



UNIVERSIDADE FEDERAL DO CEARÁ

Trabalho Final de Segurança da Informação

Davi Bezerra yada da Silva
George Ricardo Rodrigues Silva
Carlos Eduardo Alves Almeida

Quixadá-CE
2024

Este trabalho final foi reproduzido nas máquinas de cada integrante da equipe, mas para fins de relatório, serão utilizadas apenas as configurações feitas na máquina do aluno Davi Bezerra Yada da Silva.

*

Máquinas com respectivas interfaces e IPs

- **Firewall:**
Bridge enp0s3 (192.168.100.181);
Host-Only enp0s8 (192.168.243.4);
Host-Only enp0s9 (192.168.15.4)
- **Cliente:**
Host-Only enp0s8 (192.168.243.3)
- **WebServer**
Host-Only enp0s9 (192.168.15.3)

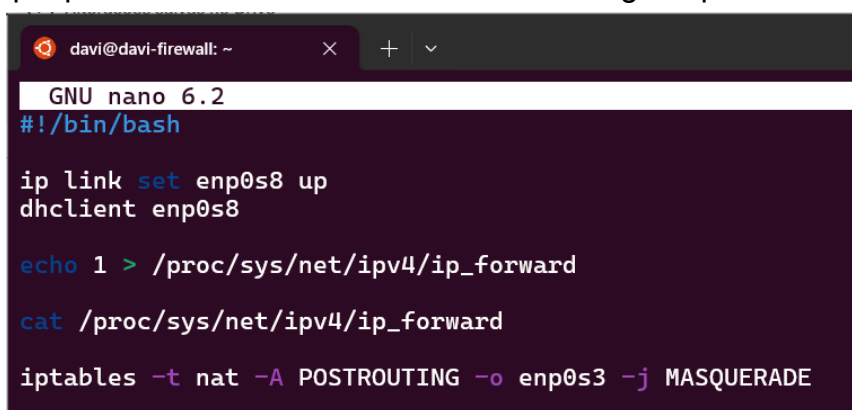
*

Preparação de ambiente

No Virtual Box, duas máquinas são criadas para a formação inicial da rede, uma máquina **Cliente** e uma máquina **Firewall**. A máquina Cliente possui uma interface de rede Host-Only enp0s8 e o Firewall possui uma interface no modo Bridge enp0s3 e uma Host-Only enp0s8. A máquina cliente conseguiria acesso à internet via firewall pela rede Host-Only.

Configuração Inicial do Firewall

Com as interfaces criadas, um script é usado para ativar a interface de rede Host-Only e permitir o encaminhamento ipv4, além de adicionar uma regra iptables que permite o uso de NAT via interface Bridge enp0s3.



```
davi@davi-firewall: ~
GNU nano 6.2
#!/bin/bash

ip link set enp0s8 up
dhclient enp0s8

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

cat /proc/sys/net/ipv4/ip_forward

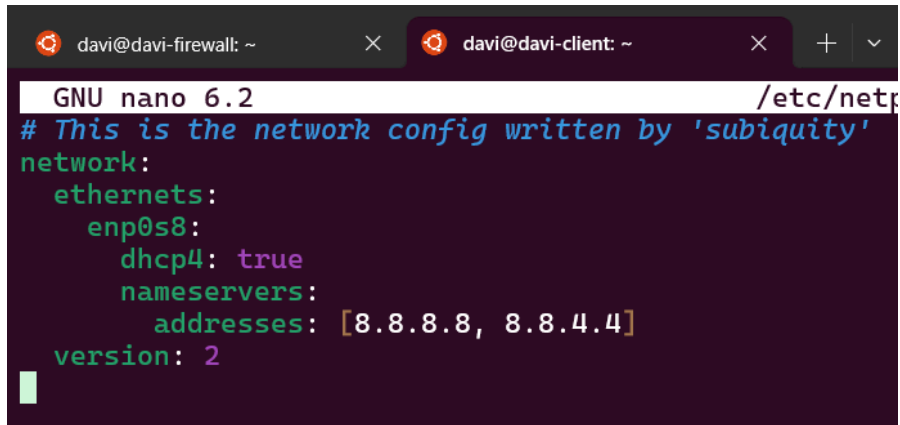
iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
```

Configuração Inicial do Cliente

Configuração de DNS

- Primeiro, acessamos o arquivo **/etc/netplan/00-installer-config.yaml** e adicionamos o ip de DNS padrão com as seguintes configurações. Após isso,

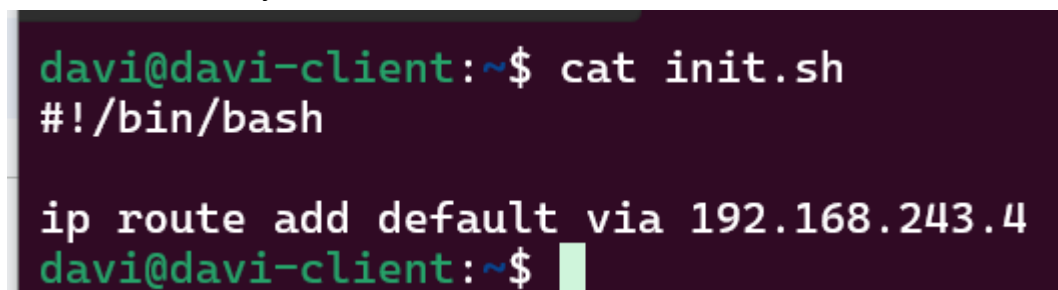
executamos o comando **sudo netplan apply** para aplicação das configurações.



```
GNU nano 6.2 /etc/netplan/01-netcfg.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s8:
      dhcp4: true
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
  version: 2
```

Configuração de Gateway padrão

- Um script é usado para definir a rota para acesso à rede externa com o ip da interface Host-Only do Firewall



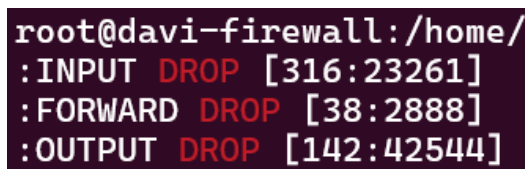
```
davi@davi-client:~$ cat init.sh
#!/bin/bash

ip route add default via 192.168.243.4
davi@davi-client:~$
```

Configuração de IPTABLES para Firewall

Primeiramente, o iptables será utilizado para aplicação de regras de controle de tráfego, para a máquina firewall e para a máquina cliente. As seguintes regras são adicionadas ao iptables:

- Política DROP para chains INPUT, OUTPUT e FORWARD, com o comando **sudo iptables -P <nome da chain> DROP**



```
root@davi-firewall:/home/
:INPUT DROP [316:23261]
:FORWARD DROP [38:2888]
:OUTPUT DROP [142:42544]
```

Seguindo a imagem, as regras são definidas para descartar todo tráfego de entrada, encaminhamento e de saída.

- tráfego de LoopBack Firewall para comunicação local do sistema e de processos internos, com os comandos para entrada e saída:

```
sudo iptables -A INPUT -i lo -j ACCEPT
sudo iptables -A OUTPUT -o lo -j ACCEPT.
```

```
root@davi-firewall:/home/davi#
-A INPUT -i lo -j ACCEPT
-A OUTPUT -o lo -j ACCEPT
```

teste:

```
root@davi-firewall:/home/davi# ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.073 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.091 ms
^C
--- 127.0.0.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1032ms
rtt min/avg/max/mdev = 0.073/0.082/0.091/0.009 ms
root@davi-firewall:/home/davi#
```

- Permissão de tráfego estabelecido e relacionado. Antes da adição das outras regras, duas regras são adicionadas para permitir tráfego estabelecido e relacionado com conntrack para regras posteriores.

```
sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED
-j ACCEPT
```

```
sudo iptables -A OUTPUT -m conntrack --ctstate ESTABLISHED -j
ACCEPT
```

- Permissão para acesso remoto via ssh

```
root@davi-firewall:/home/davi# iptables-save | grep 22 | grep INPUT
-A INPUT -p tcp -m tcp --dport 22 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste:

```
davi@LAPTOP-51IJ0R3M:~$ ssh davi@192.168.243.4
davi@192.168.243.4's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-117-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro
```

- Permissão de ICMP

```
root@davi-firewall:/home/davi# iptables-save | grep icmp | grep INPUT
-A INPUT -p icmp -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste:

```
root@davi-firewall:/home/davi# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=70.5 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 1 received, 50% packet loss, time 1002ms
rtt min/avg/max/mdev = 70.546/70.546/70.546/0.000 ms
root@davi-firewall:/home/davi#
```

- Permissão de consultas DNS

```
root@davi-firewall:/home/davi# iptables-save | grep 53 | grep OUTPUT
-A OUTPUT -p udp -m udp --dport 53 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste:

```
root@davi-firewall:/home/davi# nslookup google.com
Server:          127.0.0.53
Address:         127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 142.251.132.46
Name:   google.com
Address: 2800:3f0:4001:834::200e
```

Configuração de IPTABLES para Cliente

Após adição de regras para o Firewall, algumas regras são definidas para o cliente para permissão de tráfegos via encaminhamento (chain FORWARD)

- Permissão de acesso remoto via ssh

```
root@davi-firewall:/home/davi# iptables-save | grep 22 | grep FOR
-A FORWARD -i enp0s8 -o enp0s3 -p tcp -m tcp --dport 22 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste:

```
davi@LAPTOP-51IJ0R3M:~$ ssh davi@192.168.243.6
davi@192.168.243.6's password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-117-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage
```

- Permissão de Consultas DNS

```
root@davi-firewall:/home/davi# iptables-save | grep 53 | grep FOR
-A FORWARD -i enp0s8 -o enp0s3 -p udp -m udp --dport 53 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste:

```
davi@davi-client:~$ nslookup google.com
Server:          127.0.0.53
Address:         127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 142.251.128.238
Name:   google.com
Address: 2800:3f0:4001:805::200e
```

- Permissão de HTTP, HTTPS

sudo iptables -A FORWARD -i enp0s8 -o enp0s3 -p tcp -m multiport --dports 80,443 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
(a imagem estava muito grande, adicionamos apenas o comando)

teste:

```
davi@davi-client:~$ wget google.com
--2024-09-22 22:48:33-- http://google.com/
Resolving google.com (google.com)... 142.251.128.238, 2800:3f0:4001:805::200e
Connecting to google.com (google.com)|142.251.128.238|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.google.com/ [following]
--2024-09-22 22:48:33-- http://www.google.com/
Resolving www.google.com (www.google.com)... 172.217.29.36, 2800:3f0:4001:81a::2004
Connecting to www.google.com (www.google.com)|172.217.29.36|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html'

index.html [ <=>]
2024-09-22 22:48:34 (403 KB/s) - 'index.html' saved [21560]
```

- Permissão para tráfego FTP

```
root@davi-firewall:/home/davi# iptables-save | grep 21 | grep FOR
-A FORWARD -i enp0s8 -o enp0s3 -p tcp -m multiport --dports 20,21 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste:

```
davi@davi-client:~$ telnet ftp.uem.br 21
Trying 186.233.154.17...
Connected to armazem.uem.br.
Escape character is '^]'.
220 FTP Server ready.
hello
500 HELLO not understood
quit
221 Goodbye.
Connection closed by foreign host.
davi@davi-client:~$
```

- Permissão de tráfego SMTP

```
root@davi-firewall:/home/davi# iptables-save | grep 587 | grep FOR
-A FORWARD -i enp0s8 -o enp0s3 -p tcp -m multiport --dports 465,587 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste:

```
davi@davi-client:~$ telnet smtp.gmail.com 587
Trying 142.250.0.108...
Connected to smtp.gmail.com.
Escape character is '^]'.
220 smtp.gmail.com ESMTP 6a1803df08f44-6c75e45eaf0sm41489726d6.31 - gsmt
quit
221 2.0.0 closing connection 6a1803df08f44-6c75e45eaf0sm41489726d6.31 - gsmt
Connection closed by foreign host.
```

- Permissão de tráfego ICMP

```
root@davi-firewall:/home/davi# iptables-save | grep icmp | grep enp0s8
-A FORWARD -i enp0s8 -o enp0s3 -p icmp -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste tanto para ip da interface firewall quanto para rede externa:

```
davi@davi-client:~$ ping 192.168.243.4
PING 192.168.243.4 (192.168.243.4) 56(84) bytes of data.
64 bytes from 192.168.243.4: icmp_seq=1 ttl=64 time=1.98 ms
64 bytes from 192.168.243.4: icmp_seq=2 ttl=64 time=0.626 ms
^C
--- 192.168.243.4 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.626/1.304/1.982/0.678 ms
davi@davi-client:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=83.9 ms
^C
--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 83.868/83.868/83.868/0.000 ms
```

Configuração Inicial para Proxy Squid

O squid é o serviço que será usado como proxy para requisições http e https que será instalado no Firewall.

- Configuração de NAT para redirecionamento: Adicionamos regras iptables para redirecionar tráfego http e https (portas 80 e 443, respectivamente) para as portas 3129 e 3130 no squid com os comandos:

```
sudo iptables -t nat -A PREROUTING -i enp0s8 -p tcp -m tcp --dport 80 -j REDIRECT --to-ports 3129
```

```
sudo iptables -t nat -A PREROUTING -i enp0s8 -p tcp -m tcp --dport 443 -j REDIRECT --to-ports 3130
```

- Instalação e Configuração

A instalação do squid é feita com o comando **sudo apt install squid**

Após instalação, acessamos o arquivo de configuração **/etc/squid/squid.conf** e adicionar as regras para interceptação das portas 80 e 443 definidas no iptables via portas 3129 e 3130:

```
GNU nano 6.2
http_port 3128
http_port 3129 intercept
http_port 3130 intercept
```

- Para controlar requisições feitas, algumas regras chamadas ACL são adicionadas ao arquivo de configuração, que definem o que será feito com cada requisição baseado em rede de origem e URLs permitidas em cada acesso. As ACLs podem ser vistas na imagem do arquivo de configuração a seguir.


```
include /etc/squid/conf.d/*.conf

acl redetf src 192.168.243.0/24

acl blockuece url_regex uece
http_access deny redetf blockuece

acl blockufc url_regex ufc
http_access deny redetf blockufc

acl blacklist dstdomain "/etc/squid/blacklist"
http_access deny blacklist

http_access allow redetf

http_access deny all
```

- A rede 192.168.243.0/24 (rede Firewall) é adicionada como origem com a acl redetf src 192.168.243.0/24.
- As próximas ACLs definem URLs de acordo com strings presentes no nome de domínio, como uece e ufc. Estas tem seu tráfego bloqueado com o comando **http_access deny <nome da acl>**. Por exemplo, a ACL blockufc irá bloquear requisições que contenham a string ufc no nome de domínio.
- A ACL blacklist traz essa mesma abordagem, mas as strings procuradas são informadas por um arquivo de blacklist que contém os nomes definidos, como mostra a imagem a seguir.

```
#/etc/squid/blacklist
facebook
netflix
youtube
httpforever
threads
```

- A linha **http_access allow redetf** irá permitir todas as outras requisições para a origem redetf
- A linha **http_access deny all** irá bloquear o resto de qualquer origem.

Após configuração do Squid, utilizamos alguns comandos para habilitar as edições feitas.

- Para recarregar o arquivo de configurações, utilizamos **sudo invoke-rc.d squid reload**
- Para verificação de erros no arquivo, utilizamos

sudo squid -k parse

- Enfim, para reiniciar o Squid, usamos **sudo invoke-rc.d squid restart**

Para teste, tentamos um wget na máquina cliente para um dos domínios bloqueados por ACL:

```
davi@davi-client:~$ wget uece.br
--2024-09-22 23:40:34-- http://uece.br/
Resolving uece.br (uece.br)... 54.232.162.15, 18.229.230.129
Connecting to uece.br (uece.br)|54.232.162.15|:80... connected.
HTTP request sent, awaiting response... 403 Forbidden
2024-09-22 23:40:35 ERROR 403: Forbidden.

davi@davi-client:~$ █
```

teste com item de blacklist:

```
davi@davi-client:~$ wget facebook.com
URL transformed to HTTPS due to an HSTS policy
--2024-09-23 01:01:02-- https://facebook.com/
Resolving facebook.com (facebook.com)... 157.240.216.35, 2a03:2880:f159:82:face:b00c:0:25de
Connecting to facebook.com (facebook.com)|157.240.216.35|:443... connected.
OpenSSL: error:0A00010B:SSL routines::wrong version number
Unable to establish SSL connection.
```

Configuração de ambiente DMZ

- Uma máquina Webserver é adicionada com uma nova interface Host-Only enp0s9 conectada ao Firewall. Nesta, instalamos e ativamos o Apache2. Para isso, clonamos a máquina cliente e adicionamos a interface Host-Only enp0s9, tanto no webserver quanto no Firewall.
- No Firewall, adicionamos as seguintes linhas ao script inicial para configuração da interface criada:

```
ip link set enp0s9 up
dhclient enp0s9
```

- No webserver, um script também é usado para definir a rota de acesso à rede externa via ip da interface Host-Only enp0s9 do Firewall:

```
davi@webserver:~$ cat init.sh
#!/bin/bash

ip route add default via 192.168.15.4
davi@webserver:~$ █
```

- Para prosseguir com a configuração do DMZ, adicionamos regras para permissão de tráfego http e https, além de tráfego icmp.

```
root@davi-firewall:/home/davi# iptables-save | grep 80 | grep enp0s9
-A FORWARD -i enp0s3 -o enp0s9 -p tcp -m multiport --dports 80,443 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

```
root@davi-firewall:/home/davi# iptables-save | grep icmp | grep enp0s9
-A FORWARD -i enp0s3 -o enp0s9 -p icmp -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
-A FORWARD -i enp0s9 -o enp0s3 -p icmp -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
root@davi-firewall:/home/davi#
```

teste:

```
davi@webserver:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=107 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=114 time=68.5 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 68.475/87.796/107.117/19.321 ms
davi@webserver:~$
```

- Após isso, instalamos o Apache2 com o comando **sudo apt install apache2**
- Para verificar, podemos ver seu status com o comando **sudo systemctl status apache2**

```
davi@webserver:~$ sudo systemctl status apache2
[sudo] password for davi:
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-09-23 01:09:00 UTC; 24min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 628 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Main PID: 835 (apache2)
    Tasks: 55 (limit: 2219)
   Memory: 7.6M
      CPU: 406ms
   CGroup: /system.slice/apache2.service
           └─835 /usr/sbin/apache2 -k start
             └─842 /usr/sbin/apache2 -k start
               └─843 /usr/sbin/apache2 -k start

Sep 23 01:08:52 webserver systemd[1]: Starting The Apache HTTP Server...
Sep 23 01:09:00 webserver apachectl[694]: AH00558: apache2: Could not reliably determine the server's fully c
Sep 23 01:09:00 webserver systemd[1]: Started The Apache HTTP Server.
lines 1-17/17 (END)
```

- Também podemos utilizar **wget localhost**. Isso demonstra que as regras http e https funcionam no dmz, sem ser afetado pelo proxy squid. Também demonstra o serviço Apache2 em execução:

```
davi@webserver:~$ wget localhost
--2024-09-23 01:34:18-- http://localhost/
Resolving localhost (localhost)... 127.0.0.1
Connecting to localhost (localhost)|127.0.0.1|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10671 (10K) [text/html]
Saving to: 'index.html'

index.html                                100%[=====]
2024-09-23 01:34:18 (377 MB/s) - 'index.html' saved [10671/10671]

davi@webserver:~$ █
```

- Para receber requisições da rede externa ao serviço Web, devemos fazer com que as requisições cheguem à interface Bridge enp0s3 e encaminhem para o IP do servidor Web. Para isso, adicionamos uma regra iptables de redirecionamento utilizando DNAT com o comando

```
sudo iptables -t nat -A PREROUTING -d 192.168.100.181/32 -p tcp -m tcp --dport 80 -j DNAT --to-destination 192.168.15.3:80
```

O primeiro endereço no comando representa a interface Bridge enp0s3, que receberá a requisição da rede externa e encaminhará para o endereço da interface Host-Only enp0s9 da máquina WebServer, para que chegue ao Servidor Web.

- Para finalizar, utilizamos a Máquina Local para representar a rede externa que irá fazer as requisições para dentro do ambiente de rede do VirtualBox. Como definido, o endereço do servidor está sendo representado por 192.168.100.181, a interface Bridge, enquanto seu endereço real é 192.168.15.3. Podemos testar o acesso ao servidor Apache2 através do browser da máquina local, com a URL:
http://<ip da interface enp0s3 firewall>

No caso, o acesso foi feito com **http://192.168.100.181**



CORREÇÕES FEITAS APÓS APRESENTAÇÃO

Máquina local de George Ricardo

WebServer - enp0s9 (192.168.210.6)

Permissão para ICMP de cliente para DMZ

```
-A FORWARD -i enp0s8 -o enp0s9 -p icmp -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT  
-A FORWARD -i enp0s9 -o enp0s8 -p icmp -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
```

teste:

```
cliente@cliente:~$ ping 192.168.210.6  
PING 192.168.210.6 (192.168.210.6) 56(84) bytes of data:  
64 bytes from 192.168.210.6: icmp_seq=1 ttl=63 time=1.78 ms  
64 bytes from 192.168.210.6: icmp_seq=2 ttl=63 time=1.26 ms  
64 bytes from 192.168.210.6: icmp_seq=3 ttl=63 time=3.87 ms  
64 bytes from 192.168.210.6: icmp_seq=4 ttl=63 time=14.2 ms  
64 bytes from 192.168.210.6: icmp_seq=5 ttl=63 time=2.20 ms  
64 bytes from 192.168.210.6: icmp_seq=6 ttl=63 time=2.12 ms  
64 bytes from 192.168.210.6: icmp_seq=7 ttl=63 time=2.08 ms  
64 bytes from 192.168.210.6: icmp_seq=8 ttl=63 time=2.16 ms  
^C  
--- 192.168.210.6 ping statistics ---  
8 packets transmitted, 8 received, 0% packet loss, time 7017ms  
rtt min/avg/max/mdev = 1.258/3.710/14.227/4.035 ms  
cliente@cliente:~$
```

Permissão de SSH do cliente para máquina remota

```
-A FORWARD -i enp0s8 -o enp0s9 -p tcp -m tcp --dport 22 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT  
-A FORWARD -i enp0s9 -o enp0s8 -p tcp -m tcp --sport 22 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT
```

teste:

```
cliente@cliente:~$ ssh cliente@192.168.210.6
The authenticity of host '192.168.210.6 (192.168.210.6)' can't be established.
ED25519 key fingerprint is SHA256:KuDnxrFpB1Eayrn1vV/t4ERrUWV0Pp/X6F0fYPFdBAQ.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.210.6' (ED25519) to the list of known hosts.
cliente@192.168.210.6's password:
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-40-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Manutenção de Segurança Expandida para Applications não está ativa.

3 as atualizações podem ser aplicadas imediatamente.
Para ver as actualizações adicionais corre o comando: apt list --upgradable

Ativar ESM Apps para poder receber possíveis futuras atualizações de segurança.
Consulte https://ubuntu.com/esm ou execute: sudo pro status

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Wed Sep 25 12:02:47 2024 from 192.168.163.6
cliente@web-server:~$
```