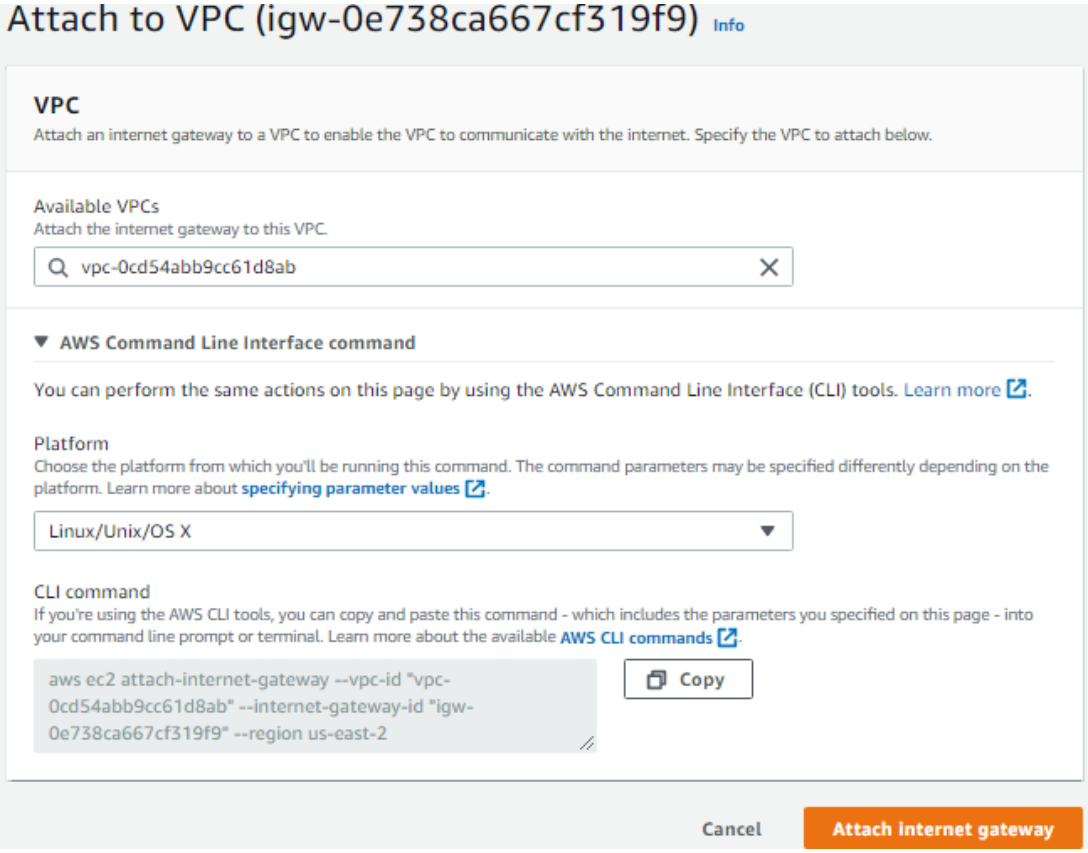
In the left column on the VPC page, select the `Route Tables` option and then the `Create Route Table` button. Create two route tables named as `Public_RouteTable` and `Private_RouteTable` and choose `vyos_vpn1` from the VPC drop-down list.

Subnets have to be associated with the routing table. Select the created routing tables and click `Actions`, from the drop down menu, select `Edit subnet Association` and add the subnets.



To have internet reachability from our public subnet we need to create an `internet gateway` and attach it to VPC. Go to `Internet Gateways` and click on `Create Internet Gateway`. Fill the `Name tag` field with `my-vyos-internet-gw` and click the `Create` button.

Next, click on `Actions` and choose `Attach VPC` from the drop-down list. Or you can attach the VPC from the highlighted notification.



When IGW is attached to VPC we should point the default route in Public_RouteTable. In the left column click on `Route Tables` and select `Public_RouteTable`. Then go to `Routes`, and click on `Edit` button. Select `Add another route` and add `0.0.0.0` in the Destination field, click on Target field and select gateway `my-vyos-internet-gw` from drop-down list and select `Attach Internet gateway`.

Now, we need to create a VyOS instance in AWS. Refer this guide for the installation steps.

While configuring the instance, from the `Network` drop-down list select VPC `vyos_vpn1` and from `Subnet` drop-down list select public subnet `Public_10.10.1.0/24`.

Also we need to add a second network interface for VyOS router. Scroll down to `Network interfaces` and click `Add Device`. An additional interface eth1 will appear, from the `Subnet` drop-down list select our private subnet. Next go to the `6. Configure Security Group` on the top of the page.

Just for demo purposes we will allow all traffic to the public interface of our VyOS instance. Select `Create a new security group` radio button and give it a name, we will call it

Permit_Any-SecGr. In the Type field select All traffic, write 0.0.0.0/0 in the Source field. Take note, that this ACL allows you to communicate any instance on the internet with your router, never use such ACLs in production. Then select the Review and Launch. After launching the instance, it will ask to select the SSH key pair.

In the left column, select Network Interfaces and add vyos_public and vyos_private in the name fields of the interfaces.

To assign public IP-address to VyOS public interface, go to Elastic IPs and click Allocate new address button, in the appeared page click on Allocate.

Next step is to associate that allocated address with VyOS public interface.Go back to Network Interfaces option, select vyos_public and choose Associate address from Actions drop-down menu.

Now we need to disable source and destination checking on VyOS interfaces. Each EC2 instance performs source/destination checks by default. This means that the instance must be the source or destination of any traffic it sends or receives, but most of the time routers are only intermediate devices, not sources or destinations. Go to Network Interfaces, check vyos_public interface and from the context menu click on Change Source/Dest. Check. In a pop-up window check radio button Source/dest. Check:Disabled

Perform the same steps for vyos_private to uncheck the Change Source/Dest. Check

At this point we have installed a VyOS instance, configured subnets and public route table. Next step is to make VyOS a default gateway for the private subnet. Go to Services:VPC. In the left column click on Route Tables and select Private_RouteTable. Then click on Routes and add route destination 0.0.0.0/0, in the Target field type vyos_private and click Save routes.

Now traffic of all AWS instances which are running in vyos_private subnet and have configured default gateway to VyOS private interface IP-address will go through VyOS router.

Let’s connect through SSH to VyOS and configure interfaces and routing. Interface eth0 is auto-configured by AWS, so we have to configure only eth1.

```
vyos@ip-10-10-1-226# set interfaces ethernet eth1 address
10.10.0.226/24

[edit]

vyos@ip-10-10-1-226# commit
```

```
[edit]

vyos@ip-10-10-1-226# save
Saving configuration to '/config/config.boot'...
Done
```

VyOS config part:

Configure a VTI with a dummy IP address

```
set interfaces vti vti0 address '10.100.100.2/30'
```

IPSec Phase I configuration

```
set vpn ipsec ike-group vyos-ike-aws dead-peer-detection action
'clear'
set vpn ipsec ike-group vyos-ike-aws dead-peer-detection interval
'30'
set vpn ipsec ike-group vyos-ike-aws dead-peer-detection timeout '90'
set vpn ipsec ike-group vyos-ike-aws ikev2-reauth 'no'
set vpn ipsec ike-group vyos-ike-aws key-exchange 'ikev1'
set vpn ipsec ike-group vyos-ike-aws lifetime '3600'
set vpn ipsec ike-group vyos-ike-aws proposal 1 dh-group '2'
set vpn ipsec ike-group vyos-ike-aws proposal 1 encryption 'aes256'
set vpn ipsec ike-group vyos-ike-aws proposal 1 hash 'sha256'
```

IPSec configuration.Phase 2

```
set vpn ipsec esp-group vyos-esp-aws compression 'disable'
set vpn ipsec esp-group vyos-esp-aws lifetime '1800'
set vpn ipsec esp-group vyos-esp-aws mode 'tunnel'
set vpn ipsec esp-group vyos-esp-aws pfs 'dh-group2'
set vpn ipsec esp-group vyos-esp-aws proposal 1 encryption 'aes256'
set vpn ipsec esp-group vyos-esp-aws proposal 1 hash 'sha256'
```

Enable IPsec on eth0 and peer configuration

```
set vpn ipsec ipsec-interfaces interface 'eth0'
set vpn ipsec site-to-site peer 185.144.208.250 authentication id
'18.222.200.181'
set vpn ipsec site-to-site peer 185.144.208.250 authentication mode
```

```
'pre-shared-secret'
set vpn ipsec site-to-site peer 185.144.208.250 authentication pre-
shared-secret 'T0pSecr3tP@ss'
set vpn ipsec site-to-site peer 185.144.208.250 connection-type
'initiate'
set vpn ipsec site-to-site peer 185.144.208.250 default-esp-group
'vyos-esp-aws'
set vpn ipsec site-to-site peer 185.144.208.250 ike-group 'vyos-ike-
aws'
set vpn ipsec site-to-site peer 185.144.208.250 ikev2-reauth
'inherit'
set vpn ipsec site-to-site peer 185.144.208.250 local-address
'10.10.1.226'
set vpn ipsec site-to-site peer 185.144.208.250 vti bind 'vti0'
set vpn ipsec site-to-site peer 185.144.208.250 vti esp-group 'vyos-
esp-aws'
```

Clamp the VTI's MSS to 1394 to avoid PMTU blackholes and also set MTU 1436 to VTI

```
set firewall options interface vti0 adjust-mss 1394
set interfaces vti vti0 mtu 1436
```

Configure your BGP Settings

```
set protocols bgp 65002 address-family ipv4-unicast network
10.10.0.0/24
set protocols bgp 65002 neighbor 10.100.100.1 address-family ipv4-
unicast soft-reconfiguration inbound
set protocols bgp 65002 neighbor 10.100.100.1 disable-connected-check
set protocols bgp 65002 neighbor 10.100.100.1 remote-as '65001'
set protocols bgp 65002 neighbor 10.100.100.1 timers holdtime '30'
set protocols bgp 65002 neighbor 10.100.100.1 timers keepalive '10'
```

Add an interface route to reach BGP listener

```
set protocols static interface-route 10.100.100.0/30 next-hop-
interface vti0
```

Vyos Configuration on the ESXi host:

Configure a VTI with a dummy IP address

```
set interfaces vti vti0 address '10.100.100.1/30'
```

IPSec Configuration Phase 1

```
set vpn ipsec ike-group vyos-ike-onprem dead-peer-detection action
'clear'
set vpn ipsec ike-group vyos-ike-onprem dead-peer-detection interval
'30'
set vpn ipsec ike-group vyos-ike-onprem dead-peer-detection timeout
'90'
set vpn ipsec ike-group vyos-ike-onprem ikev2-reauth 'no'
set vpn ipsec ike-group vyos-ike-onprem key-exchange 'ikev1'
set vpn ipsec ike-group vyos-ike-onprem lifetime '3600'
set vpn ipsec ike-group vyos-ike-onprem proposal 1 dh-group '2'
set vpn ipsec ike-group vyos-ike-onprem proposal 1 encryption
'aes256'
set vpn ipsec ike-group vyos-ike-onprem proposal 1 hash 'sha256'
```

#IPSec Configuration Phase 2

```
set vpn ipsec esp-group vyos-esp-onprem compression 'disable'
set vpn ipsec esp-group vyos-esp-onprem lifetime '1800'
set vpn ipsec esp-group vyos-esp-onprem mode 'tunnel'
set vpn ipsec esp-group vyos-esp-onprem pfs 'dh-group2'
set vpn ipsec esp-group vyos-esp-onprem proposal 1 encryption
'aes256'
set vpn ipsec esp-group vyos-esp-onprem proposal 1 hash 'sha256'
```

Enable IPsec on eth0 and peer configuration

```
set vpn ipsec ipsec-interfaces interface 'eth0'
set vpn ipsec site-to-site peer 18.222.200.181 authentication id
'185.144.208.250'
set vpn ipsec site-to-site peer 18.222.200.181 authentication mode
'pre-shared-secret'
set vpn ipsec site-to-site peer 18.222.200.181 authentication pre-
shared-secret 'T0pSecr3tP@ss'
```

```
set vpn ipsec site-to-site peer 18.222.200.181 connection-type
'initiate'
set vpn ipsec site-to-site peer 18.222.200.181 default-esp-group
'vyos-esp-onprem'
set vpn ipsec site-to-site peer 18.222.200.181 ike-group 'vyos-ike-
onprem'
set vpn ipsec site-to-site peer 18.222.200.181 ikev2-reauth 'inherit'
set vpn ipsec site-to-site peer 18.222.200.181 local-address
'192.168.255.88'
set vpn ipsec site-to-site peer 18.222.200.181 vti bind 'vti0'
set vpn ipsec site-to-site peer 18.222.200.181 vti esp-group 'vyos-
esp-onprem'
```

BGP configuration:

```
set protocols bgp 65001 address-family ipv4-unicast network
192.168.254.0/24
set protocols bgp 65001 neighbor 10.100.100.2 address-family ipv4-
unicast soft-reconfiguration inbound
set protocols bgp 65001 neighbor 10.100.100.2 disable-connected-check
set protocols bgp 65001 neighbor 10.100.100.2 remote-as '65002'
set protocols bgp 65001 neighbor 10.100.100.2 timers holdtime '30'
set protocols bgp 65001 neighbor 10.100.100.2 timers keepalive '10'
```

Add an interface route to reach BGP listener

```
set protocols static interface-route 10.100.100.0/30 next-hop-
interface vti0
```

Clamp the VTI’s MSS to 1394 to avoid PMTU blackholes and also set MTU 1436 to VTI

```
set firewall options interface vti0 adjust-mss 1394
set interfaces vti vti0 mtu 1436
```

Note: AWS recommends setting the MSS on your customer gateway device to 1359 when using the SHA2-384 or SHA2-512 hashing algorithms. This is necessary to accommodate for the larger header.

Output:

```
vyos@ip-10-10-1-226:~$ sh vpn ike sa

Peer ID / IP                            Local ID / IP

------------                            -------------

185.144.208.250                         10.10.1.226

    State  IKEVer  Encrypt  Hash    D-H Group      NAT-T  A-Time  L-
Time

    -----  ------  -------  ----    ---------      -----  ------  ---
---

    up     IKEv1   aes256   sha256_128 2(MODP_1024)   no     1980
3600




vyos@ip-10-10-1-226:~$ sh vpn ipsec sa

Connection                        State    Up          Bytes In/Out
Remote address    Remote ID    Proposal

-------------------------------  -------  ----------  --------------
---------------  -----------  ------------------------------------
-------------------

peer-185.144.208.250-tunnel-vti  up       19 minutes  7K/9K
185.144.208.250   N/A
AES_CBC_256/HMAC_SHA2_256_128/PRF_HMAC_SHA2_256/MODP_1024
```