

CHECK POINT 04 – 2º SEMESTRE

NOME	RM
VITOR TORRES DANTAS	88415
MATHEUS DE OLIVEIRA	88430

Encaminhamento de porta SSH

O encaminhamento de porta SSH, ou tunelamento SSH, é uma técnica de **rede segura em que dados são trocados entre dispositivos** — como uma máquina local e remota — usando uma conexão SSH.

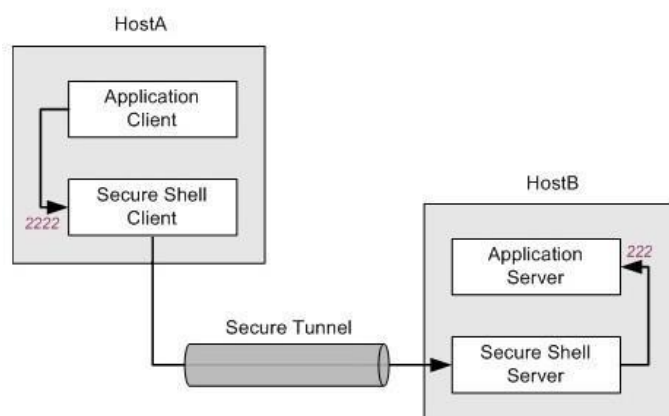
SSH (Secure Shell) é um protocolo de comunicação de rede que usa criptografia para habilitar conexões remotas e criptografadas de e para dispositivos, permitindo que **dois computadores compartilhem dados e se comuniquem com segurança em redes não seguras**. Ao criptografar dados, o encaminhamento de porta SSH aprimora a segurança transmitida pelo túnel e os **protege de possíveis interceptações ou espionagens**. É um recurso poderoso frequentemente usado por **administradores de sistema, desenvolvedores e usuários que precisam de acesso seguro** a recursos em diferentes redes ou para contornar restrições de rede.

Existem três tipos de encaminhamento de porta SSH:

- 1. Encaminhamento de porta local (VOCÊ <-- CLIENTE):** redireciona o tráfego de uma porta local na máquina cliente para uma porta especificada em um servidor remoto por meio de uma conexão SSH.
- 2. Encaminhamento de porta remota (VOCÊ --> CLIENTE):** redireciona o tráfego de uma porta no servidor remoto para uma porta especificada na máquina cliente.
- 3. Encaminhamento de porta dinâmico:** cria um proxy SOCKS na máquina cliente, permitindo o encaminhamento de tráfego de vários aplicativos por meio da conexão SSH.

Exploraremos o **encaminhamento de porta local** neste Check Point. Primeiro, vamos entender como o encaminhamento de porta funciona e como executá-lo.

Encaminhamento de porta local



Quando um usuário precisa acessar um recurso ou serviço localizado em um servidor remoto, mas não consegue fazê-lo diretamente devido a configurações de firewall, configurações de rede ou limitações de rede privada, o encaminhamento de porta local é utilizado.

Isso envolve usar um SSH para estabelecer um túnel seguro entre uma máquina local (o cliente) e um servidor remoto. Ele permite que os usuários acessem serviços ou recursos no servidor de destino que, de outra forma, seriam inacessíveis devido a configurações de firewall ou limitações de rede.

Para estabelecer o encaminhamento de porta local, o usuário inicia uma conexão SSH com o servidor remoto com a `-L` opção seguida pelas especificações de encaminhamento. A sintaxe para configurar o encaminhamento de porta local é:

```
ssh -L 8080:192.168.1.100:80 user@remote\_server
```

Por que o encaminhamento de portas é importante?

O encaminhamento de porta é crucial quando se trata de rede e obtenção de acesso a recursos dentro de redes privadas. A seguir estão algumas das principais razões para a importância do encaminhamento de porta:

Acessando serviços de fora da rede local: O encaminhamento de porta permite que usuários ou dispositivos externos da Internet acessem serviços, aplicativos ou recursos hospedados em dispositivos dentro de uma rede privada.

Execução de aplicativos especializados: o encaminhamento de porta garante que os aplicativos possam enviar e receber dados pelas portas necessárias, permitindo que funcionem corretamente.

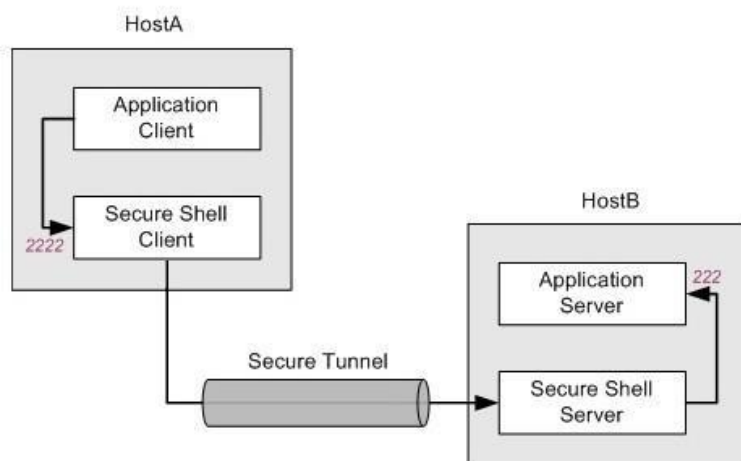
Jogos e conexões ponto a ponto: os jogadores geralmente utilizam o encaminhamento de porta para hospedar servidores de jogos ou participar de jogos multijogador, permitindo conexões de entrada por meio de portas específicas. **Habilitando gerenciamento e controle remotos:** Facilita o gerenciamento remoto de dispositivos como câmeras, dispositivos IoT ou sistemas de armazenamento conectado à rede (NAS) dentro da rede local.

Segurança e controle: permite a abertura seletiva de portas, garantindo que apenas serviços ou aplicativos específicos sejam acessíveis, reduzindo assim o risco de acesso não autorizado.

Flexibilidade na configuração de rede: o encaminhamento de porta oferece flexibilidade na configuração de rede, permitindo que os usuários personalizem como o tráfego de entrada é direcionado e quais dispositivos ou serviços podem ser acessados de fora da rede local.

ATIVIDADE A SER REALIZADA:

Na configuração mostrada abaixo, o cliente do aplicativo e o cliente Secure Shell são executados no HostA. O servidor Secure Shell e o servidor do aplicativo são executados no HostB. Todos os dados enviados para a porta 2222 no HostA são encaminhados para a porta 222 no HostB. Nesse arranjo, todos os dados em trânsito são criptografados com segurança. O comando a seguir (no qual localhost identifica o endereço de loopback no HostB) configura isso:



```
ssh -L listening_port:app_host:hostport user@sshserver
```

1. Realize um laboratório onde contenha dois host: cliente e servidor (1 pt)

Cliente: eth0: 192.168.10.20

Servidor:

enp0s3: 192.168.10.10

Servidor apache na porta 80 e 8080 (ou python3); Servidor ssh 22;

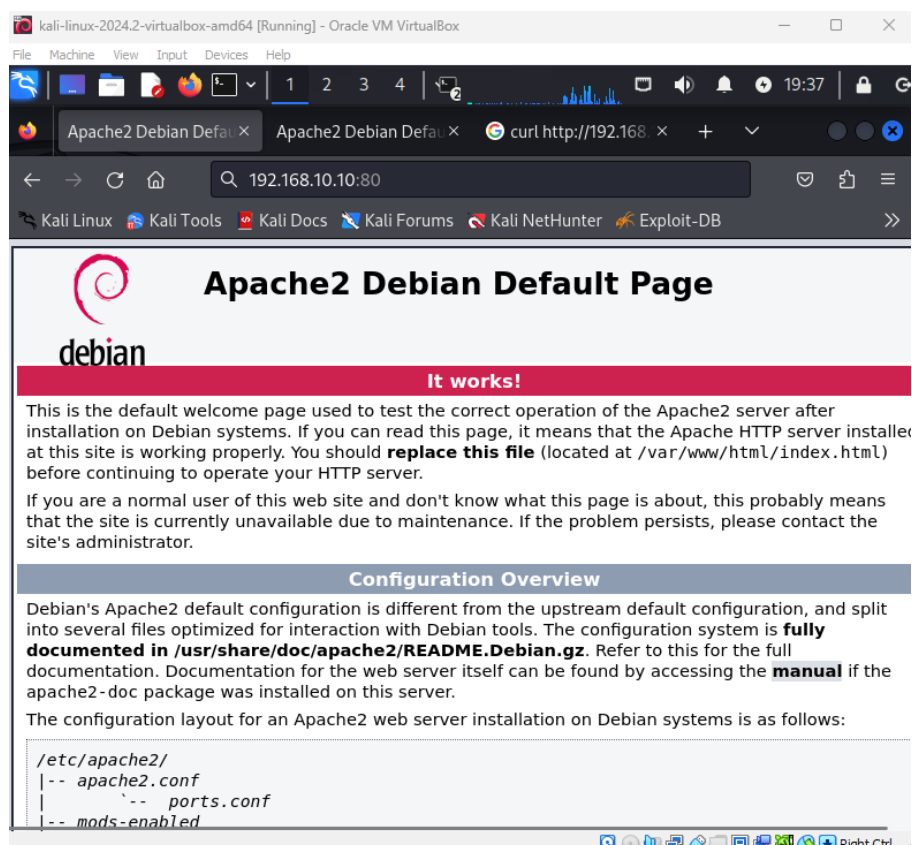
2. Exibir dois sites a escolha do aluno: (1 pt) site A: na porta 80 Site B: na porta 8080

Site na porta 80: apache2

Site na porta 8080: python3

3. Acesso: (2 pts)

Cliente acessa o site na porta 80 sem o ssh;

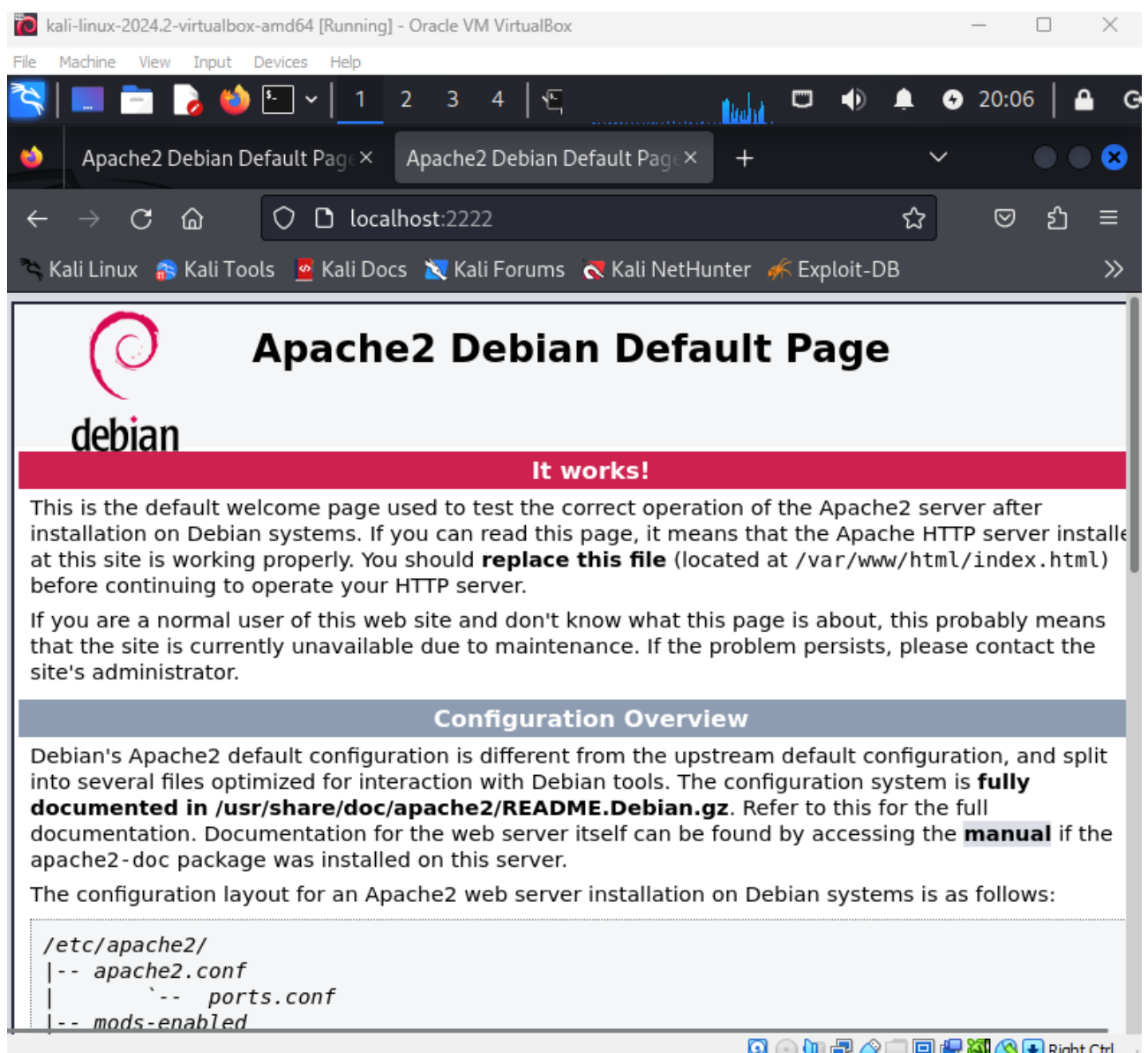


Cliente acessa o site na porta 8080 **após ativar** o encaminhamento de porta local através do **ssh** na porta 2222.

```
File Actions Edit View Help
aluno@debian:~$ ssh -L 2222:localhost:8080 aluno@192.168.10.10
aluno@192.168.10.10's password:
Linux debian 5.10.0-32-amd64 #1 SMP Debian 5.10.223-1 (2024-08-10) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Aug 21 20:51:06 2024 from 192.168.10.10
aluno@debian:~$
```



4. Prints das seguintes telas: (3 pts)

1. Configuração das placas de redes do cliente e servidor;

Servidor

```
vm2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@debian:~# ip -br -c a
lo                UNKNOWN      127.0.0.1/8 ::1/128
enp0s3            UP          10.0.2.15/24 fe80::a00:27ff:feeb:37e0/64
enp0s8            UP          192.168.10.10/24 fe80::a00:27ff:fec1:f89a/64
root@debian:~# _
```

Cliente

```
root@kali: /etc/network
File Actions Edit View Help
(root@kali)-[/etc/network]
# ip -br -c a
lo                UNKNOWN      127.0.0.1/8 ::1/128
eth0              UP          10.0.2.15/24 fe80::d105:a093:3ca5:c0eb/64
eth1              UP          192.168.10.20/24
```

2. Arquivo de configuração do servidor da porta 80,

```
root@debian:/etc/apache2# cat ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

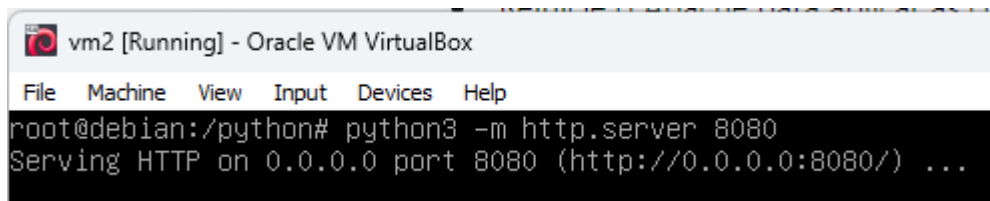
Listen 80
Listen 8080

<IfModule ssl_module>
    Listen 443
</IfModule>

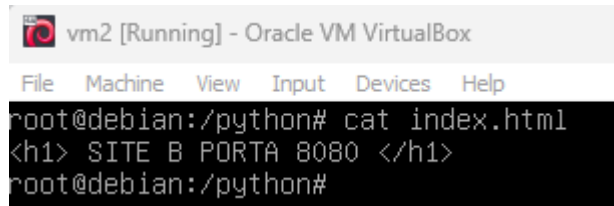
<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
root@debian:/etc/apache2#
```

3. Arquivo de configuração do servidor na porta 8080;

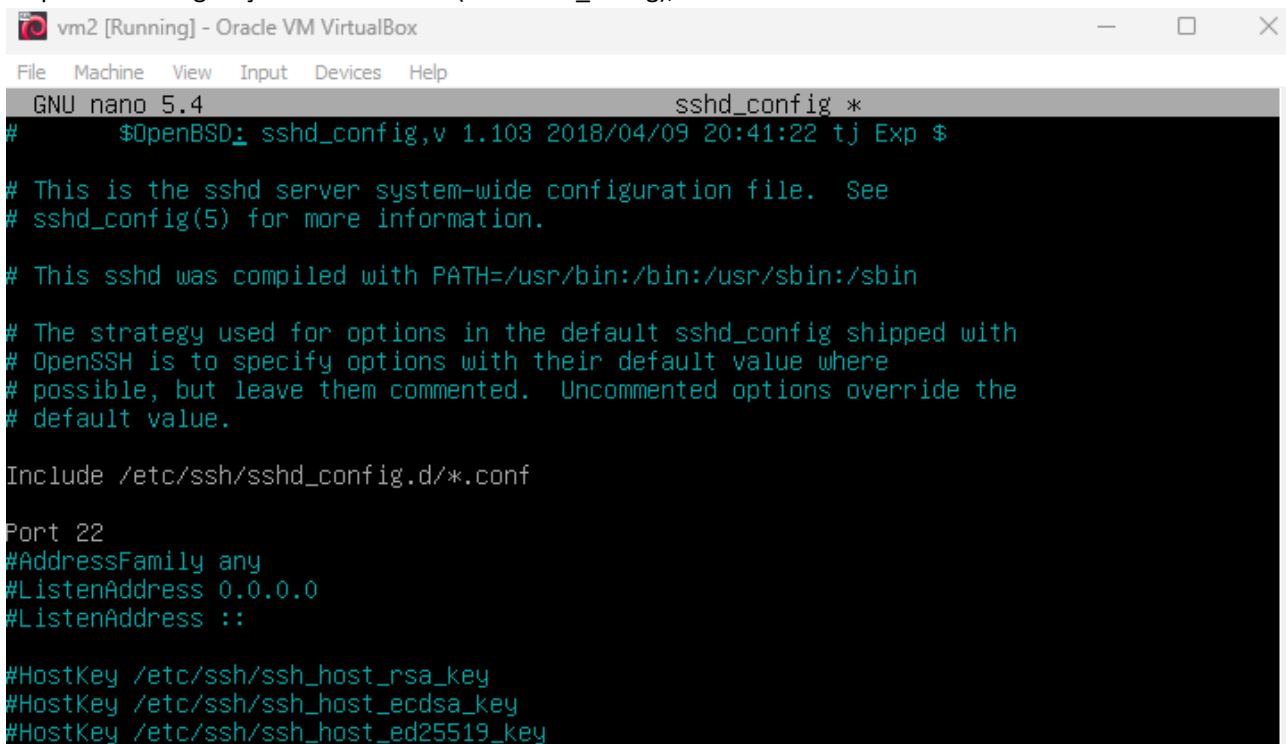


```
vm2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@debian:/python# python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
```



```
vm2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
root@debian:/python# cat index.html
<h1> SITE B PORTA 8080 </h1>
root@debian:/python#
```

4. Arquivo de configuração e conexão ssh (dica: sshd_config);



vm2 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

GNU nano 5.4 sshd_config *

```
# $OpenBSD: sshd_config,v 1.103 2018/04/09 20:41:22 tj Exp $

# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

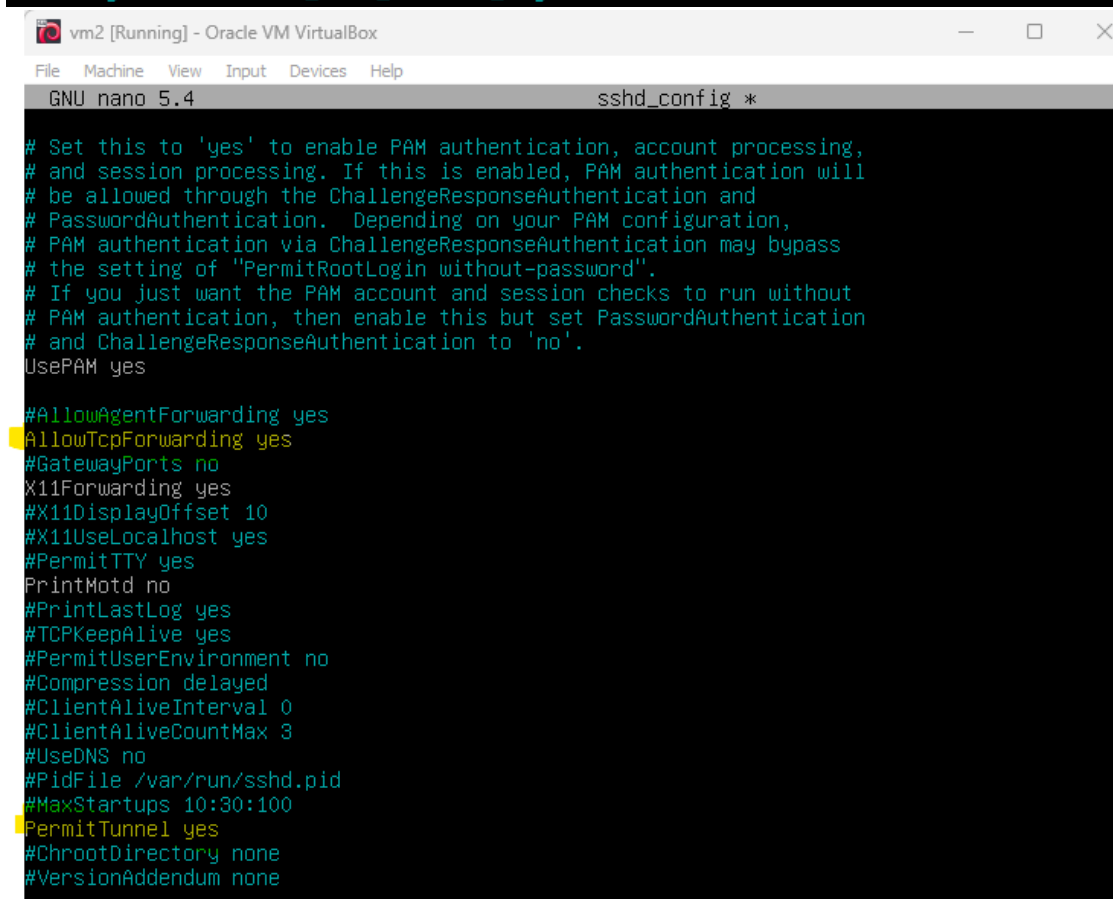
# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key
```



vm2 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

GNU nano 5.4 sshd_config *

```
# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing.  If this is enabled, PAM authentication will
# be allowed through the ChallengeResponseAuthentication and
# PasswordAuthentication.  Depending on your PAM configuration,
# PAM authentication via ChallengeResponseAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and ChallengeResponseAuthentication to 'no'.
UsePAM yes

#AllowAgentForwarding yes
AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes
PrintMotd no
#PrintLastLog yes
#TCPKeepAlive yes
#PermitUserEnvironment no
#Compression delayed
#ClientAliveInterval 0
#ClientAliveCountMax 3
#UseDNS no
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
PermitTunnel yes
#ChrootDirectory none
#VersionAddendum none
```


5. Comunicação entre as placas de redes;

```
root@kali: ~  
File Actions Edit View Help  
  
(root@kali)~  
# ping 192.168.10.10  
PING 192.168.10.10 (192.168.10.10) 56(84) bytes of data.  
64 bytes from 192.168.10.10: icmp_seq=1 ttl=64 time=1.38 ms  
64 bytes from 192.168.10.10: icmp_seq=2 ttl=64 time=0.803 ms  
64 bytes from 192.168.10.10: icmp_seq=3 ttl=64 time=0.627 ms  
64 bytes from 192.168.10.10: icmp_seq=4 ttl=64 time=0.728 ms  
64 bytes from 192.168.10.10: icmp_seq=5 ttl=64 time=0.625 ms  
^C  
— 192.168.10.10 ping statistics —  
5 packets transmitted, 5 received, 0% packet loss, time 4056ms  
rtt min/avg/max/mdev = 0.625/0.833/1.382/0.282 ms
```

Ping do cliente para o servidor

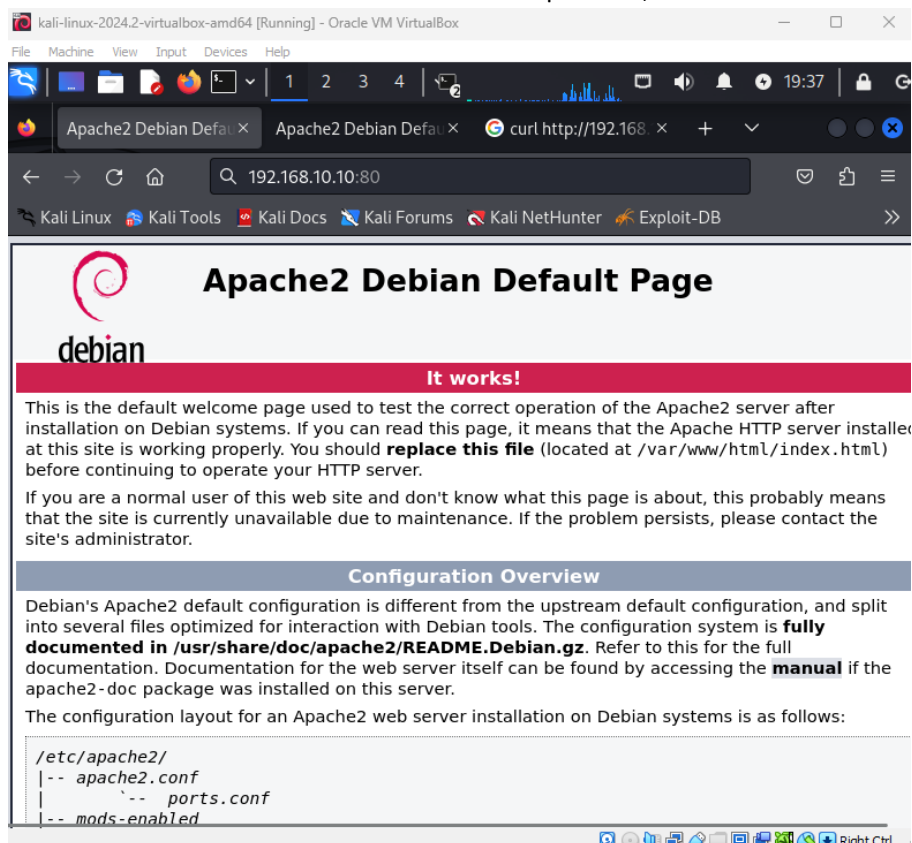
```
vm2 [Running] - Oracle VM VirtualBox  
File Machine View Input Devices Help  
  
root@debian:/home# ping 192.168.10.20  
PING 192.168.10.20 (192.168.10.20) 56(84) bytes of data.  
64 bytes from 192.168.10.20: icmp_seq=1 ttl=64 time=0.886 ms  
64 bytes from 192.168.10.20: icmp_seq=2 ttl=64 time=0.651 ms  
64 bytes from 192.168.10.20: icmp_seq=3 ttl=64 time=0.656 ms  
64 bytes from 192.168.10.20: icmp_seq=4 ttl=64 time=0.637 ms  
^C  
--- 192.168.10.20 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3046ms  
rtt min/avg/max/mdev = 0.637/0.707/0.886/0.103 ms
```

Ping do servidor para o cliente

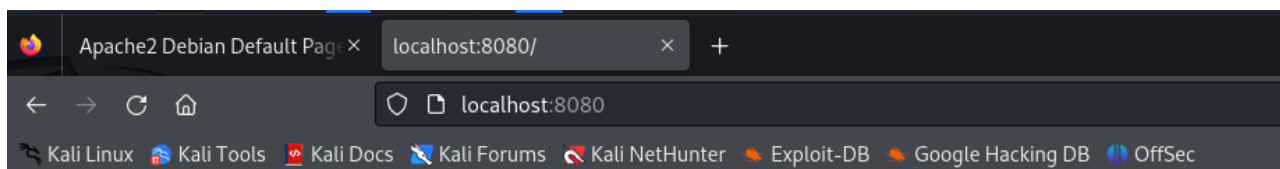
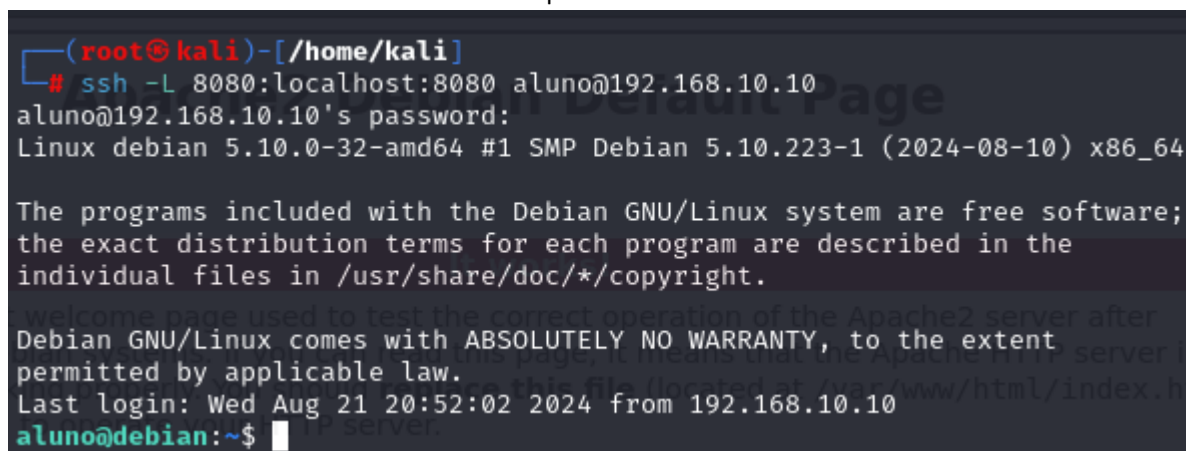
6. Portas abertas do servidor;

```
vm2 [Running] - Oracle VM VirtualBox  
File Machine View Input Devices Help  
  
root@debian:/python# ss -nltp  
State      Recv-Q      Send-Q      Local Address:Port      Peer Address:Port  
Process  
LISTEN     0            128         0.0.0.0:22               0.0.0.0:*  
users:(("sshd",pid=460,fd=3))  
LISTEN     0            511         *:80                    *:80  
users:(("apache2",pid=690,fd=4),("apache2",pid=689,fd=4),("apache2",pid=688,fd=4))  
LISTEN     0            128         [::]:22                 [::]:*  
users:(("sshd",pid=460,fd=4))  
root@debian:/python# _
```

7. Site do servidor acessado através do cliente na porta 80;



8. Site do servidor acessado através do cliente na porta 8080 acessado no localhost.



SITE B PORTA 8080

5. Pitch de 15 segundos (NO MÁXIMO) gravando as telas do CLIENTE E SERVIDOR funcionando o acesso local do security shell, contendo: (2 pts)

1. Sem ativar a porta local do ssh:

- a. Tela do cliente conectando o site A na porta 80;
- b. Tela do cliente conectando o site B na porta 8080;

2. Ativando a porta local do ssh:

- a. Ativação do encaminhamento de porta 8080;
- b. Tela do cliente conectando o site A na porta 80;
- c. Tela do cliente conectando o site B na porta 8080;

6. Entrega do arquivo compactado (.rar ou .zip) com nome: turma_rm_aluno_sobrenome.pdf contendo os seguintes arquivos: (1 pt)

- a) um arquivo com nome: turma_rm_aluno_nome_sobrenome.pdf
- b) um pitch com nome: pitch_turma_rm_aluno_sobrenome.pdf de 10 segundos gravando a tela do cliente e do servidor;

EXEMPLO: 4SIA_1388_JOAO_SILVA.rar (e dentro do .rar: 4SIA_1388_JOAO_SILVA.pdf e 4SIA_1388_JOAO_SILVA.mov (OU OUTRA EXTENSÃO))

observação: o Check Point será em dupla