Firewall and NAT Assignment

1 . Create a virtual machine having the os centos

a. Install firewall in the vm(centos might have firewall installed in default).(firewalld or iptables)

Ans:
Firewalld is already installed in centos by default but if we need to install it we can use the command

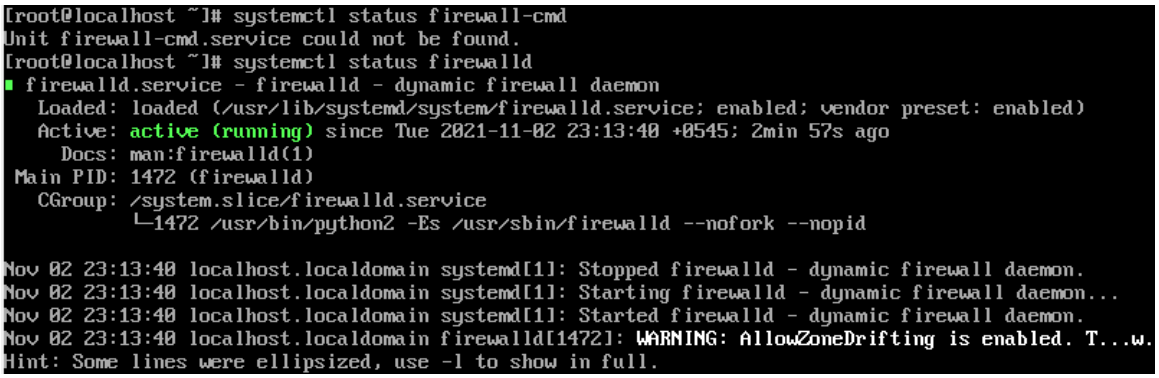
~ sudo yum install firewalld

We can check the status using the command

~ sudo firewall-cmd –state

OR

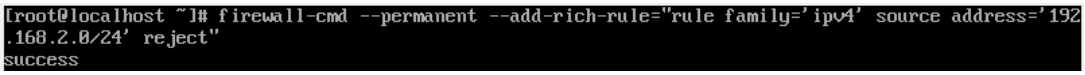
~ sudo systemctl status firwalld



b. Block certain ip range/subnet using firewalld.

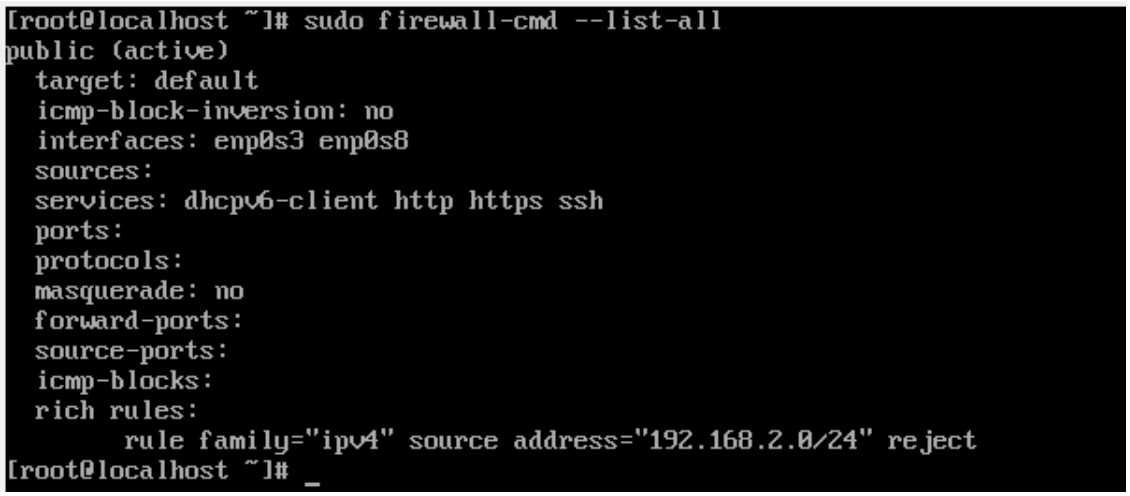
Ans:
To block certain Ip range/subnet we can use the command , here the ip range of 192.168.2.0 - 192.168.2.255 is blocked

~ sudo firewall-cmd --permanent –add-rich-rule=”rule family=’ipv4’ source address=’192.168.2.0/24’ reject”



Now the new rule can be seen using the command

~ sudo firewall-cmd --list-all



c. Allow http, https and ssh connection using firewall.

Ans:
Http, https and ssh are services which can be allowed or blocked, to allow them we can use the following command

~ sudo firewall-cmd --permanent --zone=public --add-service=<service name>

```
[root@localhost ~]# sudo firewall-cmd --permanent --zone=public --add-service=http
success
[root@localhost ~]# sudo firewall-cmd --permanent --zone=public --add-service=https
success
[root@localhost ~]# sudo firewall-cmd --permanent --zone=public --add-service=ssh
success
```

We can check the services allowed with the command

~ sudo firewall-cmd --permanent --zone=public --list-services

```
[root@localhost ~]# sudo firewall-cmd --zone=public --list-services
dhcpv6-client http https ssh
[root@localhost ~]#
```

d. You can add other rules as well as you prefer.

Note: The firewall rules should be saved permanently

Ans:

We can as well open ports or port range to the zone using the command

~ sudo firewall-cmd --permanent --zone=public --add-port=4040/tcp

```
[root@localhost ~]# sudo firewall-cmd --permanent --zone=public --add-port=4040/tcp
success
[root@localhost ~]# _
```

I almost all of the above commands **--permanent** flags were used to set the options permanently, but these options are not effective immediately, obly afffter service restart/reload.

So, we need to use restart the service after these so we need to use the command

~ sudo firewall-cmd --reload

So finally, the rules would be

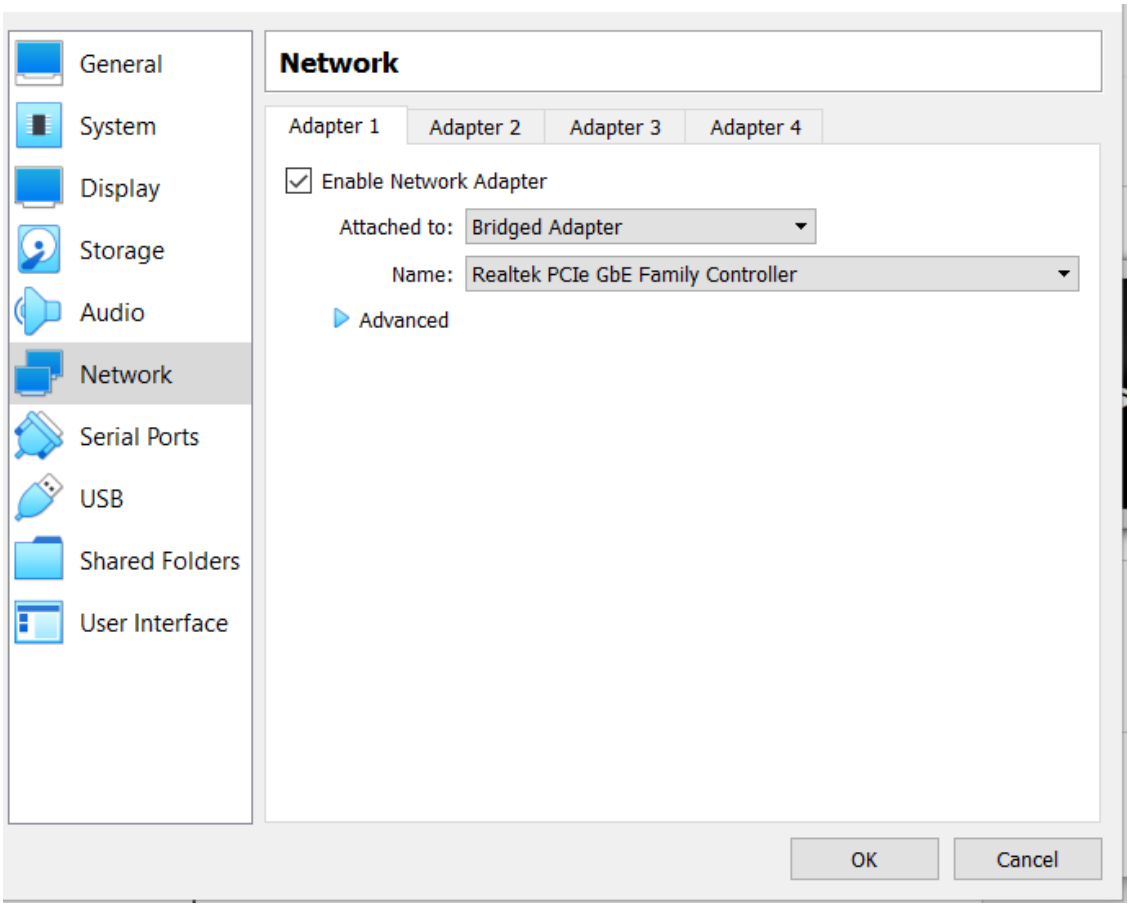
```
[root@localhost ~]# sudo firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3 enp0s8
  sources:
  services: dhcpv6-client http https ssh
  ports: 4040/tcp
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
        rule family="ipv4" source address="192.168.2.0/24" reject
[root@localhost ~]# _
```

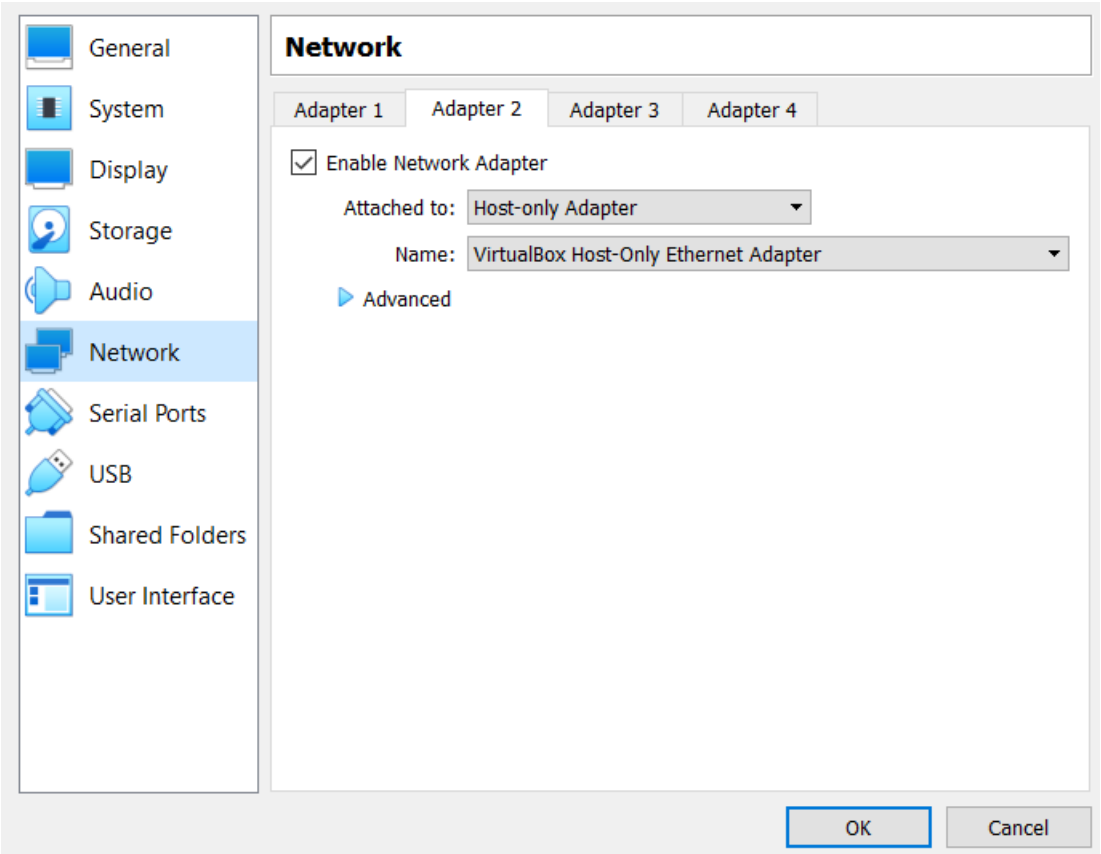
1. Create one vm with 2 network interfaces one should behave as WAN and another as LAN. Create another vm attaching the previously created LAN interface to it.
    a. Implement NAT in the first vm, so that the second vm can access the internet.

Note: Configure the first vm as a router, so make the LAN interfaces in the first vm as gateway to the LAN network. And in the second vm configure the gateway to the ip of the first vm LAN ip.

Ans:

First VM needs to have two network adapters set as:





Second VM needs only one adapter, i.e the Host -only adapter (Name must be same)

To configure the First VM

Firstly we set IP forward as 1

```
~ echo 1 > /proc/sys/net/ipv4/ip_forward
```

Run these commands to set the IPTABLE Routing rules for NATing

```
~ modprobe iptable_nat
~ iptables -F
~ iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
~ iptables -A FORWARD -i enp0s8 -o enp0s3 -j ACCEPT
```

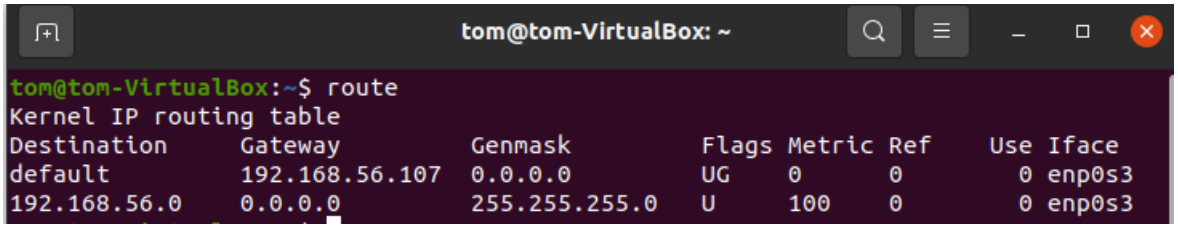
To configure the Second VM

We need to set the default gateway as the private IP of the first VM (Server), which can be done with the command

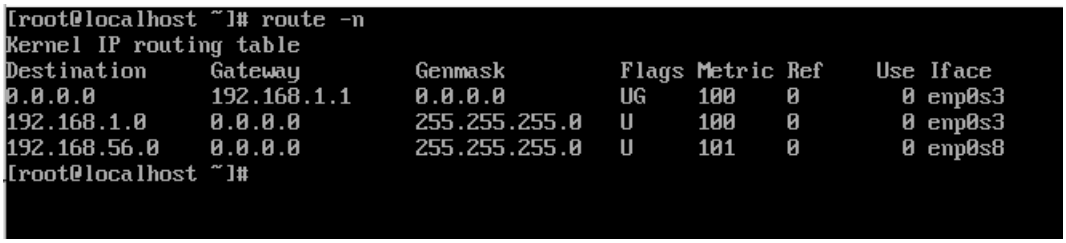
```
~ route add default gw 192.168.56.1
```

In this way Linux system can be used as NAT server and so can be used to access private ip

Client Routes (VM2):



Server Routes (VM1):



Pinging to google.com