

Passos do Relatório

- 1 Configurar a rede IPv4 e IPv6.
- 2 Testar a comunicação com o comando ping.
- 3 Criar uma página Web com HTML e CSS com seus dados (matrícula e nome).
- 4 Testar o servidor Web utilizando telnet e curl (métodos GET, POST e HEAD).
- 5 Apresentar os códigos de status de respostas HTTP.
- 6 Alterar a porta do servidor Web de 80 para 8080.
- 7 Visualizar os arquivos de log.
- 8 Alterar a página de erro padrão (código 404).
- 9 Visualizar a comunicação HTTP no Wireshark.

Dados das Máquinas

Cliente

Sistema Operacional: debian-10.2.0-amd64-netinst

Socket de Rede 1: Rede Interna

Socket de Rede 2: NAT

Login: user

Senha: user

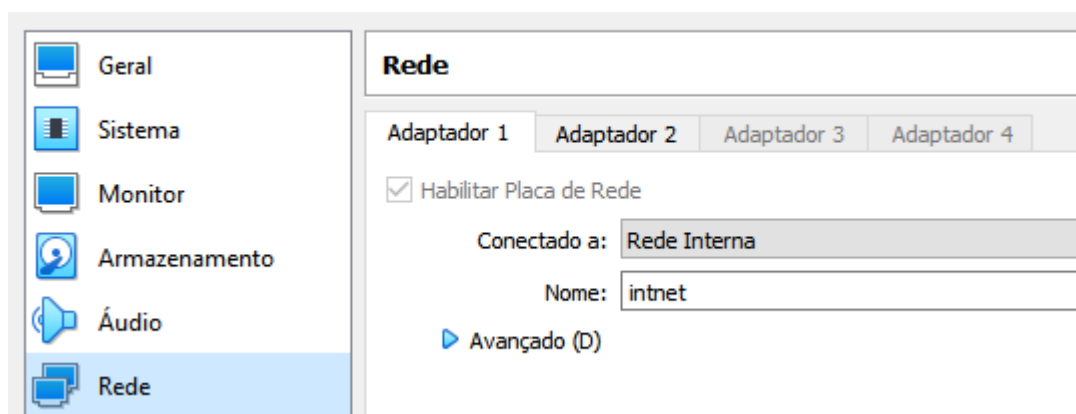
Senha do Superusuário: user

enp0s3: Rede Interna

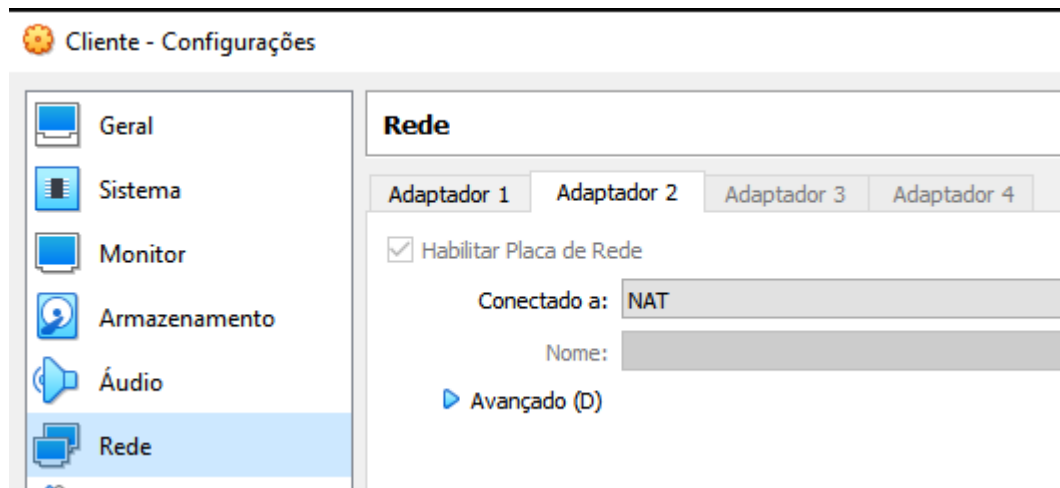
enp0s8: NAT

Primeiro adaptador do Cliente (enp0s3)

🔴 Cliente - Configurações



Segundo adaptador do Cliente (enp0s8)



Servidor

Sistema Operacional: debian-11.2.0-amd64-netinst

Socket de Rede 1: Rede Interna

Socket de Rede 2: NAT

Login: server

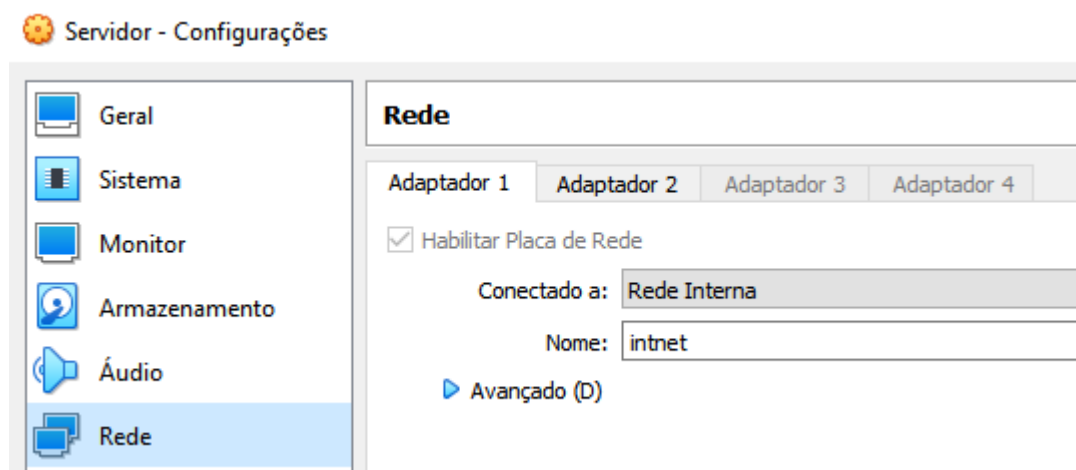
Senha: server

Senha do Superusuário: server

enp0s3: Rede Interna

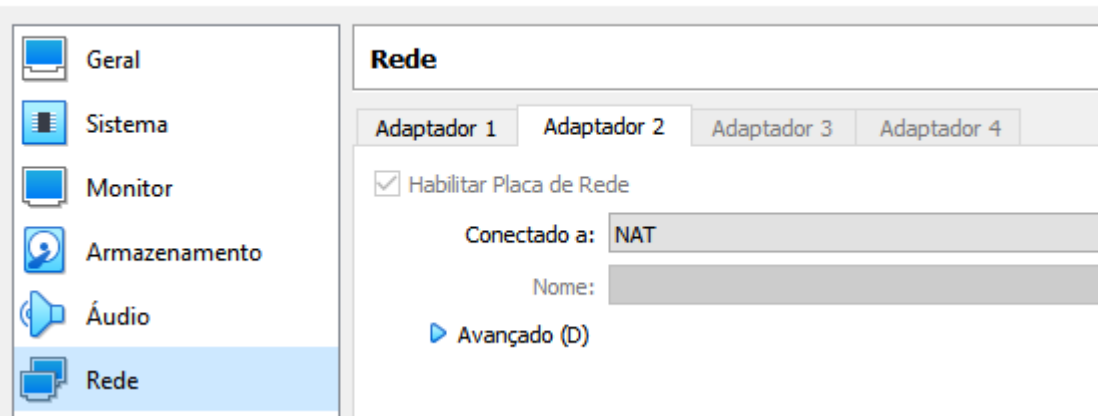
enp0s8: NAT

Primeiro adaptador do Servidor (enp0s3)



Segundo adaptador do Servidor (enp0s8)

Servidor - Configurações



1 Configurar a rede IPv4 e IPv6

Configuração realizada da rede IPv4 e IPv6 das duas máquinas e dos adaptadores.

1.1 Configuração do Cliente

IPv4 e IPv6 (dhcp) configurados da rede interna e IPv4 (dhcp) da rede NAT.

DHCP: fornece automaticamente um host ip (protocolo IP) com seu endereço IP.

Comando para acesso: `/etc/network/interfaces`; Comando para restart: `/etc/init.d/networking restart`

```
GNU nano 3.2 /etc/network/interfaces Modificado
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
allow-hotplug enp0s3
iface enp0s3 inet static
    address 10.0.0.9
    netmask 255.255.255.0

iface enp0s3 inet6 dhcp

auto enp0s8
allow-hotplug enp0s8
iface enp0s8 inet dhcp
```

Mostrar os endereços IPv4 e IPv6 da máquina

Comando para acesso: `ip address`

```
Cliente [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
root@user:/home/user# ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_f
000
    link/ether 08:00:27:7a:17:3c brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.9/24 brd 10.0.0.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe7a:173c/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_f
000
    link/ether 08:00:27:fe:d3:9b brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.15/24 brd 10.0.3.255 scope global dynamic enp0s8
        valid_lft 84936sec preferred_lft 84936sec
    inet6 fe80::a00:27ff:fefe:d39b/64 scope link
        valid_lft forever preferred_lft forever
```

1.2 Configuração do Servidor

IPv4 e IPv6 (dhcp) configurados da rede interna e IPv4 (dhcp) da rede NAT.

Comando para acesso: /etc/network/interfaces; Comando para restart: /etc/init.d/networking restart

```
GNU nano 3.2 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s3
allow-hotplug enp0s3
iface enp0s3 inet static
    address 10.0.0.10
    netmask 255.255.255.0

iface enp0s3 inet6 dhcp

auto enp0s8
allow-hotplug enp0s8
iface enp0s8 inet dhcp
```

Mostrar os endereços IPv4 e IPv6 da máquina

Comando para acesso: ip address

```

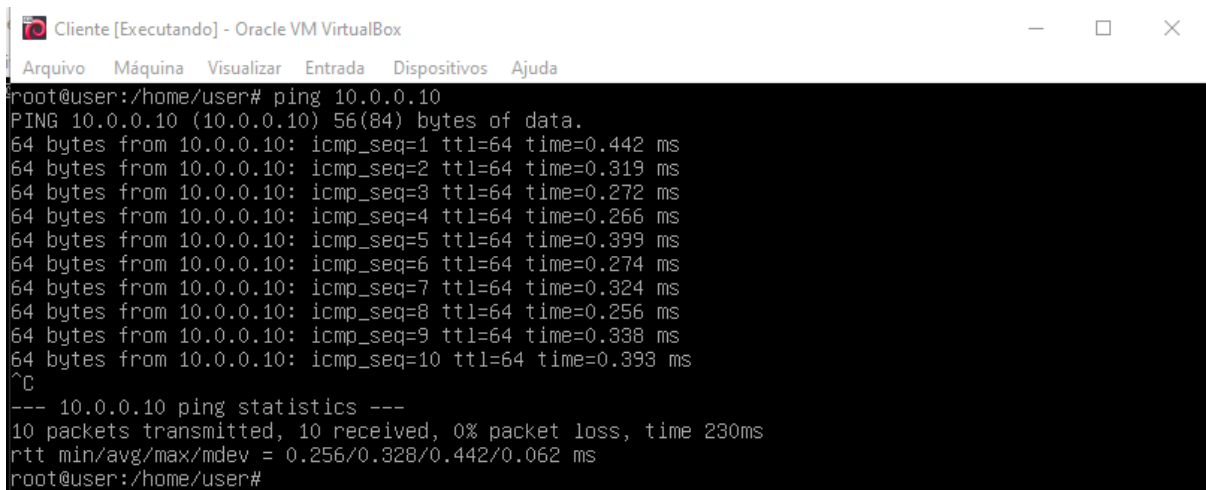
root@server:/data/www# ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_f
000
    link/ether 08:00:27:d2:08:f2 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.10/24 brd 10.0.0.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fed2:8f2/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_f
000
    link/ether 08:00:27:22:df:d9 brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.15/24 brd 10.0.3.255 scope global dynamic enp0s8
        valid_lft 85039sec preferred_lft 85039sec
    inet6 fe80::a00:27ff:fe22:dfd9/64 scope link
        valid_lft forever preferred_lft forever

```

2 Testar a comunicação com o comando ping

2.1 Comunicação Cliente-Servidor

Foram emitidos 10 pacotes para testar a comunicação, os 10 foram recebidos e nenhum foi perdido, com o tempo de 230 ms.



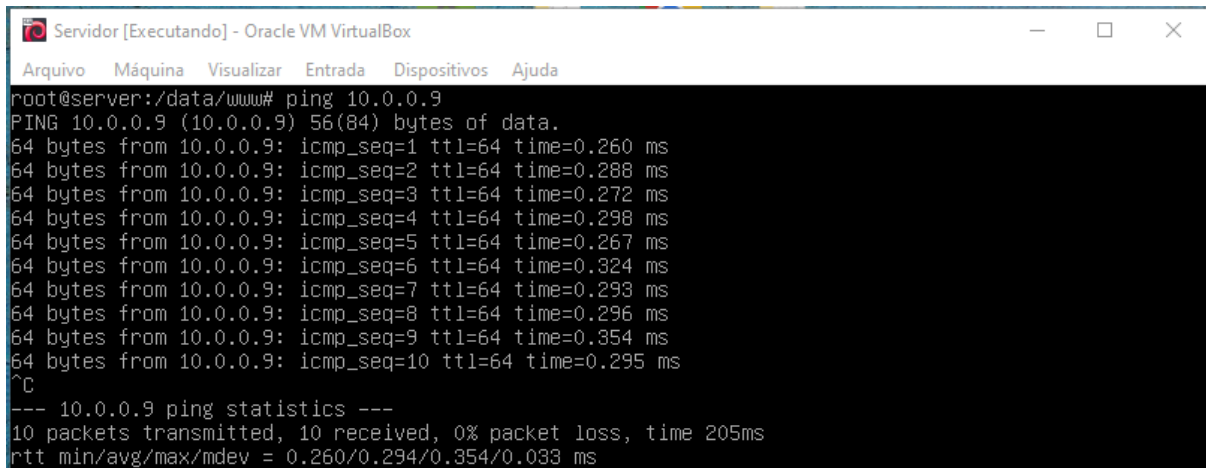
```

Cliente [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
root@user:/home/user# ping 10.0.0.10
PING 10.0.0.10 (10.0.0.10) 56(84) bytes of data.
64 bytes from 10.0.0.10: icmp_seq=1 ttl=64 time=0.442 ms
64 bytes from 10.0.0.10: icmp_seq=2 ttl=64 time=0.319 ms
64 bytes from 10.0.0.10: icmp_seq=3 ttl=64 time=0.272 ms
64 bytes from 10.0.0.10: icmp_seq=4 ttl=64 time=0.266 ms
64 bytes from 10.0.0.10: icmp_seq=5 ttl=64 time=0.399 ms
64 bytes from 10.0.0.10: icmp_seq=6 ttl=64 time=0.274 ms
64 bytes from 10.0.0.10: icmp_seq=7 ttl=64 time=0.324 ms
64 bytes from 10.0.0.10: icmp_seq=8 ttl=64 time=0.256 ms
64 bytes from 10.0.0.10: icmp_seq=9 ttl=64 time=0.338 ms
64 bytes from 10.0.0.10: icmp_seq=10 ttl=64 time=0.393 ms
^C
--- 10.0.0.10 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 230ms
rtt min/avg/max/mdev = 0.256/0.328/0.442/0.062 ms
root@user:/home/user#

```

2.2 Comunicação Servidor-Cliente

Foram emitidos 10 pacotes para testar a comunicação, os 10 foram recebidos e nenhum foi perdido, com o tempo de 205 ms.



```
Servidor [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
root@server:/data/www# ping 10.0.0.9
PING 10.0.0.9 (10.0.0.9) 56(84) bytes of data.
64 bytes from 10.0.0.9: icmp_seq=1 ttl=64 time=0.260 ms
64 bytes from 10.0.0.9: icmp_seq=2 ttl=64 time=0.288 ms
64 bytes from 10.0.0.9: icmp_seq=3 ttl=64 time=0.272 ms
64 bytes from 10.0.0.9: icmp_seq=4 ttl=64 time=0.298 ms
64 bytes from 10.0.0.9: icmp_seq=5 ttl=64 time=0.267 ms
64 bytes from 10.0.0.9: icmp_seq=6 ttl=64 time=0.324 ms
64 bytes from 10.0.0.9: icmp_seq=7 ttl=64 time=0.293 ms
64 bytes from 10.0.0.9: icmp_seq=8 ttl=64 time=0.296 ms
64 bytes from 10.0.0.9: icmp_seq=9 ttl=64 time=0.354 ms
64 bytes from 10.0.0.9: icmp_seq=10 ttl=64 time=0.295 ms
^C
--- 10.0.0.9 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 205ms
rtt min/avg/max/mdev = 0.260/0.294/0.354/0.033 ms
```

3 Criar uma página Web com HTML e CSS com seus dados (matrícula e nome).

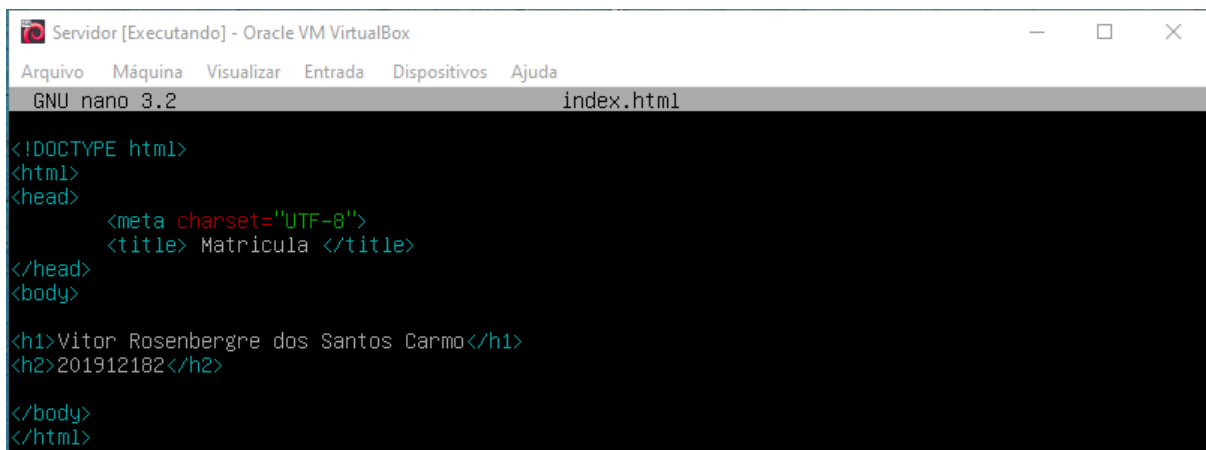
3.1 Criação da Página Web

Passos (comandos) para criação da página Web:

`mkdir /data/www` (diretórios)

`nano index.html` (criando o arquivo html, dentro da pasta)

Editando arquivo html com o nano:



```
Servidor [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
GNU nano 3.2                                index.html
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title> Matricula </title>
</head>
<body>

<h1>Vitor Rosenbergre dos Santos Carmo</h1>
<h2>201912182</h2>

</body>
</html>
```

3.2 Baixando e Configurando o Servidor Web nginx

Instalação do nginx:

`apt install nginx`

Status do servidor nginx:

`systemctl status nginx`

```

root@server:/data/www# systemctl status nginx
• nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2022-05-19 19:06:05 -03; 5h 29min ago
     Docs: man:nginx(8)
  Process: 1507 ExecReload=/usr/sbin/nginx -g daemon on; master_process on; -s reload (code=exited,
 Main PID: 459 (nginx)
    Tasks: 2 (limit: 1149)
   Memory: 14.2M
   CGroup: /system.slice/nginx.service
           └─ 459 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
              1508 nginx: worker process

mai 19 21:44:25 server systemd[1]: Reloading A high performance web server and a reverse proxy serve
mai 19 21:44:25 server systemd[1]: Reloaded A high performance web server and a reverse proxy server
mai 19 21:49:58 server systemd[1]: Reloading A high performance web server and a reverse proxy serve
mai 19 21:49:58 server systemd[1]: Reloaded A high performance web server and a reverse proxy server
mai 19 21:54:24 server systemd[1]: Reloading A high performance web server and a reverse proxy serve
mai 19 21:54:24 server systemd[1]: Reloaded A high performance web server and a reverse proxy server
mai 19 21:56:46 server systemd[1]: Reloading A high performance web server and a reverse proxy serve
mai 19 21:56:46 server systemd[1]: Reloaded A high performance web server and a reverse proxy server
mai 19 23:44:50 server systemd[1]: Reloading A high performance web server and a reverse proxy serve
mai 19 23:44:50 server systemd[1]: Reloaded A high performance web server and a reverse proxy server
lines 1-22/22 (END)

```

Configuração do servidor nginx:

Comando para acessar documento de configuração do servidor: nano
/etc/nginx/nginx.conf

Comando para atualizar: /etc/init.d/nginx reload

```

GNU nano 3.2 /etc/nginx/nginx.conf

user www-data;
worker_processes auto;
pid /run/nginx.pid;
include /etc/nginx/modules-enabled/*.conf;

events {
    worker_connections 768;
    # multi_accept on;
}

http {

    server {
        listen 10.0.0.10:80;
        location / {
            root /data/www;
        }
    }
    ##
    # Basic Settings
    ##

    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    keepalive_timeout 65;
    types_hash_max_size 2048;
    # server_tokens off;

    # server_names_hash_bucket_size 64;
    # server_name_in_redirect off;

    include /etc/nginx/mime.types;

```

listen - indica que o ip da placa de rede interna com ip 10.0.0.10 vai estar atendo a porta 80.

location - na requisição com / URL passa o documento da raiz root /data/www (a location aqui foi feita especialmente para o html, mas poderia ser adicionado outro bloco para outros arquivos, como por exemplo, imagens).

Os dados acima foram encontrados na página do nginx. URL - https://nginx.org/en/docs/beginners_guide.html

4 Testar o servidor Web utilizando telnet e curl (métodos GET, POST e HEAD).

O HTTP define um conjunto de métodos de requisição que indicam a ação desejada que vai ser performada a partir de uma informação dada. Esses métodos são considerados idempotente, ou seja, podem ser testados várias vezes e o resultado vai ser o mesmo.

Os principais métodos são os:

- GET - requisita a representação da informação requisitada.
- HEAD - pede por uma resposta equivalente a requisição do GET, no entanto, sem o corpo do objeto.
- POST - apresenta uma entidade para o arquivo requisitado. Ele pode causar uma mudança no estado ou algum outro efeito no servidor, por isso é muitas vezes declinado pelos servidores.

4.1 Telnet

O comando telnet testa a conectividade com uma porta de serviço. Através disso, pode ser identificado se há algum bloqueio de rede na porta especificada. Este protocolo já vem habilitado nas versões atuais do Debian.

4.1.1 GET

Comandos utilizados: telnet 10.0.0.10 80(porta)

GET / HTTP/1.1 (requisição, protocolo e versão)

HOST: www.data (local do arquivo)



```
Cliente [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
root@user:/home/user# telnet 10.0.0.10 80
Trying 10.0.0.10...
Connected to 10.0.0.10.
Escape character is '^]'.
GET / HTTP/1.1
Host: www.data

HTTP/1.1 200 OK
Server: nginx/1.18.0
Date: Fri, 20 May 2022 15:17:13 GMT
Content-Type: text/html
Content-Length: 178
Last-Modified: Fri, 20 May 2022 15:12:06 GMT
Connection: keep-alive
ETag: "6287afc6-b2"
Accept-Ranges: bytes

<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title> Matricula </title>
</head>
<body>

<h1>Vitor Rosenbergre dos Santos Carmo</h1>
<h2>201912182</h2>

</body>
</html>
^CConnection closed by foreign host.
root@user:/home/user# _
```

4.1.2 HEAD

Comandos utilizados: telnet 10.0.0.10 80(porta)

HEAD / HTTP/1.1 (requisição, protocolo e versão)

HOST: www.data (local do arquivo)

```
Cliente [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
root@user:/home/user# telnet 10.0.0.10 80
Trying 10.0.0.10...
Connected to 10.0.0.10.
Escape character is '^]'.
HEAD / HTTP/1.1
HOST: www.data

HTTP/1.1 200 OK
Server: nginx/1.18.0
Date: Fri, 20 May 2022 15:52:58 GMT
Content-Type: text/html
Content-Length: 178
Last-Modified: Fri, 20 May 2022 15:12:06 GMT
Connection: keep-alive
ETag: "6287afc6-b2"
Accept-Ranges: bytes
```

4.1.3 POST

Comandos utilizados: telnet 10.0.0.10 80(porta)
POST / HTTP/1.1 (requisição, protocolo e versão)
HOST: www.data (local do arquivo)

```
Cliente [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
root@user:/home/user# telnet 10.0.0.10 80
Trying 10.0.0.10...
Connected to 10.0.0.10.
Escape character is '^]'.
POST / HTTP/1.1
HOST: www.data

HTTP/1.1 405 Not Allowed
Server: nginx/1.18.0
Date: Fri, 20 May 2022 15:53:45 GMT
Content-Type: text/html
Content-Length: 157
Connection: keep-alive

<html>
<head><title>405 Not Allowed</title></head>
<body>
<center><h1>405 Not Allowed</h1></center>
<hr><center>nginx/1.18.0</center>
</body>
</html>
```

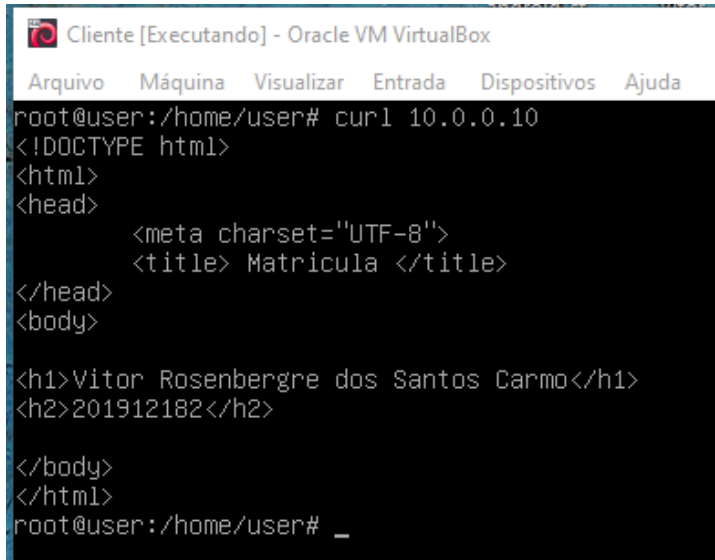
4.2 Curl

O Curl é uma ferramenta para transferir dados de/para um servidor, usando um dos protocolos suportados. Foi necessária a sua instalação no debian utilizando o comando: `sudo apt install curl`. Não há necessidade do usuário digitar o cabeçalho com o curl, tudo é feito na linha de comando com o URL.

Fonte dos comandos: <https://reqbin.com>

4.2.1 Método GET no Curl

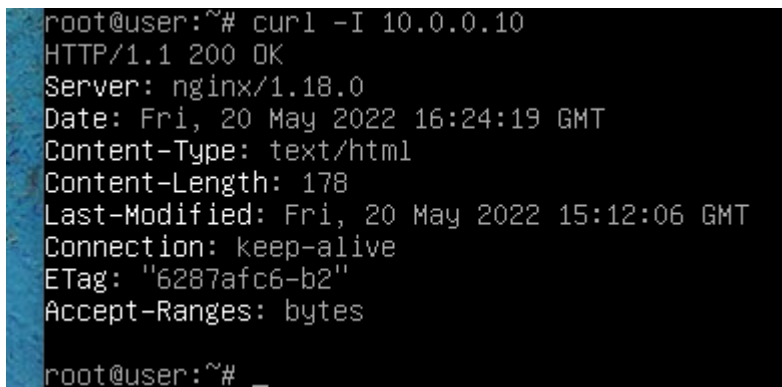
Comando: `curl 10.0.0.10`

A screenshot of a terminal window titled "Cliente [Executando] - Oracle VM VirtualBox". The window has a menu bar with "Arquivo", "Máquina", "Visualizar", "Entrada", "Dispositivos", and "Ajuda". The terminal shows the command `root@user:/home/user# curl 10.0.0.10` and its output, which is an HTML document. The output starts with `<!DOCTYPE html>`, followed by `<html>` and `<head>`. Inside the head, there is a `<meta charset="UTF-8">` and a `<title> Matricula </title>`. The body contains `<h1>Vitor Rosenbergre dos Santos Carmo</h1>` and `<h2>201912182</h2>`. The terminal ends with `root@user:/home/user# _`.

```
Cliente [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
root@user:/home/user# curl 10.0.0.10
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title> Matricula </title>
</head>
<body>
  <h1>Vitor Rosenbergre dos Santos Carmo</h1>
  <h2>201912182</h2>
</body>
</html>
root@user:/home/user# _
```

4.2.2 Método HEAD no Curl

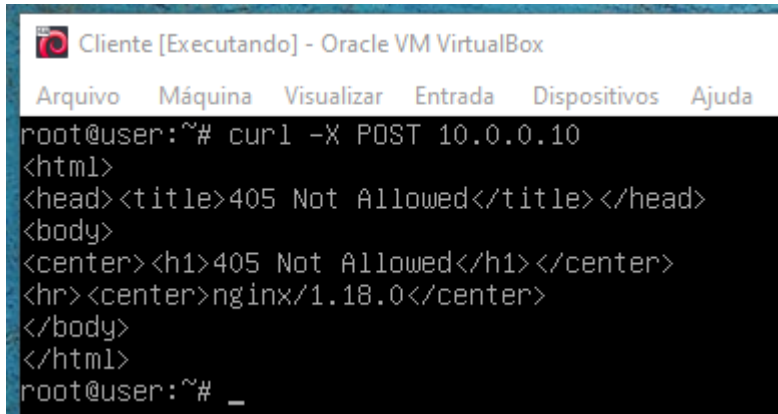
Comando: `curl -I 10.0.0.10`

A screenshot of a terminal window showing the output of the command `root@user:~# curl -I 10.0.0.10`. The output displays HTTP headers: `HTTP/1.1 200 OK`, `Server: nginx/1.18.0`, `Date: Fri, 20 May 2022 16:24:19 GMT`, `Content-Type: text/html`, `Content-Length: 178`, `Last-Modified: Fri, 20 May 2022 15:12:06 GMT`, `Connection: keep-alive`, `ETag: "6287afc6-b2"`, and `Accept-Ranges: bytes`. The terminal ends with `root@user:~# _`.

```
root@user:~# curl -I 10.0.0.10
HTTP/1.1 200 OK
Server: nginx/1.18.0
Date: Fri, 20 May 2022 16:24:19 GMT
Content-Type: text/html
Content-Length: 178
Last-Modified: Fri, 20 May 2022 15:12:06 GMT
Connection: keep-alive
ETag: "6287afc6-b2"
Accept-Ranges: bytes
root@user:~# _
```

4.2.3 Método POST no Curl

Comando: curl -X POST 10.0.0.10

A screenshot of a terminal window titled 'Cliente [Executando] - Oracle VM VirtualBox'. The window has a menu bar with 'Arquivo', 'Máquina', 'Visualizar', 'Entrada', 'Dispositivos', and 'Ajuda'. The terminal shows a root user at a prompt '~#'. The user enters the command 'curl -X POST 10.0.0.10'. The output is an HTML response: '<html>\n<head><title>405 Not Allowed</title></head>\n<body>\n<center><h1>405 Not Allowed</h1></center>\n<hr><center>nginx/1.18.0</center>\n</body>\n</html>'. The prompt returns to '~# _'.

5 Apresentar os códigos de status de respostas HTTP

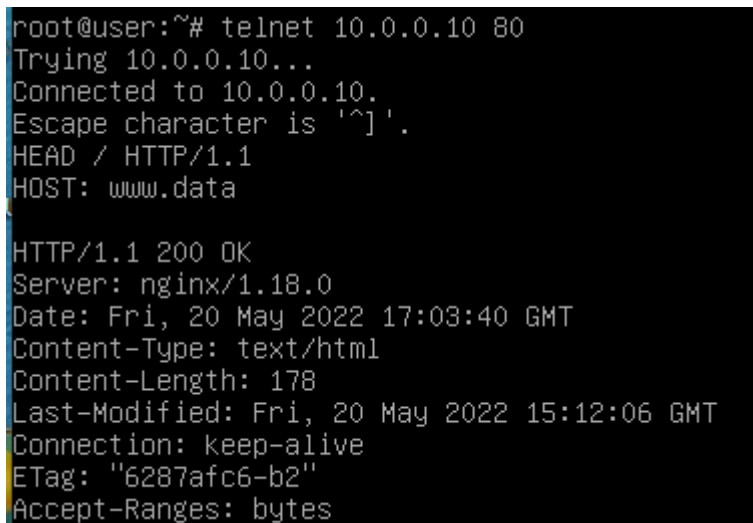
A codificação de status e a frase associada a ela indicam o resultado da requisição. Eis algumas codificações de status e frases associadas comuns:

- 200 OK - a requisição foi bem-sucedida e a informação volta para a resposta.
- 400 Bad Request - uma codificação genérica de erro mostrando que a requisição não pode ser entendida pelo servidor.
- 403 Forbidden - O cliente não tem direitos de acesso ao conteúdo, portanto o servidor está rejeitando dar a resposta.
- 404 Not Found - o documento requisitado não existe no servidor.
- 405 Not Allowed - acessa o servidor mas a requisição não é permitida.

5.1 200 OK

Comando: telnet 10.0.0.10 80

HEAD / HTTP/1.1

A screenshot of a terminal window showing the output of a telnet command. The prompt is 'root@user:~#'. The user enters 'telnet 10.0.0.10 80'. The output is: 'Trying 10.0.0.10...\nConnected to 10.0.0.10.\nEscape character is '^]'.\nHEAD / HTTP/1.1\nHOST: www.data\n\nHTTP/1.1 200 OK\nServer: nginx/1.18.0\nDate: Fri, 20 May 2022 17:03:40 GMT\nContent-Type: text/html\nContent-Length: 178\nLast-Modified: Fri, 20 May 2022 15:12:06 GMT\nConnection: keep-alive\nETag: "6287afc6-b2"\nAccept-Ranges: bytes'.

5.2 400 Bad Request

Comando: telnet 10.0.0.10 80

GET / HTTP/0.9

Versão errada -> Erro.

```
root@user:~# telnet 10.0.0.10 80
Trying 10.0.0.10...
Connected to 10.0.0.10.
Escape character is '^]'.
GET / HTTP/0.9
HTTP/1.1 400 Bad Request
Server: nginx/1.18.0
Date: Fri, 20 May 2022 17:10:10 GMT
Content-Type: text/html
Content-Length: 157
Connection: close

<html>
<head><title>400 Bad Request</title></head>
<body>
<center><h1>400 Bad Request</h1></center>
<hr><center>nginx/1.18.0</center>
</body>
</html>
Connection closed by foreign host.
root@user:~#
```

5.3 403 Forbidden

Comando: telnet 10.0.0.10 80

GET / HTTP/1.1

Erro ocorreu no HOST: www.data, já que eu alterei a configuração para não aceitar requisições.

```
root@user:~# telnet 10.0.0.10 80
Trying 10.0.0.10...
Connected to 10.0.0.10.
Escape character is '^]'.
GET / HTTP/1.1
HOST: www.data

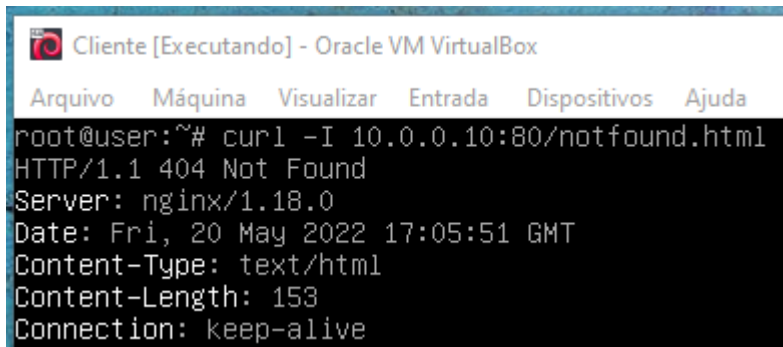
HTTP/1.1 403 Forbidden
Server: nginx/1.18.0
Date: Fri, 20 May 2022 17:17:27 GMT
Content-Type: text/html
Content-Length: 153
Connection: keep-alive

<html>
<head><title>403 Forbidden</title></head>
<body>
<center><h1>403 Forbidden</h1></center>
<hr><center>nginx/1.18.0</center>
</body>
</html>
```

5.4 404 Not Found

Comandos: `curl -I 10.0.0.10:80/notfound.html`

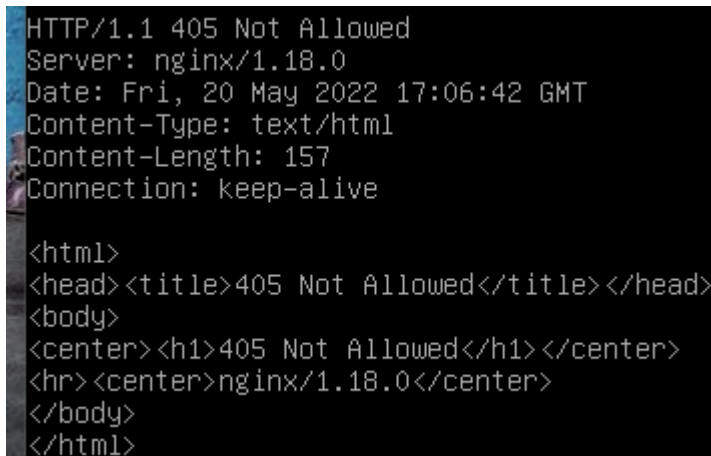
Não encontrou o arquivo “notfound.html” no servidor.



```
Cliente [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
root@user:~# curl -I 10.0.0.10:80/notfound.html
HTTP/1.1 404 Not Found
Server: nginx/1.18.0
Date: Fri, 20 May 2022 17:05:51 GMT
Content-Type: text/html
Content-Length: 153
Connection: keep-alive
```

5.5 405 Not Allowed

Comando POST não foi ativado para requisição.



```
HTTP/1.1 405 Not Allowed
Server: nginx/1.18.0
Date: Fri, 20 May 2022 17:06:42 GMT
Content-Type: text/html
Content-Length: 157
Connection: keep-alive

<html>
<head><title>405 Not Allowed</title></head>
<body>
<center><h1>405 Not Allowed</h1></center>
<hr><center>nginx/1.18.0</center>
</body>
</html>
```

6 Alterar a porta do servidor Web de 80 para 8080.

A porta de escuta vai ser alterada para 8080, e assim os testes vão ser feitos.

6.1 Alterando a porta nas configurações

Comando de acesso:

`nano /etc/nginx/nginx.conf`

Mudança na porta de escuta do servidor para 8080:

```
http {  
    server {  
        # Mudança da porta de escuta de 80 para 8080  
        listen 10.0.0.10:8080;  
        location / {  
            root /data/www;  
        }  
    }  
}
```

Comando para atualização do servidor com a nova porta:
/etc/init.d/nginx reload

6.2 Teste da nova porta 8080

Tentativa da requisição HEAD por meio da porta 80.

Comando do teste: curl -I 10.0.0.10:80/index.html

```
root@user:~# curl -I 10.0.0.10:80/index.html  
HTTP/1.1 404 Not Found  
Server: nginx/1.18.0  
Date: Fri, 20 May 2022 17:47:15 GMT  
Content-Type: text/html  
Content-Length: 153  
Connection: keep-alive  
  
root@user:~#
```

Requisição HEAD por meio da porta 8080.

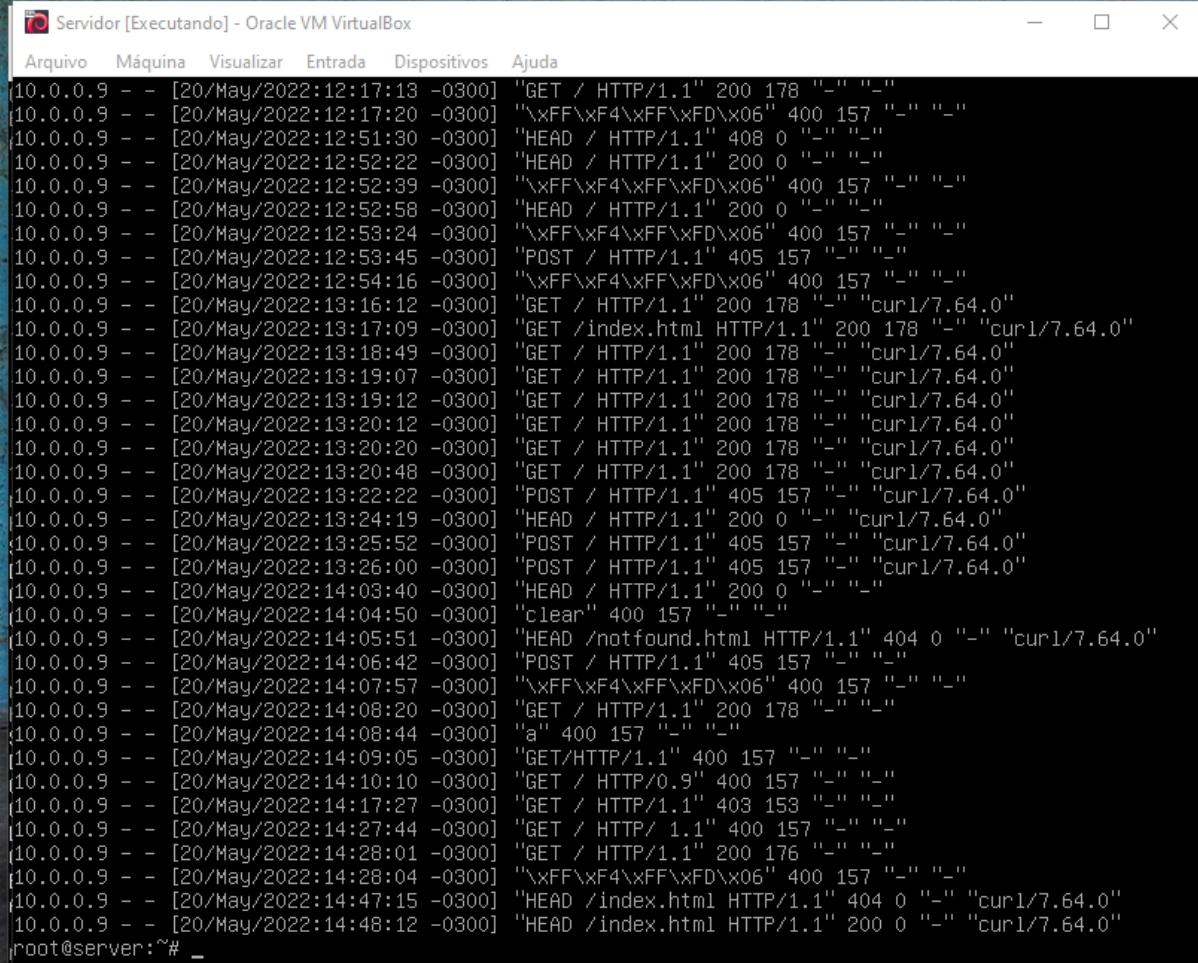
Comando do teste: curl -I 10.0.0.10:8080/index.html

```
root@user:~# curl -I 10.0.0.10:80/index.html  
HTTP/1.1 404 Not Found  
Server: nginx/1.18.0  
Date: Fri, 20 May 2022 17:47:15 GMT  
Content-Type: text/html  
Content-Length: 153  
Connection: keep-alive  
  
root@user:~# curl -I 10.0.0.10:8080/index.html  
HTTP/1.1 200 OK  
Server: nginx/1.18.0  
Date: Fri, 20 May 2022 17:48:12 GMT  
Content-Type: text/html  
Content-Length: 176  
Last-Modified: Fri, 20 May 2022 17:26:28 GMT  
Connection: keep-alive  
ETag: "6287cf44-b0"  
Accept-Ranges: bytes  
  
root@user:~#
```

O Servidor está escutando a porta 8080, então qualquer requisição agora deve ser feita nesta porta.

7 Visualizar os arquivos de log

A visualização do log vai ser dada utilizando o cat, que é usado aqui para exibir (concatenar) os conteúdos dos registros. Os logs estão no diretório /var/log/nginx/access.log . O comando utilizado para exibição: cat /var/log/nginx/access.log.



```
Servidor [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
10.0.0.9 - - [20/May/2022:12:17:13 -0300] "GET / HTTP/1.1" 200 178 "-" "-"
10.0.0.9 - - [20/May/2022:12:17:20 -0300] "\xFF\xF4\xFF\xFD\x06" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:12:51:30 -0300] "HEAD / HTTP/1.1" 408 0 "-" "-"
10.0.0.9 - - [20/May/2022:12:52:22 -0300] "HEAD / HTTP/1.1" 200 0 "-" "-"
10.0.0.9 - - [20/May/2022:12:52:39 -0300] "\xFF\xF4\xFF\xFD\x06" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:12:52:58 -0300] "HEAD / HTTP/1.1" 200 0 "-" "-"
10.0.0.9 - - [20/May/2022:12:53:24 -0300] "\xFF\xF4\xFF\xFD\x06" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:12:53:45 -0300] "POST / HTTP/1.1" 405 157 "-" "-"
10.0.0.9 - - [20/May/2022:12:54:16 -0300] "\xFF\xF4\xFF\xFD\x06" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:13:16:12 -0300] "GET / HTTP/1.1" 200 178 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:17:09 -0300] "GET /index.html HTTP/1.1" 200 178 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:18:49 -0300] "GET / HTTP/1.1" 200 178 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:19:07 -0300] "GET / HTTP/1.1" 200 178 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:19:12 -0300] "GET / HTTP/1.1" 200 178 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:20:12 -0300] "GET / HTTP/1.1" 200 178 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:20:20 -0300] "GET / HTTP/1.1" 200 178 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:20:48 -0300] "GET / HTTP/1.1" 200 178 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:22:22 -0300] "POST / HTTP/1.1" 405 157 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:24:19 -0300] "HEAD / HTTP/1.1" 200 0 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:25:52 -0300] "POST / HTTP/1.1" 405 157 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:13:26:00 -0300] "POST / HTTP/1.1" 405 157 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:14:03:40 -0300] "HEAD / HTTP/1.1" 200 0 "-" "-"
10.0.0.9 - - [20/May/2022:14:04:50 -0300] "clear" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:14:05:51 -0300] "HEAD /notfound.html HTTP/1.1" 404 0 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:14:06:42 -0300] "POST / HTTP/1.1" 405 157 "-" "-"
10.0.0.9 - - [20/May/2022:14:07:57 -0300] "\xFF\xF4\xFF\xFD\x06" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:14:08:20 -0300] "GET / HTTP/1.1" 200 178 "-" "-"
10.0.0.9 - - [20/May/2022:14:08:44 -0300] "a" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:14:09:05 -0300] "GET/HTTP/1.1" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:14:10:10 -0300] "GET / HTTP/0.9" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:14:17:27 -0300] "GET / HTTP/1.1" 403 153 "-" "-"
10.0.0.9 - - [20/May/2022:14:27:44 -0300] "GET / HTTP/ 1.1" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:14:28:01 -0300] "GET / HTTP/1.1" 200 176 "-" "-"
10.0.0.9 - - [20/May/2022:14:28:04 -0300] "\xFF\xF4\xFF\xFD\x06" 400 157 "-" "-"
10.0.0.9 - - [20/May/2022:14:47:15 -0300] "HEAD /index.html HTTP/1.1" 404 0 "-" "curl/7.64.0"
10.0.0.9 - - [20/May/2022:14:48:12 -0300] "HEAD /index.html HTTP/1.1" 200 0 "-" "curl/7.64.0"
root@server:~#
```

O erro foi mostrado utilizando o comando cat /var/log/nginx/error.log :


```

root@server:~# cat /var/log/nginx/error.log
2022/05/20 12:12:52 [notice] 1106#1106: using inherited sockets from "6;7;"
2022/05/20 12:14:20 [notice] 1156#1156: signal process started
2022/05/20 12:16:29 [notice] 1175#1175: signal process started
2022/05/20 14:05:51 [error] 1176#1176: *19 open() "/data/www/notfound.html" failed (2: No such file or directory), client: 10.0.0.9, server: , request: "HEAD /notfound.html HTTP/1.1", host: "10.0.0.10"
2022/05/20 14:17:27 [error] 1176#1176: *24 directory index of "/data/www/" is forbidden, client: 10.0.0.9, server: , request: "GET / HTTP/1.1", host: "www.data"
2022/05/20 14:27:28 [notice] 1374#1374: signal process started
2022/05/20 14:42:20 [notice] 1400#1400: signal process started
root@server:~#


```

8 Alterar a página de erro padrão (código 404).

A alteração da página error_page 404 foi feita utilizando o tutorial: <https://techexpert.tips/pt-br/nginx-pt-br/nginx-redirecionar-o-erro-404-para-uma-pagina/#:~:text=Tutorial%20Nginx%20-%20Redirecione%20o%20erro,Instale%20o%20servidor%20Nginx.&text=Edite%20o%20arquivo%20de%20configura%C3%A7%C3%A3o%20Nginx%20para%20o%20site%20padr%C3%A3o.&text=Adicione%20a%20segu%C3%ADnte%20linha%20ao%20arquivo%20de%20configura%C3%A7%C3%A3o.&text=Aqui%20est%C3%A1%20o%20arquivo%20C%20antes%20de%20nossa%20configura%C3%A7%C3%A3o>. Ela vai redirecionar o visualização da página para a página html criada, chamada error404.html.

8.1 Criando a nova pagina código 404 (error404.html)

No diretório /data/www, vai ser criado a página de erro, com o comando nano error404.html . A página é representada abaixo:



```

Servidor [Executando] - Oracle VM VirtualBox
Arquivo  Máquina  Visualizar  Entrada  Dispositivos  Ajuda
GNU nano 5.4                                     error404.html
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title> ERRO 404 - NOT FOUND</title>
</head>
<body>
    <h1> O DOCUMENTO REQUISITADO NAO EXISTE NO SERVIDOR </h1>
</body>
</html>

```

8.2 Configurando o servidor para redirecionar a página código 404

Dentro do arquivo de configuração do servidor, foi adicionado o comando error_page 404 /error404.html; no bloco do server. Como mostrado no tutorial, o comando poderia também ser inserido no bloco location, já que estaria dentro do server.

Comandos:

nano /etc/nginx/nginx.conf (acessar arquivo conf)
error_page 404 /error404.html (redirecionar o erro para a página error404.html)
/etc/init.d/nginx reload (atualiza as mudanças no servidor)

```
http {  
    server {  
        # O servidor escuta a porta 80  
        listen 10.0.0.10:80;  
        error_page 404 /error404.html;  
        location / {  
            root /data/www;  
        }  
    }  
  
    ##  
    # Basic Settings  
    ##  
  
    sendfile on;  
    tcp_nopush on;  
    types_hash_max_size 2048;  
    # server_tokens off;  
  
    # server_names_hash_bucket_size 64;  
    # server_name_in_redirect off;  
  
    include /etc/nginx/mime.types;  
  
root@server:/data/www# /etc/init.d/nginx reload  
Reloading nginx configuration (via systemctl): nginx.service.
```

8.3 Teste do erro código 404

Teste realizado utilizando a requisição GET do curl, comando: curl 10.0.0.10:80/404error.html

```
root@user:~# curl 10.0.0.10:80/404error.html  
<!DOCTYPE html>  
<html>  
<head>  
    <meta charset="UTF-8">  
    <title> ERRO 404 - NOT FOUND</title>  
</head>  
<body>  
  
<h1> O DOCUMENTO REQUISITADO NAO EXISTE NO SERVIDOR </h1>  
  
</body>  
</html>  
root@user:~# _
```

A página foi redirecionada com sucesso

9 Visualizar a comunicação HTTP no Wireshark

Para visualizar a comunicação HTTP entre o servidor e o cliente, eu tive que criar um novo cliente com o IPv4 10.0.0.8 para baixar o wireshark e fazer a análise.

9.1 Informações da Máquina e do Cliente

Sistema Operacional: debian-11.2.0-amd64-netinst

Socket de Rede 1: Rede Interna

Socket de Rede 2: NAT

Login: wire

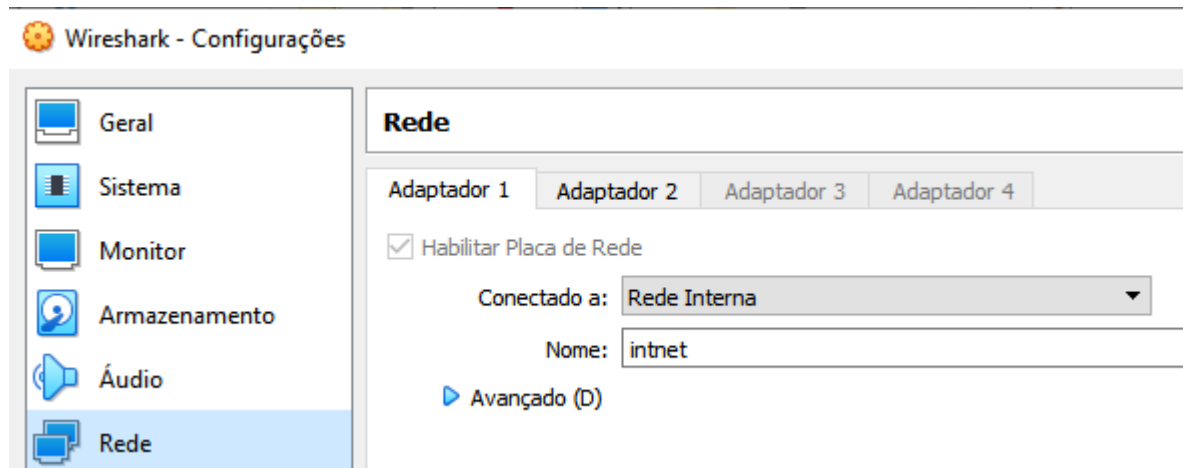
Senha: wire

Senha do Superusuário: wire

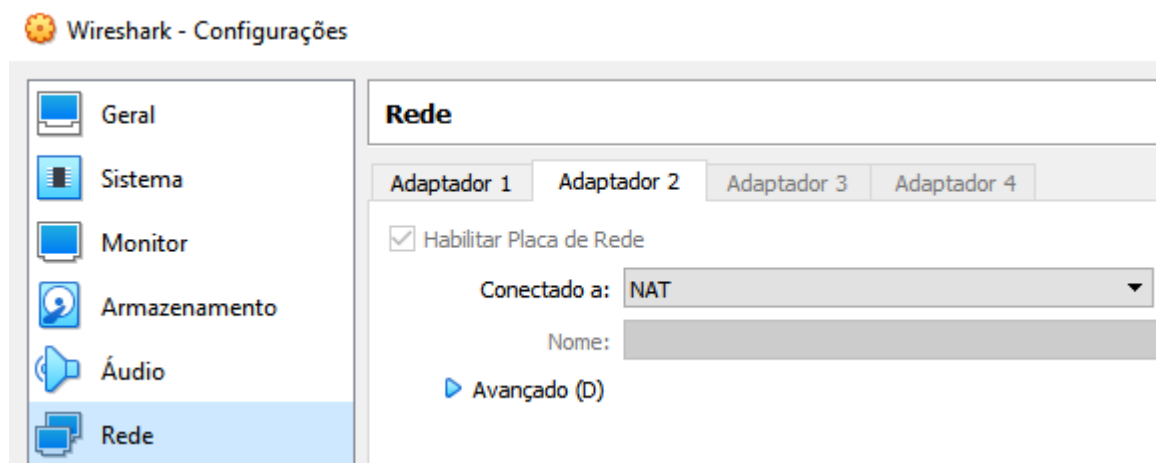
enp0s3: Rede Interna

enp0s8: NAT

Primeiro adaptador do Cliente (enp0s3)



Segundo adaptador do Cliente (enp0s8)



IPv4 e IPv6 (dhcp) configurados da rede interna e IPv4 (dhcp) da rede NAT.

DHCP: fornece automaticamente um host ip (protocolo IP) com seu endereço IP.

Comando para acesso: /etc/network/interfaces; Comando para restart: /etc/init.d/networking restart

```
wire@wireshark: ~  
GNU nano 5.4 /etc/network/interfaces *  
# This file describes the network interfaces available on your system  
# and how to activate them. For more information, see interfaces(5).  
  
source /etc/network/interfaces.d/*  
  
# The loopback network interface  
auto lo  
iface lo inet loopback  
  
auto enp0s3  
allow-hotplug enp0s3  
iface enp0s3 inet static  
    address 10.0.0.8  
    netmask 255.255.255.0  
  
iface enp0s8 inet6 dhcp  
  
auto enp0s8  
allow-hotplug enp0s8  
iface enp0s8 inet dhcp  
  
^G Ajuda      ^O Gravar    ^W Onde está? ^K Recortar   ^T Executar   ^C Local  
^X Sair      ^R Ler o arq ^\ Substituir ^U Colar      ^J Justificar ^_ Ir p/ linha
```

Mostrar os endereços IPv4 e IPv6 da máquina

Comando para acesso: ip address

```
wire@wireshark: ~  
root@wireshark:/home/wire# ip address  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP  
    group default qlen 1000  
    link/ether 08:00:27:2a:e7:1e brd ff:ff:ff:ff:ff:ff  
    inet 10.0.0.8/24 brd 10.0.0.255 scope global enp0s3  
        valid_lft forever preferred_lft forever  
    inet6 fe80::a00:27ff:fe2a:e71e/64 scope link  
        valid_lft forever preferred_lft forever  
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP  
    group default qlen 1000  
    link/ether 08:00:27:81:63:27 brd ff:ff:ff:ff:ff:ff  
    inet 10.0.3.15/24 brd 10.0.3.255 scope global dynamic enp0s8  
        valid_lft 85187sec preferred_lft 85187sec  
    inet6 fe80::a00:27ff:fe81:6327/64 scope link  
        valid_lft forever preferred_lft forever  
root@wireshark:/home/wire#
```

9.2 Informações sobre o Wireshark

Áreas do Wireshark (usando a rede interna):

The screenshot shows the Wireshark interface with the following components and annotations:

- Packet List:** A table showing captured packets. A red box highlights the first five packets, with a label "Chegada dos Pacotes" (Arrival of Packets) pointing to it.
- Packet Details:** A tree view showing the structure of the selected packet (Frame 1). A green box highlights the details of the selected packet, with a label "Detalhes do Pacote Selecionado" (Selected Packet Details) pointing to it.
- Packet Bytes:** A hex dump of the selected packet. A purple box highlights the hex dump, with a label "Informação do Pacote em Hexadecimal" (Hexadecimal Packet Information) pointing to it.

| Protocol | Length | Info |
|----------|--------|--|
| TCP | 74 | 38164 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T... |
| TCP | 74 | 80 → 38164 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA... |
| TCP | 66 | 38164 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1368244562... |
| HTTP | 139 | GET / HTTP/1.1 |
| TCP | 66 | 80 → 38164 [ACK] Seq=1 Ack=74 Win=65152 Len=0 TSval=101381309... |
| HTTP | 479 | HTTP/1.1 200 OK (text/html) |

Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface
Ethernet II, Src: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e), Dst: PcsCompu_d2:08:f2
Internet Protocol Version 4, Src: 10.0.0.8, Dst: 10.0.0.10
Transmission Control Protocol, Src Port: 38164, Dst Port: 80, Seq: 0, Len: 0

```
0000  08 00 27 d2 08 f2 08 00 27 2a e7 1e 08 00 45 00  ..'....'..*...E.
0010  00 3c 3b 1d 40 00 40 06 eb 8d 0a 00 00 08 0a 00  .<;.@.@.....
0020  00 0a 95 14 00 50 42 7c e2 2a 00 00 00 00 a0 02  ....PB|..*....
0030  fa f0 14 40 00 00 02 04 05 b4 04 02 08 0a 51 8d  ...@.....Q.
0040  c1 52 00 00 00 00 01 03 03 07                    .R.....
```

enp0s3: <live capture in progress> Packets: 70 · Displayed: 70 (100.0%) Profile: Default

Informações dos campos que o Wireshark mostra:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|--------------|------------------------|-------------------|----------|--------|------------------------------------|
| 1 | 0.000000000 | 10.0.0.8 | 10.0.0.10 | TCP | 74 | 38164 → 80 [SYN] Seq=0 Win=64240 |
| 2 | 0.000394999 | 10.0.0.10 | 10.0.0.8 | TCP | 74 | 80 → 38164 [SYN, ACK] Seq=0 Ack=1 |
| 3 | 0.000423361 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38164 → 80 [ACK] Seq=1 Ack=1 Win= |
| 4 | 0.000522755 | 10.0.0.8 | 10.0.0.10 | HTTP | 139 | GET / HTTP/1.1 |
| 5 | 0.000731129 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38164 [ACK] Seq=1 Ack=74 Win= |
| 6 | 0.000934044 | 10.0.0.10 | 10.0.0.8 | HTTP | 479 | HTTP/1.1 200 OK (text/html) |
| 7 | 0.000944924 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38164 → 80 [ACK] Seq=74 Ack=414 W |
| 8 | 0.001781198 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38164 → 80 [FIN, ACK] Seq=74 Ack= |
| 9 | 0.002004791 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38164 [FIN, ACK] Seq=414 Ack |
| 10 | 0.002015570 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38164 → 80 [ACK] Seq=75 Ack=415 W |
| 11 | 5.121368906 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | Who has 10.0.0.8? Tell 10.0.0.10 |
| 12 | 5.121387661 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | 10.0.0.8 is at 08:00:27:2a:e7:1e |
| 13 | 5.207539619 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | Who has 10.0.0.10? Tell 10.0.0.8 |
| 14 | 5.207764555 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | 10.0.0.10 is at 08:00:27:d2:08:f2 |
| 15 | 35.927764159 | fe80::a00:27ff:fe2a... | ff02::2 | ICMPv6 | 70 | Router Solicitation from 08:00:27 |
| 16 | 35.954068266 | 10.0.0.8 | 10.0.0.10 | TCP | 74 | 38166 → 80 [SYN] Seq=0 Win=64240 |
| 17 | 35.954542755 | 10.0.0.10 | 10.0.0.8 | TCP | 74 | 80 → 38166 [SYN, ACK] Seq=0 Ack=1 |
| 18 | 35.954575446 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38166 → 80 [ACK] Seq=1 Ack=1 Win= |
| 19 | 40.960018805 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | Who has 10.0.0.8? Tell 10.0.0.10 |
| 20 | 40.960037269 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | 10.0.0.8 is at 08:00:27:2a:e7:1e |
| 21 | 43.402938034 | 10.0.0.8 | 10.0.0.10 | TCP | 82 | 38166 → 80 [PSH, ACK] Seq=1 Ack=1 |
| 22 | 43.403297481 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38166 [ACK] Seq=1 Ack=17 Win= |
| 23 | 50.875285408 | 10.0.0.8 | 10.0.0.10 | TCP | 83 | GET / HTTP/1.1 [TCP segment of a |
| 24 | 50.875633635 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38166 [ACK] Seq=1 Ack=34 Win= |
| 25 | 51.905088584 | 10.0.0.8 | 10.0.0.10 | HTTP | 68 | GET / HTTP/1.1 |
| 26 | 51.905376419 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38166 [ACK] Seq=1 Ack=36 Win= |
| 27 | 51.905562044 | 10.0.0.10 | 10.0.0.8 | HTTP | 479 | HTTP/1.1 200 OK (text/html) |

9.3 Requisições e as respectivas ativações no Wireshark

9.3.1 Primeira ativação

Utilizando o curl para fazer uma requisição GET:

```
wire@wireshark: ~
root@wireshark:/home/wire# curl 10.0.0.10:80
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title> Matricula</title>
</head>
<body>

<h1>Vitor Rosenbergre dos Santos Carmo</h1>
<h2>201912182</h2>
</body>
</html>
root@wireshark:/home/wire#
```

Detecção da comunicação no Wireshark:

| Apply a display filter ... <Ctrl-/> | | | | | | |
|---|-------------------------|-------------------------|-------------------|--------|---|--|
| Time | Source | Destination | Protocol | Length | Info | |
| 0.000000000 | 10.0.0.8 | 10.0.0.10 | TCP | 74 | 38172 → 80 [SYN] Seq=0 Win=64240 Len=0 MS | |
| 0.000414111 | 10.0.0.10 | 10.0.0.8 | TCP | 74 | 80 → 38172 [SYN, ACK] Seq=0 Ack=1 Win=651 | |
| 0.000442525 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38172 → 80 [ACK] Seq=1 Ack=1 Win=64256 Le | |
| 0.000535932 | 10.0.0.8 | 10.0.0.10 | HTTP | 139 | GET / HTTP/1.1 | |
| 0.000688480 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38172 [ACK] Seq=1 Ack=74 Win=65152 L | |
| 0.000949624 | 10.0.0.10 | 10.0.0.8 | HTTP | 479 | HTTP/1.1 200 OK (text/html) | |
| 0.000960725 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38172 → 80 [ACK] Seq=74 Ack=414 Win=64128 | |
| 0.001897736 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38172 → 80 [FIN, ACK] Seq=74 Ack=414 Win= | |
| 0.002096361 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38172 [FIN, ACK] Seq=414 Ack=75 Win= | |
| 0.002109025 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38172 → 80 [ACK] Seq=75 Ack=415 Win=64128 | |
| ▶ Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface enp0s3, id 0 | | | | | | |
| ▼ Ethernet II, Src: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e), Dst: PcsCompu_d2:08:f2 (08:00:27:d2:08:f2) | | | | | | |
| ▶ Destination: PcsCompu_d2:08:f2 (08:00:27:d2:08:f2) | | | | | | |
| ▶ Source: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e) | | | | | | |
| Type: IPv4 (0x0800) | | | | | | |
| ▶ Internet Protocol Version 4, Src: 10.0.0.8, Dst: 10.0.0.10 | | | | | | |
| ▶ Transmission Control Protocol, Src Port: 38172, Dst Port: 80, Seq: 0, Len: 0 | | | | | | |
| 0000 | 08 00 27 d2 08 f2 08 00 | 27 2a e7 1e 08 00 45 00 | ..'. '*...E. | | | |
| 0010 | 00 3c 92 bd 40 00 40 06 | 93 ed 0a 00 00 08 0a 00 | .<...@.@. | | | |
| 0020 | 00 0a 95 1c 00 50 0c e8 | 06 fb 00 00 00 00 a0 02 |P. | | | |
| 0030 | fa f0 14 40 00 00 02 04 | 05 b4 04 02 08 0a 51 aa | ...@.....Q. | | | |
| 0040 | 70 f1 00 00 00 00 01 03 | 03 07 | p..... | | | |

9.3.2 Segunda ativação

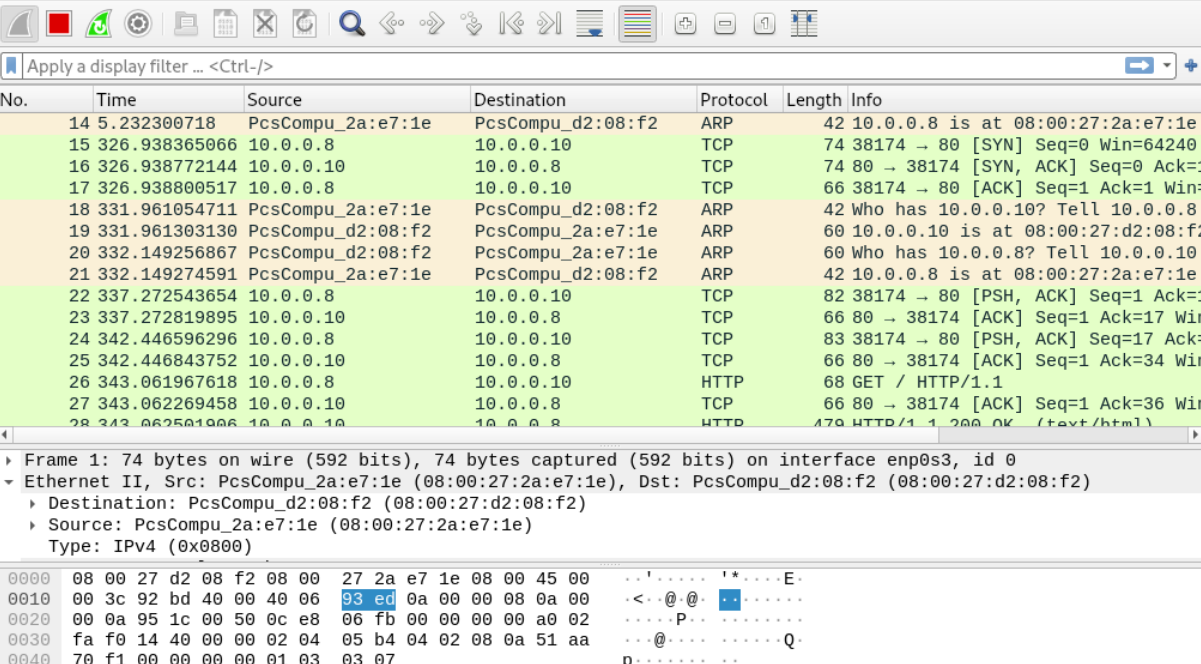
Utilizando o telnet para fazer uma requisição GET:

```
root@wireshark:/home/wire# telnet 10.0.0.10 80
Trying 10.0.0.10...
Connected to 10.0.0.10.
Escape character is '^]'.
GET / HTTP/1.1
HOST: www.data
```

```
HTTP/1.1 200 OK
Server: nginx/1.18.0
Date: Fri, 20 May 2022 21:36:18 GMT
Content-Type: text/html
Content-Length: 176
Last-Modified: Fri, 20 May 2022 17:26:28 GMT
Connection: keep-alive
ETag: "6287cf44-b0"
Accept-Ranges: bytes
```

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title> Matricula</title>
</head>
<body>
```


Detecção da comunicação no Wireshark:



Wireshark interface showing a network traffic capture. The packet list pane displays the following packets:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|---------------|-------------------|-------------------|----------|--------|-------------------------------------|
| 14 | 5.232300718 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | 10.0.0.8 is at 08:00:27:2a:e7:1e |
| 15 | 326.938365066 | 10.0.0.8 | 10.0.0.10 | TCP | 74 | 38174 → 80 [SYN] Seq=0 Win=64240 |
| 16 | 326.938772144 | 10.0.0.10 | 10.0.0.8 | TCP | 74 | 80 → 38174 [SYN, ACK] Seq=0 Ack=: |
| 17 | 326.938800517 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38174 → 80 [ACK] Seq=1 Ack=1 Win=: |
| 18 | 331.961054711 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | Who has 10.0.0.10? Tell 10.0.0.8 |
| 19 | 331.961303130 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | 10.0.0.10 is at 08:00:27:d2:08:f2 |
| 20 | 332.149256867 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | Who has 10.0.0.8? Tell 10.0.0.10 |
| 21 | 332.149274591 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | 10.0.0.8 is at 08:00:27:2a:e7:1e |
| 22 | 337.272543654 | 10.0.0.8 | 10.0.0.10 | TCP | 82 | 38174 → 80 [PSH, ACK] Seq=1 Ack=: |
| 23 | 337.272819895 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38174 [ACK] Seq=1 Ack=17 Win=: |
| 24 | 342.446596296 | 10.0.0.8 | 10.0.0.10 | TCP | 83 | 38174 → 80 [PSH, ACK] Seq=17 Ack=: |
| 25 | 342.446843752 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38174 [ACK] Seq=1 Ack=34 Win=: |
| 26 | 343.061967618 | 10.0.0.8 | 10.0.0.10 | HTTP | 68 | GET / HTTP/1.1 |
| 27 | 343.062269458 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38174 [ACK] Seq=1 Ack=36 Win=: |
| 28 | 343.062501006 | 10.0.0.10 | 10.0.0.8 | HTTP | 479 | HTTP/1.1 200 OK (text/html) |

The packet details pane for the selected packet (No. 28) shows:

- Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface enp0s3, id 0
- Ethernet II, Src: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e), Dst: PcsCompu_d2:08:f2 (08:00:27:d2:08:f2)
- Destination: PcsCompu_d2:08:f2 (08:00:27:d2:08:f2)
- Source: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e)
- Type: IPv4 (0x0800)

The packet bytes pane shows the raw data in hexadecimal and ASCII:

```
0000 08 00 27 d2 08 f2 08 00 27 2a e7 1e 08 00 45 00  ...*...E.
0010 00 3c 92 bd 40 00 40 06 93 ed 0a 00 00 08 0a 00  <...@...
0020 00 0a 95 1c 00 50 0c e8 06 fb 00 00 00 00 a0 02  ....P...
0030 fa f0 14 40 00 00 02 04 05 b4 04 02 08 0a 51 aa  ...@...Q.
0040 70 f1 00 00 00 00 01 03 03 07                    p.....
```

9.3.3 Terceira ativação

Utilizando o telnet para fazer uma requisição HEAD:

```
root@wireshark:/home/wire# telnet 10.0.0.10 80
Trying 10.0.0.10...
Connected to 10.0.0.10.
Escape character is '^]'.
HEAD / HTTP/1.1
HOST: www.data

HTTP/1.1 200 OK
Server: nginx/1.18.0
Date: Fri, 20 May 2022 21:38:50 GMT
Content-Type: text/html
Content-Length: 176
Last-Modified: Fri, 20 May 2022 17:26:28 GMT
Connection: keep-alive
ETag: "6287cf44-b0"
Accept-Ranges: bytes

^CConnection closed by foreign host.
root@wireshark:/home/wire#
```


Detecção da comunicação no Wireshark:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|---------------|-------------------|-------------------|----------|--------|------------------------------------|
| 36 | 423.353237172 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | 10.0.0.10 is at 08:00:27:d2:08:f2 |
| 37 | 478.228112661 | 10.0.0.8 | 10.0.0.10 | TCP | 74 | 38176 → 80 [SYN] Seq=0 Win=64240 |
| 38 | 478.228484342 | 10.0.0.10 | 10.0.0.8 | TCP | 74 | 80 → 38176 [SYN, ACK] Seq=0 Ack= |
| 39 | 478.228508858 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38176 → 80 [ACK] Seq=1 Ack=1 Win= |
| 40 | 483.257448464 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | Who has 10.0.0.10? Tell 10.0.0.8 |
| 41 | 483.257764741 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | 10.0.0.10 is at 08:00:27:d2:08:f2 |
| 42 | 483.446666078 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | Who has 10.0.0.8? Tell 10.0.0.10 |
| 43 | 483.446684643 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | 10.0.0.8 is at 08:00:27:2a:e7:1e |
| 44 | 488.616201059 | 10.0.0.8 | 10.0.0.10 | TCP | 83 | 38176 → 80 [PSH, ACK] Seq=1 Ack= |
| 45 | 488.616460959 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38176 [ACK] Seq=1 Ack=18 Win= |
| 46 | 493.860875518 | 10.0.0.8 | 10.0.0.10 | TCP | 83 | 38176 → 80 [PSH, ACK] Seq=18 Ack= |
| 47 | 493.861242450 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38176 [ACK] Seq=1 Ack=35 Win= |
| 48 | 494.830286352 | 10.0.0.8 | 10.0.0.10 | HTTP | 68 | HEAD / HTTP/1.1 |
| 49 | 494.830647332 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38176 [ACK] Seq=1 Ack=37 Win= |
| 50 | 494.830828755 | 10.0.0.10 | 10.0.0.8 | HTTP | 203 | HTTP/1.1 200 OK |

Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface enp0s3, id 0

Ethernet II, Src: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e), Dst: PcsCompu_d2:08:f2 (08:00:27:d2:08:f2)

- Destination: PcsCompu_d2:08:f2 (08:00:27:d2:08:f2)
- Source: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e)
- Type: IPv4 (0x0800)

| | | | |
|------|-------------------------|-------------------------|-------------------|
| 0000 | 08 00 27 d2 08 f2 08 00 | 27 2a e7 1e 08 00 45 00 | ..'. !*...E. |
| 0010 | 00 3c 92 bd 40 00 40 06 | 93 ed 0a 00 00 08 0a 00 | <...@.@. |
| 0020 | 00 0a 95 1c 00 50 0c e8 | 06 fb 00 00 00 00 a0 02 |P..... |
| 0030 | fa f0 14 40 00 00 02 04 | 05 b4 04 02 08 0a 51 aa | ...@.....Q. |
| 0040 | 70 f1 00 00 00 00 01 03 | 03 07 | p..... |

9.3.4 Quarta ativação

Utilizando o curl para fazer uma requisição HEAD:

```
root@wireshark:/home/wire# curl -I 10.0.0.10:80
HTTP/1.1 200 OK
Server: nginx/1.18.0
Date: Fri, 20 May 2022 21:40:47 GMT
Content-Type: text/html
Content-Length: 176
Last-Modified: Fri, 20 May 2022 17:26:28 GMT
Connection: keep-alive
ETag: "6287cf44-b0"
Accept-Ranges: bytes

root@wireshark:/home/wire#
```

Detecção da comunicação no Wireshark:

| | | | | | | |
|----|---------------|-------------------|-------------------|------|-----|------------------------------------|
| 59 | 612.031799476 | 10.0.0.8 | 10.0.0.10 | TCP | 74 | 38178 → 80 [SYN] Seq=0 Win=64240 |
| 60 | 612.032138456 | 10.0.0.10 | 10.0.0.8 | TCP | 74 | 80 → 38178 [SYN, ACK] Seq=0 Ack=: |
| 61 | 612.032166809 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38178 → 80 [ACK] Seq=1 Ack=1 Win= |
| 62 | 612.032259955 | 10.0.0.8 | 10.0.0.10 | HTTP | 140 | HEAD / HTTP/1.1 |
| 63 | 612.032377015 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38178 [ACK] Seq=1 Ack=75 Win= |
| 64 | 612.032563898 | 10.0.0.10 | 10.0.0.8 | HTTP | 303 | HTTP/1.1 200 OK |
| 65 | 612.032572003 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38178 → 80 [ACK] Seq=75 Ack=238 W |
| 66 | 612.033124775 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38178 → 80 [FIN, ACK] Seq=75 Ack= |
| 67 | 612.033272494 | 10.0.0.10 | 10.0.0.8 | TCP | 66 | 80 → 38178 [FIN, ACK] Seq=238 Ac |
| 68 | 612.033281992 | 10.0.0.8 | 10.0.0.10 | TCP | 66 | 38178 → 80 [ACK] Seq=76 Ack=239 W |
| 69 | 617.080266467 | PcsCompu_d2:08:f2 | PcsCompu_2a:e7:1e | ARP | 60 | Who has 10.0.0.8? Tell 10.0.0.10 |
| 70 | 617.080283600 | PcsCompu_2a:e7:1e | PcsCompu_d2:08:f2 | ARP | 42 | 10.0.0.8 is at 08:00:27:2a:e7:1e |

Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface enp0s3, id 0
Ethernet II, Src: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e), Dst: PcsCompu_d2:08:f2 (08:00:27:d2:08:f2)
Destination: PcsCompu_d2:08:f2 (08:00:27:d2:08:f2)
Source: PcsCompu_2a:e7:1e (08:00:27:2a:e7:1e)
Type: IPv4 (0x0800)

| | | |
|------|---|-------------------|
| 0000 | 08 00 27 d2 08 f2 08 00 27 2a e7 1e 08 00 45 00 | ..'. '*...E. |
| 0010 | 00 3c 92 bd 40 00 40 06 93 ed 0a 00 00 08 0a 00 | <..@. @. |
| 0020 | 00 0a 95 1c 00 50 0c e8 06 fb 00 00 00 00 a0 02 | ...P. |
| 0030 | fa f0 14 40 00 00 02 04 05 b4 04 02 08 0a 51 aa | ...@.....Q. |
| 0040 | 70 f1 00 00 00 00 01 03 03 07 | p..... |

10 Conclusão

A prática da atividade foi extremamente importante para o meu aprofundamento no conhecimento de comunicação na arquitetura Cliente/Servidor. Alguns passos utilizaram os protocolos TCP e HTTP da camada de transporte e aplicação, auxiliando no prosseguimento do curso.

Vídeo

<https://drive.google.com/file/d/1pYg0LZJ6YQmDwVpCMbULBwAVeYiVb1rx/view?usp=sharing>