

# Atividade Assíncrona - Semana 3

## Encaminhamento/Mascaramento

---

**Autor: Marcos Wesley Soares Alves**

**Matricula: 20181070130258**

Responda as seguintes perguntas:

1. O que é Servidor Nat?

R: Pode ser entendido como um servidor que será responsável por fazer a tradução de endereços de uma rede interna para em endereços da Internet.

2. Qual é o arquivo do GNU/Linux Debian que permite/bloqueia o encaminhamento de pacotes?

R: O nome do arquivo é `ip_foward` que se encontra no path: `/proc/sys/net/ipv4`

3. O que é mascaramento de pacotes?

R: É uma camada adicional de informação que permite ao servidor conhecer o host a qual o pacote deve ser encaminhado.

4. O que é necessário ativar para que o servidor torne-se provedor de internet para os cliente 01 e 02?

R: Ativar o encaminhamento de pacotes entre redes e fazer o mascaramento de pacotes.

```
echo "1" > /proc/sys/net/ipv4/ip_forward  
  
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

5. Como ativar permanentemente o encaminhamento/mascaramento de pacotes no servidor? Em outras palavras, crie um tutorial (passo-a-passo) que indique uma forma permanente que garanta que o servidor, mesmo após o processo de reboot, seja um servidor NAT para os clientes do cenário de simulação, ou seja, faça o encaminhamento e o mascaramento de pacotes.

# Tutorial

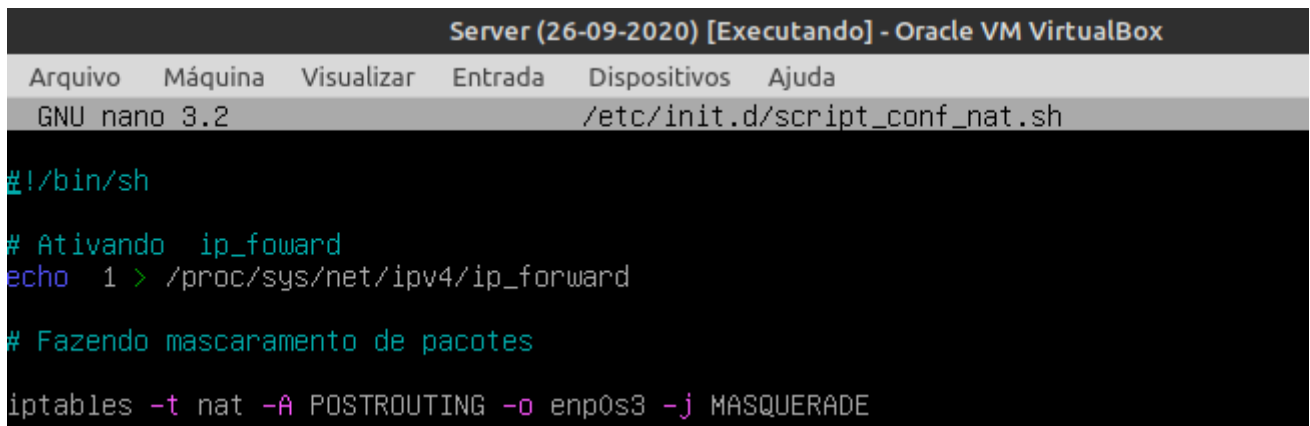
## Criação de um script shell

1 - Criar um arquivo shell script no diretório /etc/init.d que será executado sempre que o sistema operacional inicializar

```
root@servidor:~# touch /etc/init.d/script_encaminhamento_mascaramento.sh
root@servidor:~# ls /etc/init.d/
apparmor      dbus          kmod          rsyslog       teste.sh
console-setup.sh hwclock.sh    networking    script_encaminhamento_mascaramento.sh udev
cron          keyboard-setup.sh procps        ssh
```

2 - Adicionar os comando de ativação NAT e mascaramento de pacotes no arquivo shell.

```
nano script_encaminhamento_mascaramento.sh
```



The screenshot shows a terminal window titled "Server (26-09-2020) [Executando] - Oracle VM VirtualBox". The terminal is running the GNU nano 3.2 editor, editing the file /etc/init.d/script\_conf\_nat.sh. The content of the file is as follows:

```
#!/bin/sh

# Ativando ip_foward
echo 1 > /proc/sys/net/ipv4/ip_forward

# Fazendo mascaramento de pacotes
iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
```

Após salvar o arquivo digite o seguinte comando para permitir a execução do script. Lembrando que é necessário estar dentro do mesmo diretório em que o arquivo se encontra(/etc/init.d).

```
chmod+x script_encaminhamento_mascaramento.sh
```

3 - Crie um "link" desse arquivo no diretório /etc/rc2.d, porém no início do arquivo deve conter S02. O S é referente a execução do script o 02 é o numero que ele será executado ou seja ele sera o segundo,a ser executado na inicialização do seu sistema.

```
cp /etc/init.d/script_encaminhamento_mascaramento.sh
/etc/rc2.d/S02script_encaminhamento_mascaramento.sh
```

```

root@servidor:/etc# cp /etc/init.d/script_encaminhamento_mascaramento.sh /etc/rc2.d/S02script_encami
nhamento_mascaramento.sh
root@servidor:/etc# cd /etc/rc
rc0.d/ rc1.d/ rc2.d/ rc3.d/ rc4.d/ rc5.d/ rc6.d/ rcS.d/
root@servidor:/etc# cd /etc/rc2.d/
root@servidor:/etc/rc2.d# ls
S01console-setup.sh  S01dbus  S01ssh  S02teste.sh
S01cron  S01rsyslog  S02script_encaminhamento_mascaramento.sh  script_shell.sh
root@servidor:/etc/rc2.d# _

```

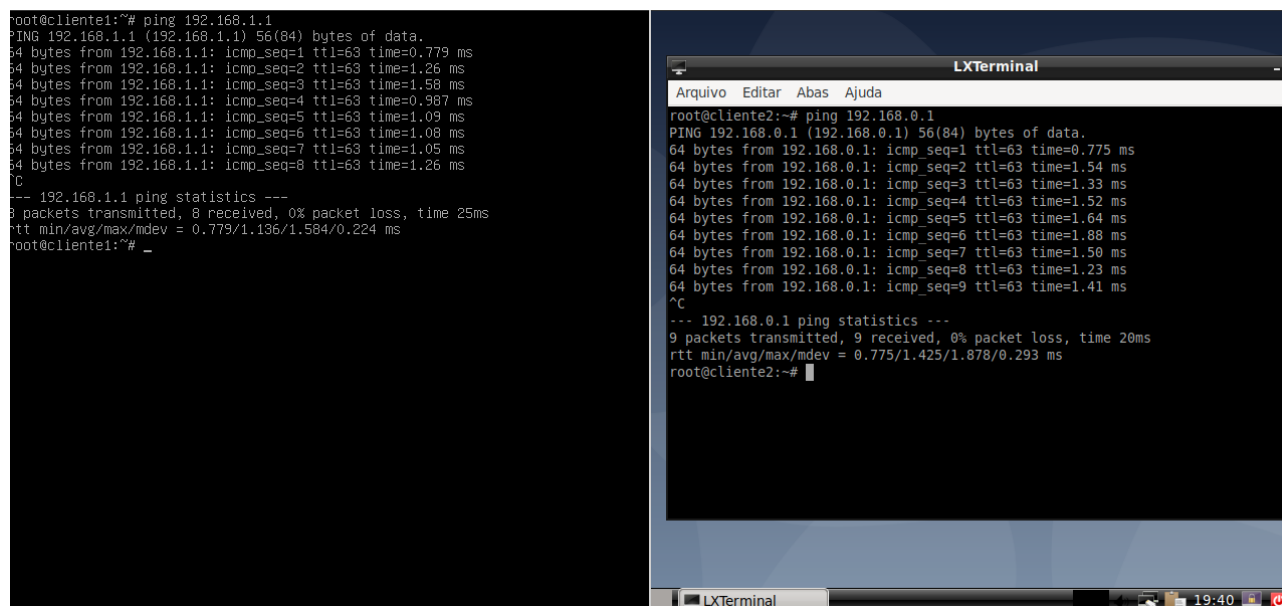
4 - Agora é só reiniciar o servidor. Para testar, tente pingar o cliente1(192.168.0.1) no cliente2(192.168.1.1). Um outro teste pode ser a atualização de pacotes.

```

ping 192.168.1.1
e/ou
apt-get update

```

- Teste com ping entre maquinas



```

root@cliente1:~# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=63 time=0.779 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=63 time=1.26 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=63 time=1.58 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=63 time=0.987 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=63 time=1.09 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=63 time=1.08 ms
64 bytes from 192.168.1.1: icmp_seq=7 ttl=63 time=1.05 ms
64 bytes from 192.168.1.1: icmp_seq=8 ttl=63 time=1.26 ms
^C
--- 192.168.1.1 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 25ms
rtt min/avg/max/mdev = 0.779/1.136/1.584/0.224 ms
root@cliente1:~# _

root@cliente2:~# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp seq=1 ttl=63 time=0.775 ms
64 bytes from 192.168.0.1: icmp seq=2 ttl=63 time=1.54 ms
64 bytes from 192.168.0.1: icmp seq=3 ttl=63 time=1.33 ms
64 bytes from 192.168.0.1: icmp seq=4 ttl=63 time=1.52 ms
64 bytes from 192.168.0.1: icmp seq=5 ttl=63 time=1.64 ms
64 bytes from 192.168.0.1: icmp seq=6 ttl=63 time=1.88 ms
64 bytes from 192.168.0.1: icmp seq=7 ttl=63 time=1.50 ms
64 bytes from 192.168.0.1: icmp seq=8 ttl=63 time=1.23 ms
64 bytes from 192.168.0.1: icmp_seq=9 ttl=63 time=1.41 ms
^C
--- 192.168.0.1 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 20ms
rtt min/avg/max/mdev = 0.775/1.425/1.878/0.293 ms
root@cliente2:~#

```

- Teste com apt-get

```

root@cliente1:~# apt-get update
Atingido:1 http://security.debian.org/debian-security buster/updates InRelease
Atingido:2 http://deb.debian.org/debian buster InRelease
Atingido:3 http://deb.debian.org/debian buster-updates InRelease
Lendo listas de pacotes... Pronto
root@cliente1:~# _

```

- Teste com interface gráfica (a partir do cliente 2)

