Nginx can be used as web server or reverse proxy or load balancer

3 machines:
1. Client1 (Host Only-192.168.100.20) ---> To Host Site
2. Client2 (Host Only-192.168.100.30) ---> To Host Site
3. Kali Linux --> Reverse Proxy(Wan-NAT, LAN(Connecting to Backend Servers)-Host Only)
All Host only are in same LAN Network

Host Diff Websites on Client1 and Client2 and their ports on 80

On Client2(192.168.100.30):

```
sudo vi /etc/httpd/conf/httpd.conf
```

```
        Change listen port to 80
```

```
cd /var/www/html
```

```
vi Index.html
```

```
        Welcome to www.amazon.com/product
```

```
systemctl restart httpd
```

```
curl localhost
```

```
        This should our web page
```

Now in iptables rule http is allowed from inside to outside as new is not mentioned in rule
NEW: means allows from outside network to inside network
and by default for http it's not allowed with NEW

```
[root@localhost html]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
REJECT     all  --  anywhere             anywhere             reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  anywhere             anywhere             reject-with icmp-host-prohibited
```

As shown NEW is used for SSH as we access SSH from outside
So Now we have to access websites from outside so add NEW to http i.e TCP 80 rule

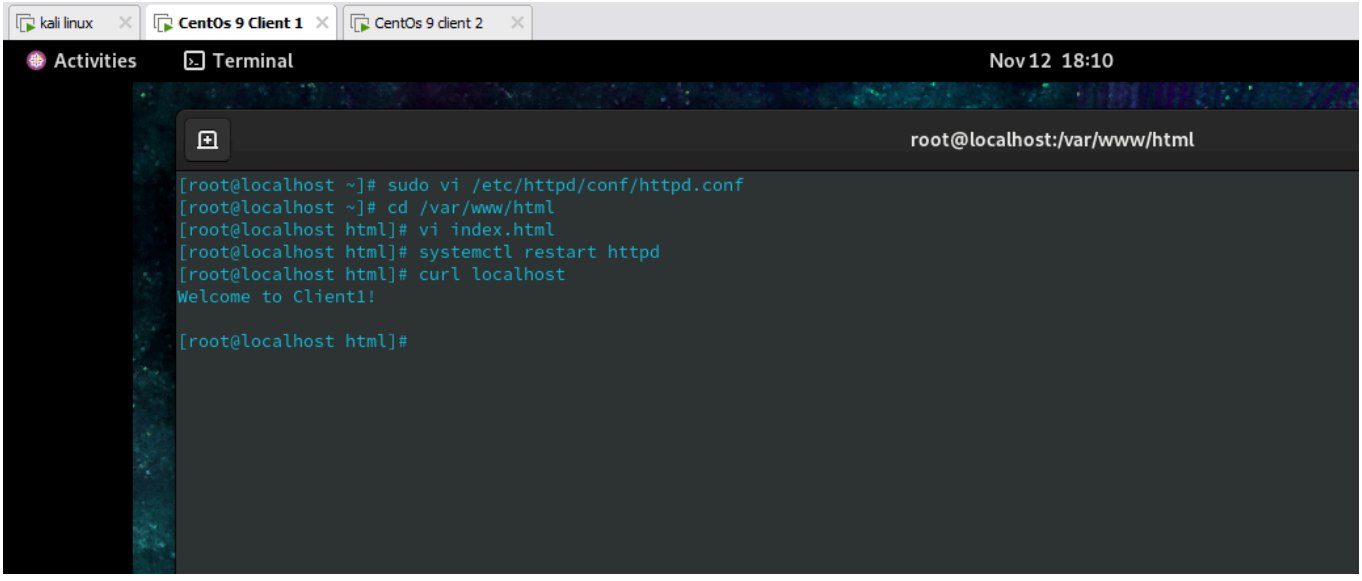
So Add Rule

```
iptables -I INPUT 2 -p tcp --dport 80 -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
```

```
[root@localhost html]# iptables -I INPUT 2 -p tcp --dport 80 -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT
[root@localhost html]# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere             tcp dpt:http state NEW,RELATED,ESTABLISHED
ACCEPT     icmp --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
REJECT     all  --  anywhere             anywhere             reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
REJECT     all  --  anywhere             anywhere             reject-with icmp-host-prohibited
```

On Client1(192.168.100.20): Same as Client2



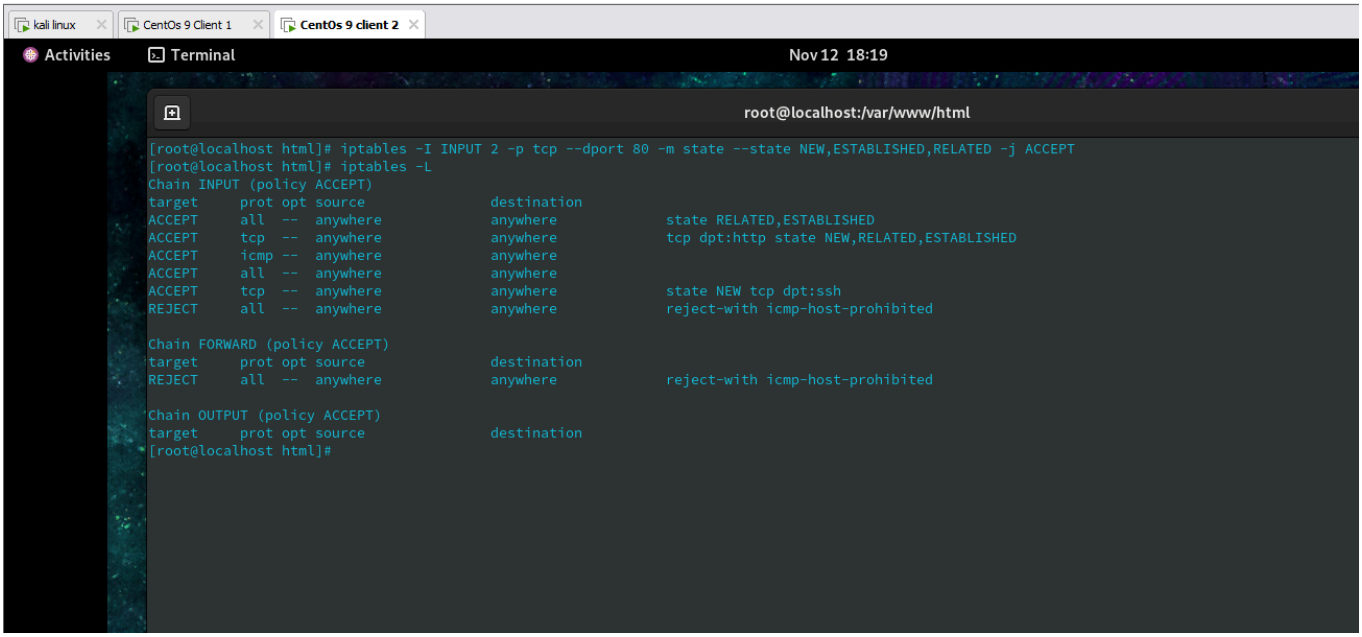
in index.html enter:

Welcome to www.amazon.com

```
iptables -I INPUT 2 -p tcp --dport 80 -m state --state NEW,ESTABLISHED,RELATED -j
ACCEPT
```

To check if rule is added:

```
iptables -L
```



Now try to access each other website like:
From Client1:

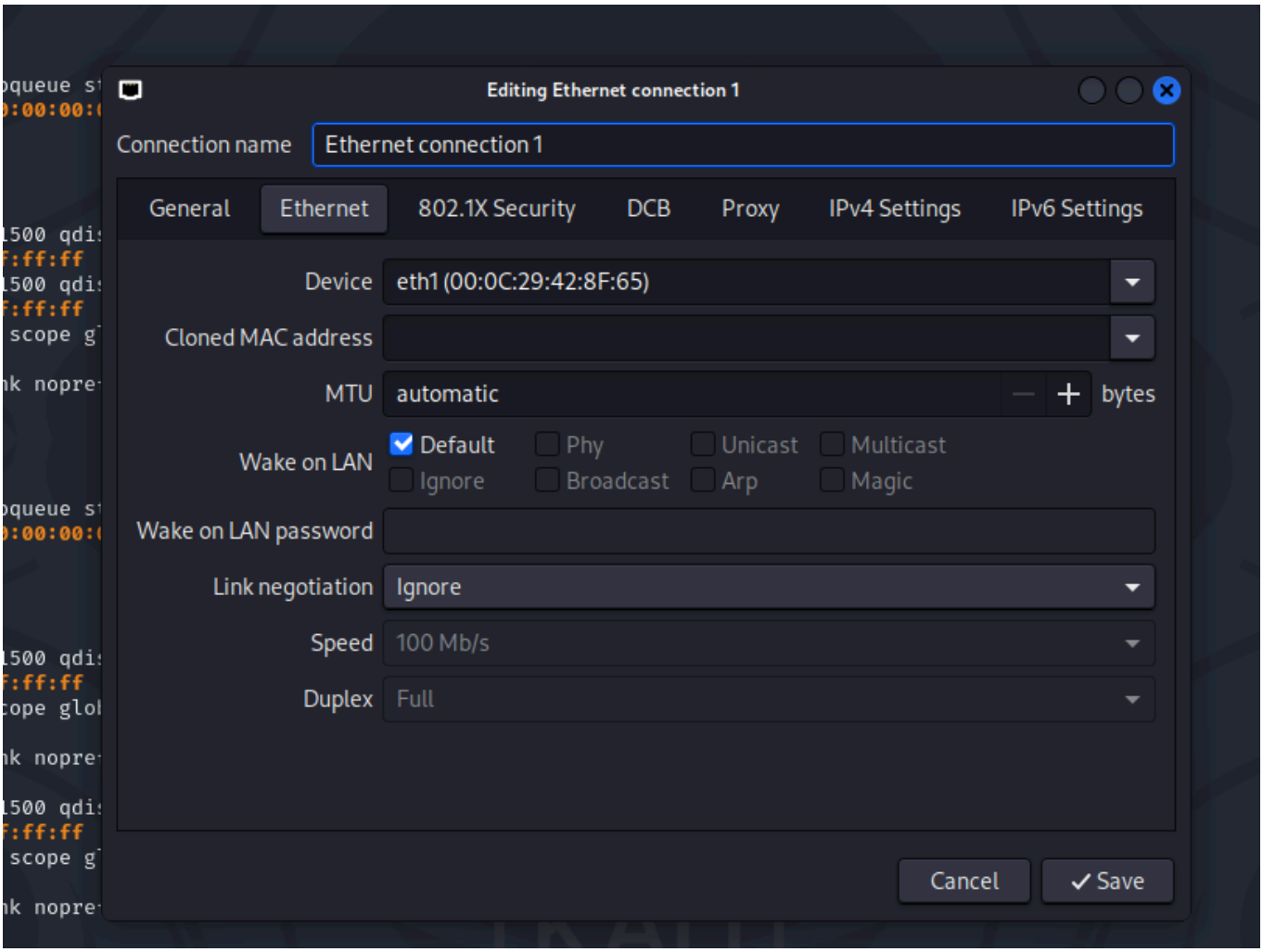
```
curl 192.168.100.30
```

From Client2:

```
curl 192.168.100.20
```

On Kali:

Setting --> Advanced Network Config --->
1. Select wiredConnection1 (For WAN(NAT)) --> Select Adapater eth0 ---> save
2. Click on add at bottom ---> Select Adapter eth1 ---> save




```
ip a
```

Host only(eth1) ---> same Network as of our Clients (192.168.100.128)
NAT(eth0) ---> Wan (192.168.75.139)

Now Install Nginx:

```
sudo apt install nginx -y
```

If this not work:

```
Sudo apt update -y
```

```
cd /etc/nginx
```

We need to create a copy of original file so if anything goes wrong we can revert back

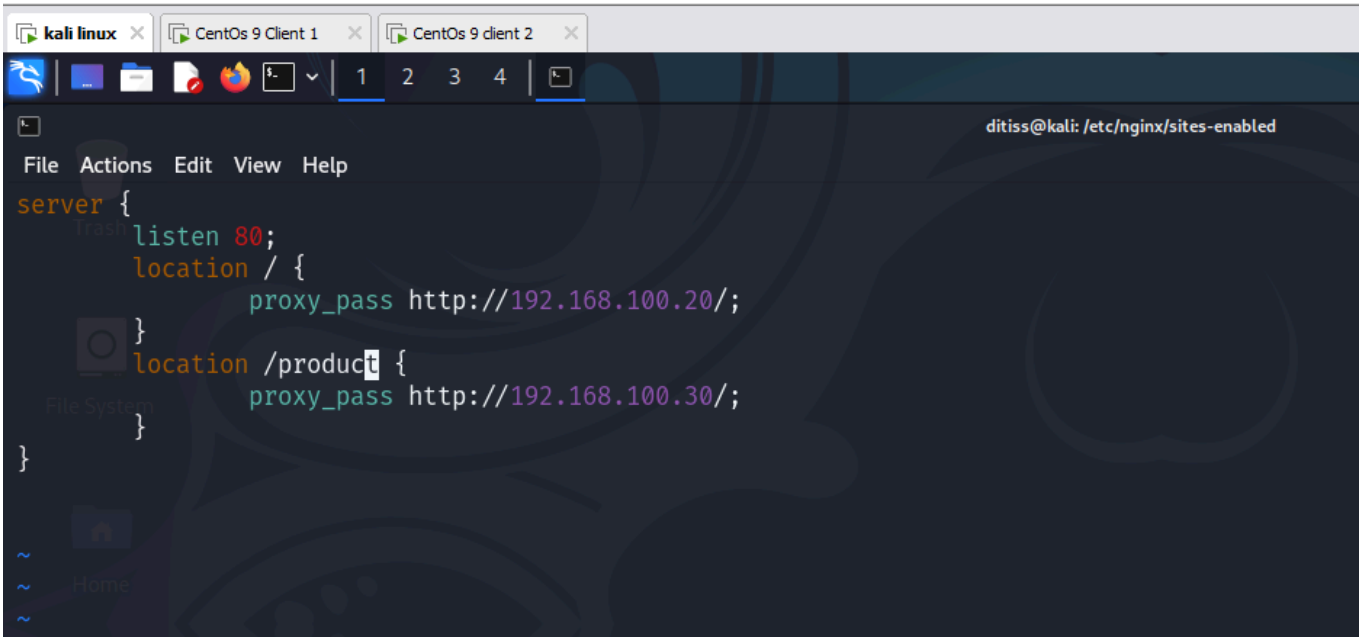
```
sudo cp nginx.conf nginx.conf.original
```

```
sudo vi /etc/nginx/nginx.conf
```

```
cd sites-enabled
```

```
sudo vi rproxy.conf
```

Enter in this file below data:

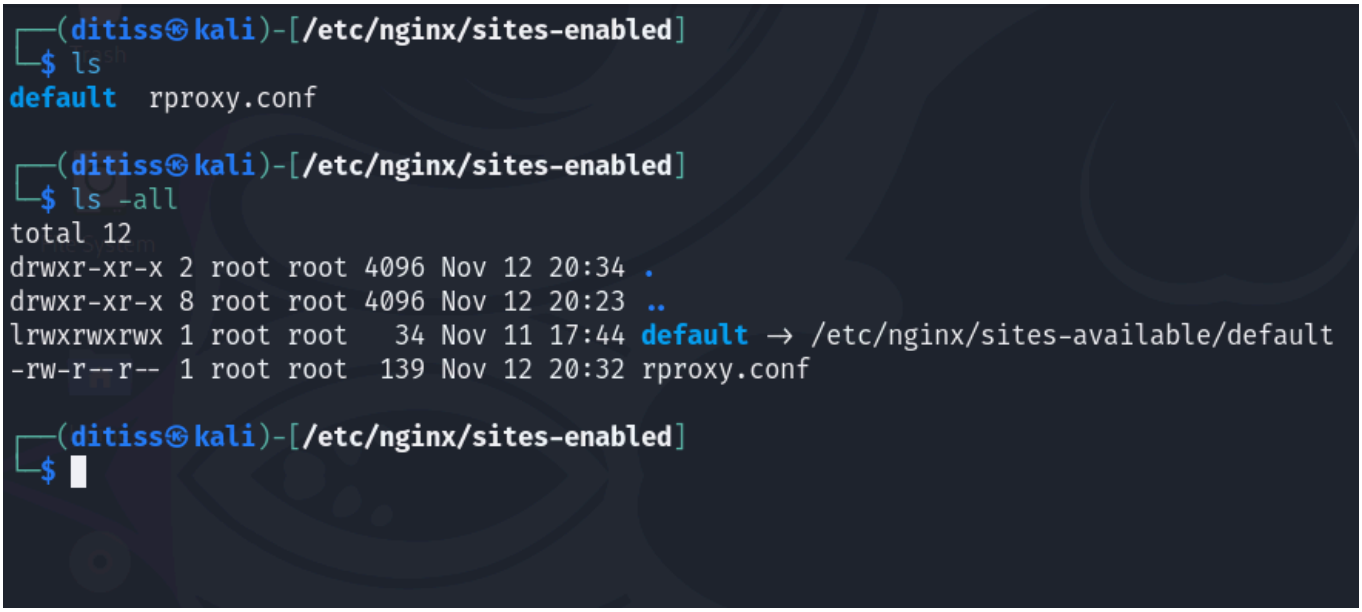


http{

}
can be only defined once and that is already define in nginx.conf file

so here only server is used

Now if try curl 192.168.75.139
it shows default page but we want of clinet1



Sites are actually made/configured in sites-available and then by creating symlink they are enabled through sites-enabled
But here we have made our sites on client1 and Client2 but on proxy we just need to enbaled them
but also the symlink named as "default" in sites-enabled is not allowing to enable our client1 and 2 sites so we will remove deafult from sites-enabled only

```
sudo rm -f default
```

```
sudo systemctl restart nginx
```

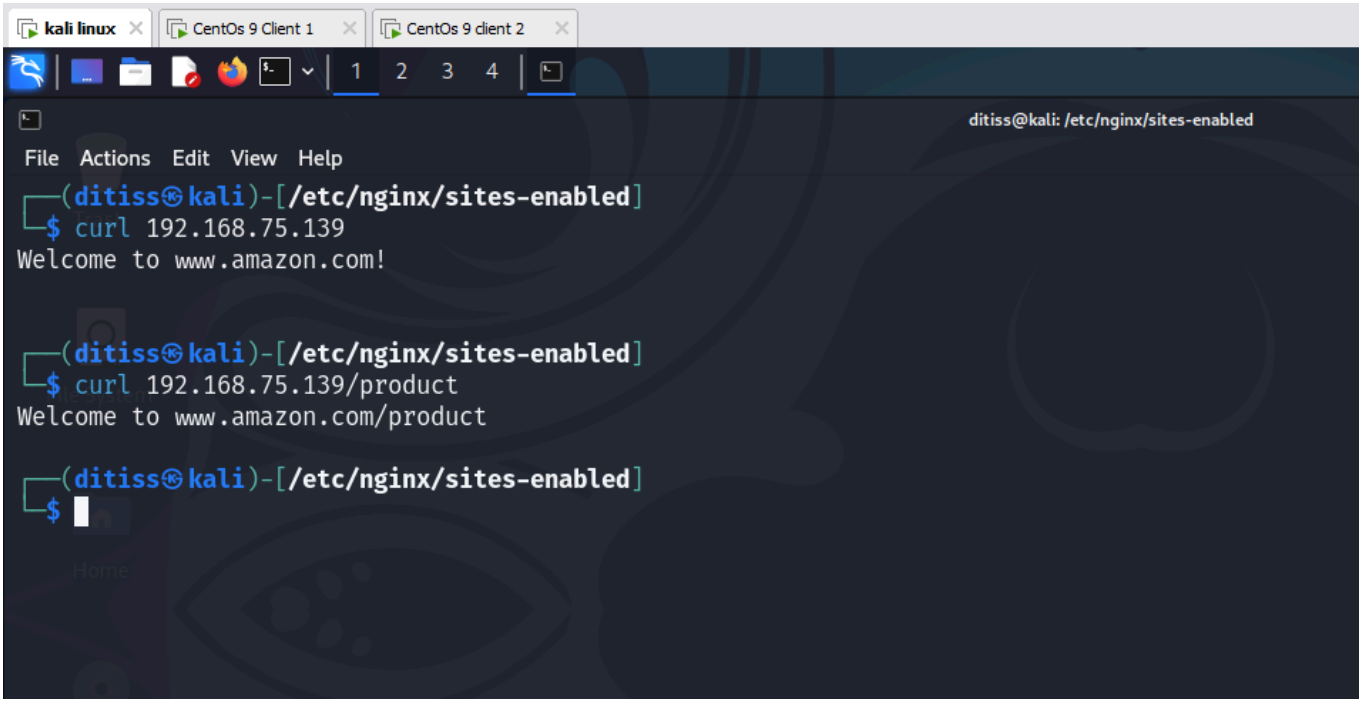
Now through WAN of proxy i.e from outside network we can access our sites

```
curl 192.168.75.139
```

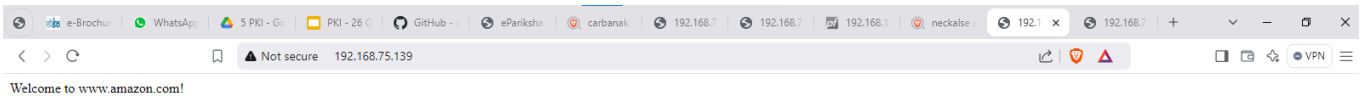
```
Welcome to www.amazon.com
```

```
curl 192.168.75.139/product
```

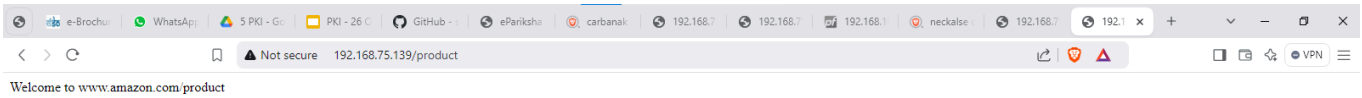
```
Welcome to www.amazon.com/product
```



Also Check from windows browser

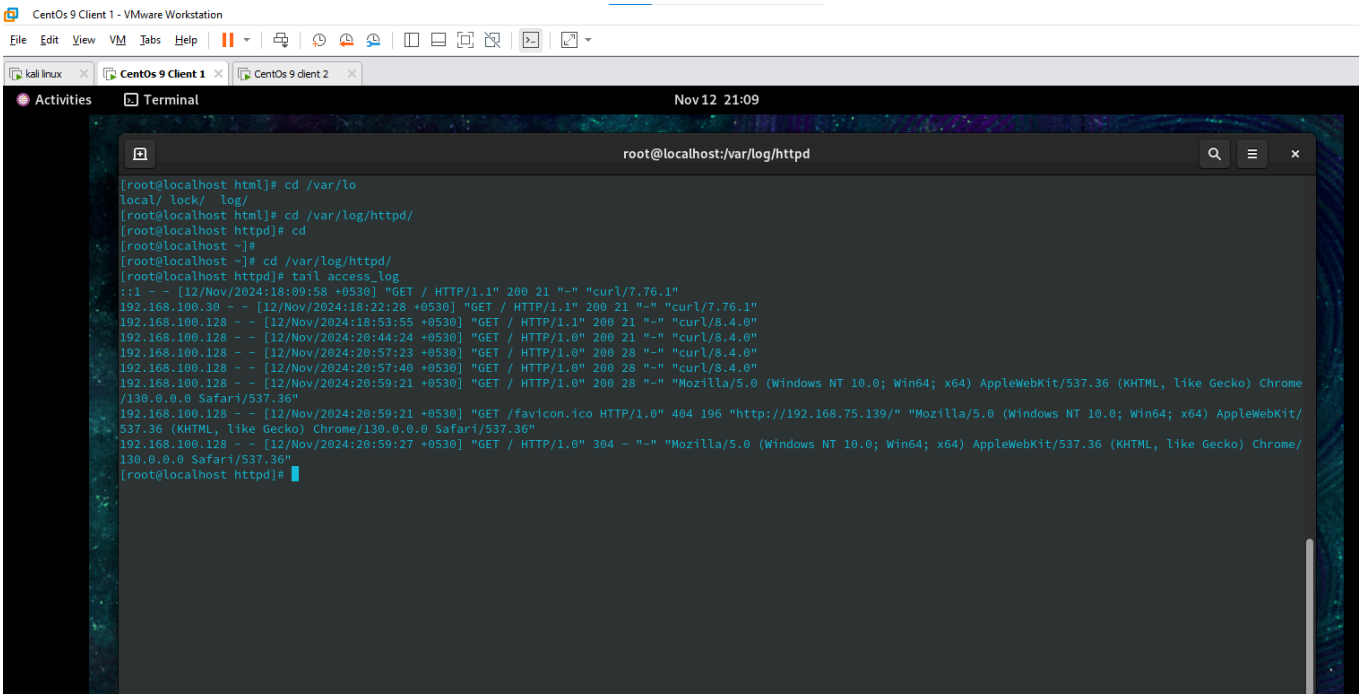


Welcome to www.amazon.com!



Welcome to www.amazon.com/product

Now we can check logs of client1(backend server)
logs contains IP of proxy LAN and not of outside client



So to forward outside clients IP:

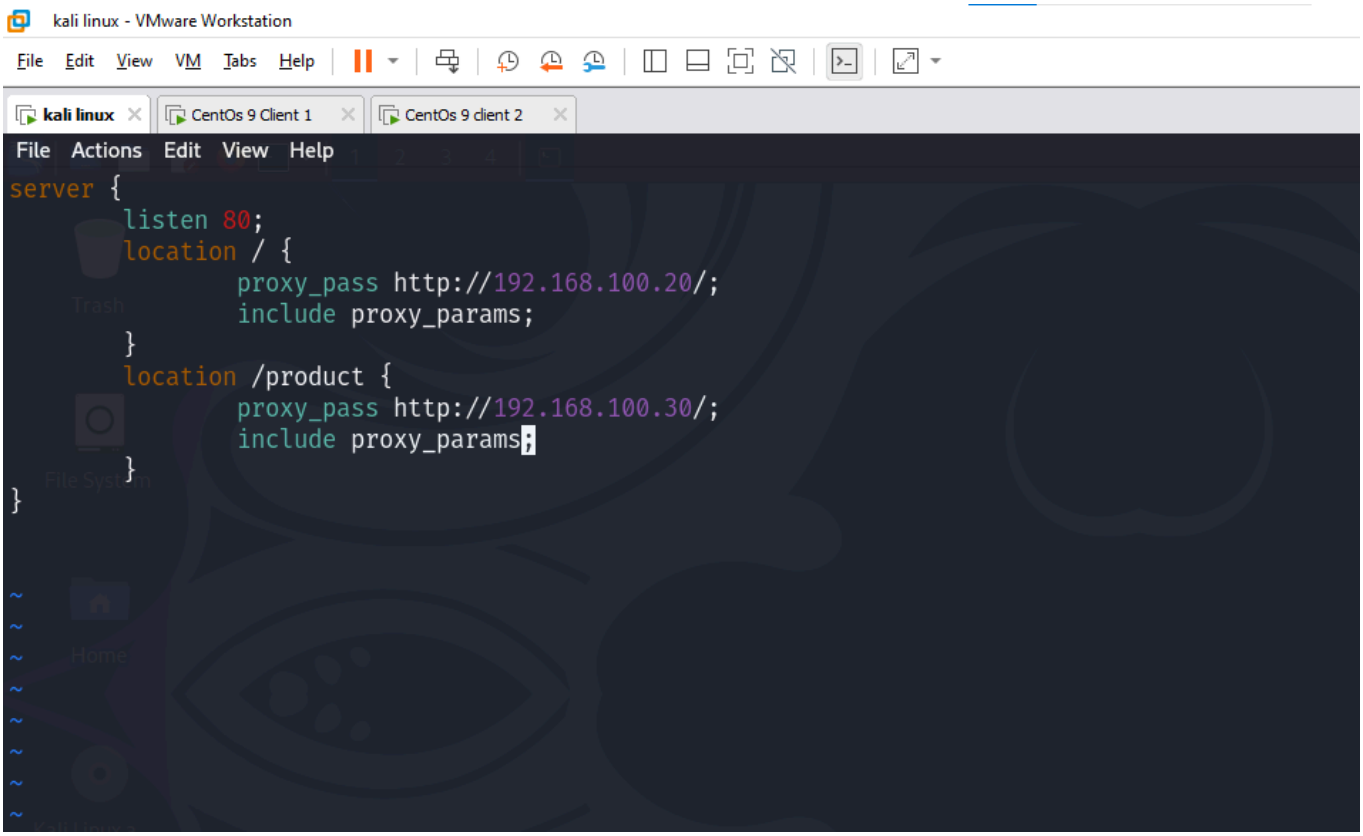
```
cd /etc/nginx/
cat proxy_params
```

```
        proxy_set_header Host $http_host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
```

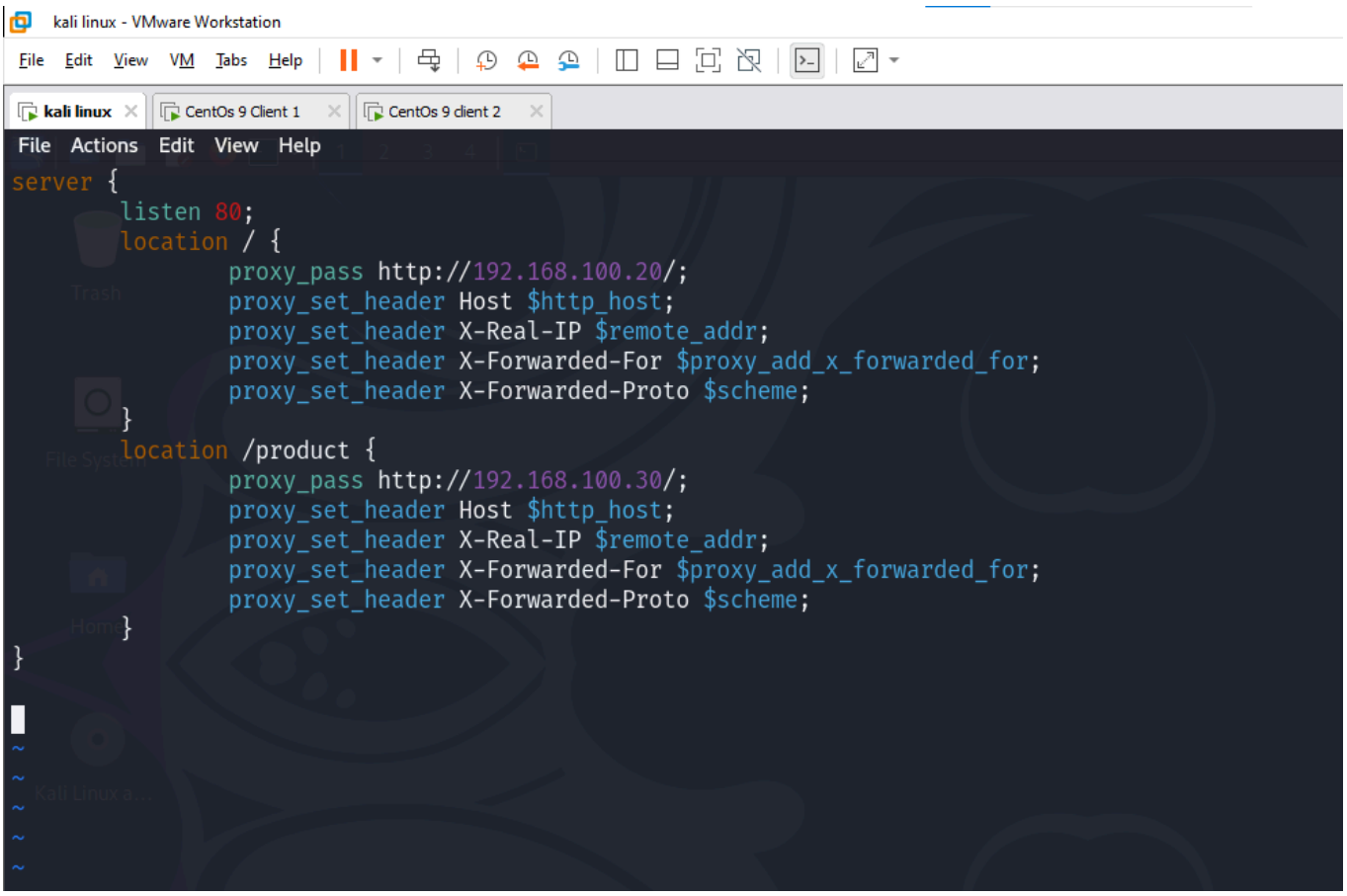
we need to add the parameters to rproxy file

```
cd /etc/nginx/sites-enabled
```

```
sudo vi rproxy.conf
```



OR



Now go to client1:

```
cd /var/log/httpd/
```

```
tail access_log
```

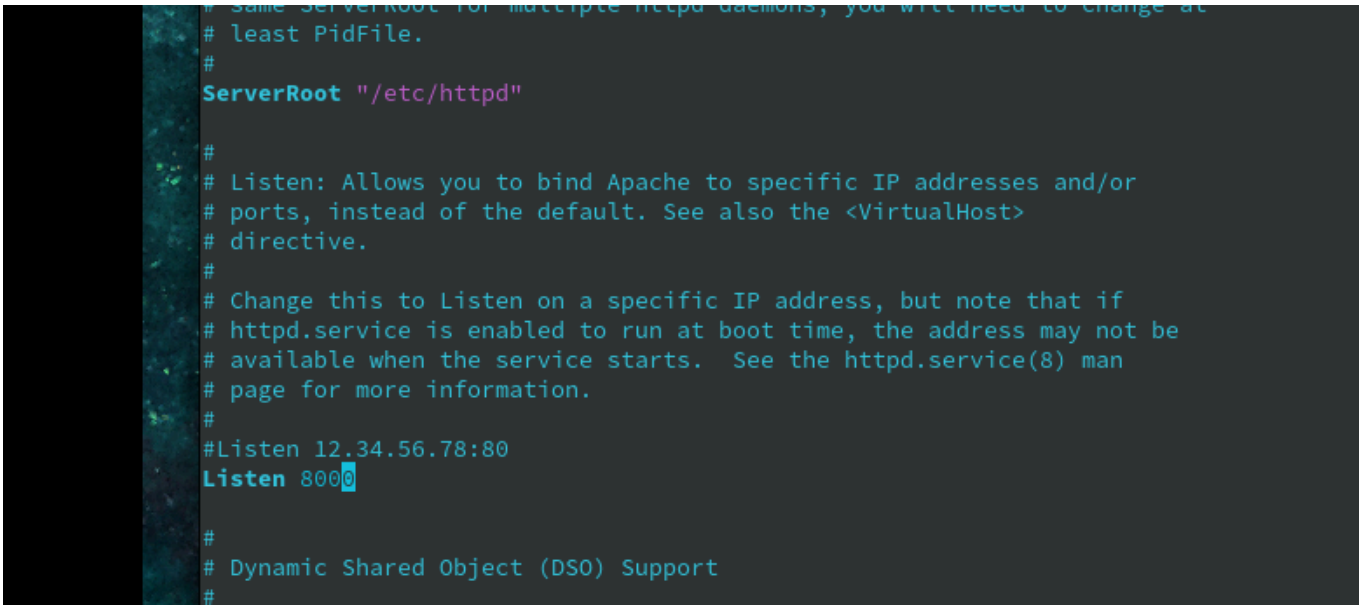
Now also we can't see outside clients IP becuase we are using VM and packets

```
sudo systemctl restart nginx
```

Now we will Change port of Client1 to 8000

```
sudo vi /etc/httpd/conf/httpd.conf
```

```
        listen 8000
```




```
sudo setenforce 0
```

```
systemctl restart httpd
```

```
sudo iptables -I INPUT 2 -p tcp --dport 8000 -j ACCEPT
```

```
curl localhost:8000
```

Now we will Change port of Client2 to 9000

```
sudo vi /etc/httpd/conf/httpd.conf
```

```
        Listen 9000
```

```
sudo setenforce 0
```

```
systemctl restart httpd
```

```
sudo iptables -I INPUT 2 -p tcp --dport 9000 -j ACCEPT
```

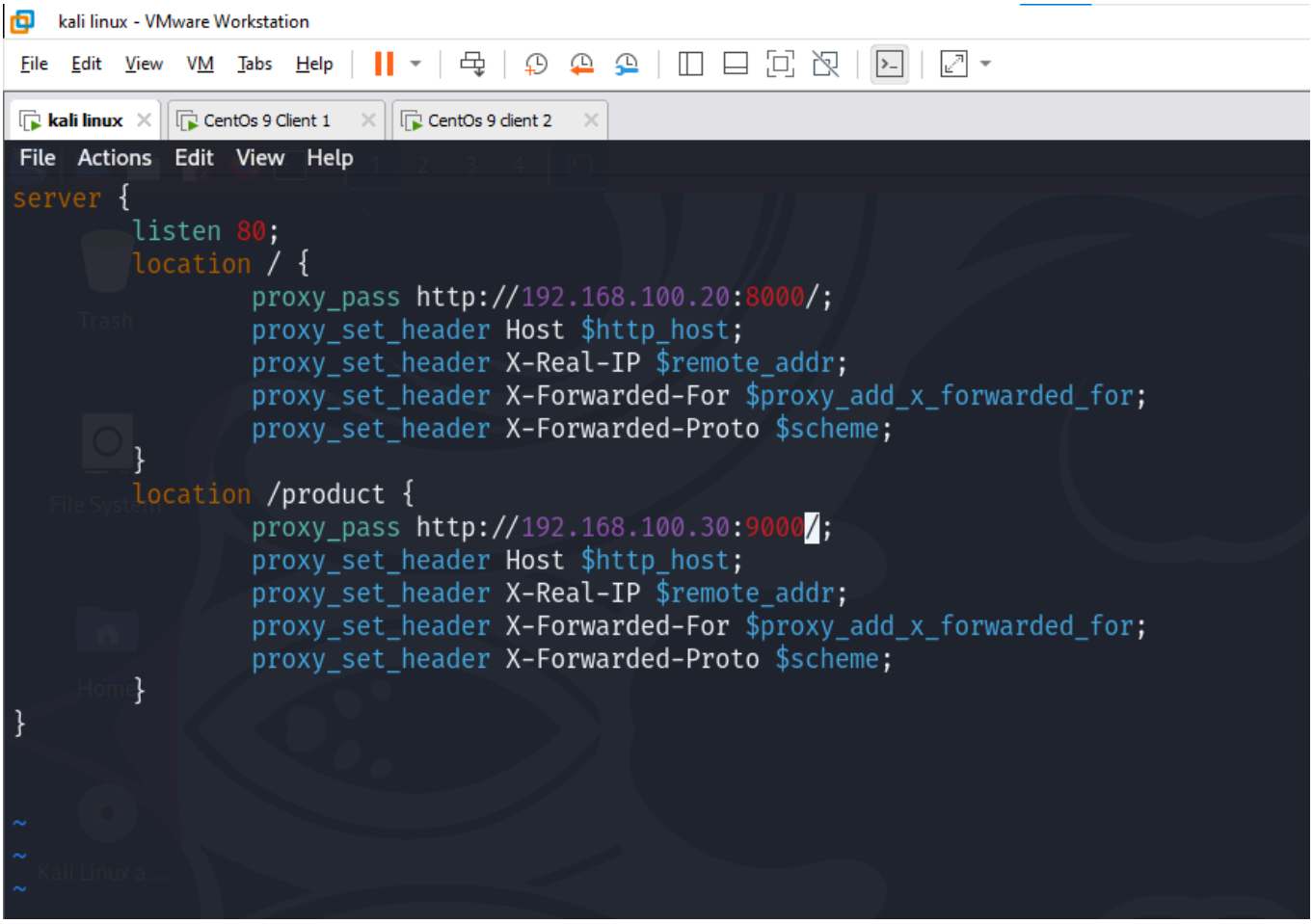
```
curl localhost:9000
```

Now go to Kali:

```
cd /etc/nginx/sites-enabled
```

```
sudo vi rproxy.conf
```

In rproxy file give port no:




```
sudo systemctl restart nginx
```

```
curl 192.168.75.139
```

```
        Welcome to www.amazon.com!
```

```
curl 192.168.75.139/product
```

```
        Welcome to www.amazon.com/product
```