Módulo 3 - Aulas 1 e 2

Tarefas

No final deste módulo você deve submeter em um ÚNICO arquivo PDF os seguintes prints:

- Atividade 3.1: passo 7.
- Atividade 3.2: passo 6.
- Atividade 3.3: passo 21.
- Atividade 3.4: passo 6.
- Atividade 3.5: passo 9.

Regras para a elaboração do documento:

1. Antes de cada Print, adicione obrigatoriamente uma frase explicativa que sinalize do que se trata o print. A inserção de prints sem a devida frase explicativa será considerada como tentativa de atrapalhar a correção do instrutor e será penalizada a critério do instrutor. Exemplo:

Print da atividade 3.1: [Imagem com o Print]

2. Os Prints devem ser tirados da TELA CHEIA. Quando tirados da VM da AWS, devem ser capturados **obrigatoriamente** da tela cheia clicando no botão “Screenshot” → “Take screenshot” da barra de ferramentas do Hypervisor da AWS.
3. Insira **somente a quantidade de Prints solicitados por atividade** usando exclusivamente 1 página por print. **A página do documento onde você vai inserir o Print deve estar com a orientação no modo PAISAGEM** ´para termos melhor aproveitamento do espaço. Ou seja, seu documento deverá ter a mesma quantidade de páginas que a quantidade do total de Prints! A inserção de prints desnecessários será considerada como tentativa de atrapalhar a correção do instrutor e será penalizada com nota 0.

4. Apresente Prints legíveis e com tamanho correto para fácil leitura. O envio de prints com letras minúsculas poderá ser considerado como tentativa de atrapalhar a correção do instrutor e será penalizada a critério do instrutor.

Atividade 3.1 – Explorando Exploits conhecidos no Kali Linux

Nesta atividade, vamos explorar a ferramenta Exploit Database para listar os Exploits conhecidos no Kali Linux. O Exploit Database contém base de dados de Exploits que podem ser executados por meio do Kali Linux. Nesta atividade, vamos explorar como fazer a busca de Exploits com fins acadêmicos. Todo material apresentado aqui deve ser usado somente para fins acadêmicos.

Vamos começar inicializando nosso Kali Linux via RDP ao IP: 192.168.98.40, com usuário “aluno” e senha “rnpesr”.

1. Abra o Terminal e execute o seguinte comando (com senha “rnpesr”) para ser super usuário:

```
┌──(aluno㉿kali)-[~]
└─$ sudo -i
[sudo] senha para aluno:
```

2. Veja onde se encontram os arquivos do exploitdb:

```
┌──(root㉿kali)-[~]
└─# ls -al /usr/share/exploitdb
total 10140
drwxr-xr-x   4 root root     4096 set  5 18:34 .
drwxr-xr-x 302 root root    12288 set  5 18:36 ..
drwxr-xr-x  65 root root     4096 jul  4  2024 exploits
-rw-r--r--   1 root root 10137459 ago 26 21:17 files_exploits.csv
-rw-r--r--   1 root root   220152 ago 26 21:17 files_shellcodes.csv
drwxr-xr-x  42 root root     4096 jul  4  2024 shellcodes
```

3. Veja os comandos básicos do Exploitdb:

```
┌──(root㉿kali)-[~]
└─# searchsploit
  Usage: searchsploit [options] term1 [term2] ... [termN]

==========
 Examples
==========
  searchsploit afd windows local
  searchsploit -t oracle windows
  searchsploit -p 39446
  searchsploit linux kernel 3.2 --exclude="(PoC)|/dos/"
  searchsploit -s Apache Struts 2.0.0
  searchsploit linux reverse password
  searchsploit -j 55555 | jq
  searchsploit --cve 2021-44228

  For more examples, see the manual: https://www.exploit-db.com/
searchsploit

=========
 Options
=========
## Search Terms
   -c, --case     [term]      Perform a case-sensitive search (Default
is inSEnsITiVe)
   -e, --exact    [term]      Perform an EXACT & order match on exploit
title (Default is an AND match on each term) [Implies "-t"]
                                 e.g. "WordPress 4.1" would not be
detect "WordPress Core 4.1")
   -s, --strict               Perform a strict search, so input values
must exist, disabling fuzzy search for version range
                                 e.g. "1.1" would not be detected in
"1.0 < 1.3")
   -t, --title    [term]      Search JUST the exploit title (Default is
title AND the file's path)
       --exclude="term"       Remove values from results. By using "|"
to separate, you can chain multiple values
                                 e.g. --exclude="term1|term2|term3"
       --cve      [CVE]       Search for Common Vulnerabilities and
Exposures (CVE) value

## Output
   -j, --json     [term]      Show result in JSON format
   -o, --overflow [term]      Exploit titles are allowed to overflow
their columns
   -p, --path     [EDB-ID]    Show the full path to an exploit (and
also copies the path to the clipboard if possible)
   -v, --verbose              Display more information in output
```

```
   -v, --verbose               Display more information in output
   -w, --www      [term]       Show URLs to Exploit-DB.com rather than
the local path
       --id                    Display the EDB-ID value rather than
local path
       --disable-colour        Disable colour highlighting in search
results

## Non-Searching
   -m, --mirror   [EDB-ID]     Mirror (aka copies) an exploit to the
current working directory
   -x, --examine  [EDB-ID]     Examine (aka opens) the exploit using
$PAGER

## Non-Searching
   -h, --help                  Show this help screen
   -u, --update                Check for and install any exploitdb
package updates (brew, deb & git)

## Automation
       --nmap     [file.xml]   Checks all results in Nmap's XML output
with service version
                                  e.g.: nmap [host] -sV -oX file.xml

=======
 Notes
=======
 * You can use any number of search terms
 * By default, search terms are not case-sensitive, ordering is
irrelevant, and will search between version ranges
   * Use '-c' if you wish to reduce results by case-sensitive searching
   * And/Or '-e' if you wish to filter results by using an exact match
   * And/Or '-s' if you wish to look for an exact version match
 * Use '-t' to exclude the file's path to filter the search results
   * Remove false positives (especially when searching using numbers -
i.e. versions)
 * When using '--nmap', adding '-v' (verbose), it will search for even
more combinations
 * When updating or displaying help, search terms will be ignored
```

A saída mostrada acima conta com os seguintes parâmetros:

- Exemplos: A seção fornece exemplos de como usar o comando searchsploit para pesquisar exploits com diferentes critérios, como termos de pesquisa, tipos de sistemas operacionais, identificadores CVE, etc.

- Opções de Busca: Define as opções de busca disponíveis, como a opção -c para pesquisa sensível a maiúsculas e minúsculas, -e para correspondência exata no título do exploit, -s para pesquisa estrita, etc.
- Opções de Saída: Descreve as opções relacionadas à exibição dos resultados, como a opção -j para mostrar resultados em formato JSON, -o para permitir que os títulos dos exploits ultrapassem as colunas, -p para exibir o caminho completo de um exploit, entre outras.
- Opções de Não-Busca: Inclui opções que não estão diretamente relacionadas à busca de exploits, como -m para espelhar (copiar) um exploit para o diretório de trabalho atual, -x para examinar (abrir) um exploit usando o paginador padrão, etc.
- Opções de Automação: Apresenta opções para automação, como --nmap para verificar todos os resultados em saída XML do Nmap com a versão do serviço.
- Notas: Oferece informações adicionais, como a capacidade de usar qualquer número de termos de pesquisa, a sensibilidade a maiúsculas e minúsculas padrão, a capacidade de pesquisar entre intervalos de versão e dicas sobre como refinar a busca.

4. Atualize a base de dados (demora ao redor de 5 minutos):

```
┌──(root㉿kali)-[~]
└─# searchsploit -u
[i] Updating via apt package management (Expect weekly-ish updates):
exploitdb

Hit:1 http://kali.download/kali kali-rolling InRelease
All packages are up to date.
exploitdb is already the newest version (20250827-0kali1).
The following packages were automatically installed and are no longer
required:
  fonts-liberation2            libkf5notifications5         librte-
bus-pci24

...

Installing:
  exploitdb-papers

Summary:
  Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 0
  Download size: 2561 MB
  Space needed: 2952 MB / 4760 MB available

Get:1 http://kali.download/kali kali-rolling/main amd64 exploitdb-
papers all 20221122-0kali1 [2561 MB]
Fetched 2561 MB in 21s (123 MB/s)
Selecting previously unselected package exploitdb-papers.
(Reading database ... 363049 files and directories currently
installed.)
Preparing to unpack .../exploitdb-papers_20221122-0kali1_all.deb ...
Unpacking exploitdb-papers (20221122-0kali1) ...
Setting up exploitdb-papers (20221122-0kali1) ...
Processing triggers for kali-menu (2025.3.2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this
host.

[+] apt update finished
```

```
[*] apt update finished
```

5. Vamos realizar a primeira busca de Exploits, a busca será relacionada ao Openssl:

```
┌──(root㉿kali)-[~]
└─# searchsploit openssl
------------------------------------------
---------------------------------
 Exploit Title                           |  Path
------------------------------------------
---------------------------------
Apache 2.4.7 + PHP 7.0.2 - 'openssl_seal() | php/remote/40142.php
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuck | unix/remote/21671.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuck | unix/remote/47080.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuck | unix/remote/764.c
Apache mod_ssl OpenSSL < 0.9.6d / < 0.9.7- | unix/remote/40347.txt
OpenSSL - 'ssl3_get_key_exchange()' Use-Af | linux/dos/34427.txt
OpenSSL - Alternative Chains Certificate F | multiple/webapps/38640.rb
OpenSSL - ASN.1 Parsing                    | multiple/remote/23199.c
OpenSSL - ASN1 BIO Memory Corruption       | multiple/dos/18756.txt
OpenSSL - Padding Oracle in AES-NI CBC MAC | multiple/dos/39768.txt
OpenSSL - Remote Denial of Service         | linux/dos/12334.c
OpenSSL 0.9.8c-1 < 0.9.8g-9 (Debian and De | linux/remote/5622.txt
OpenSSL 0.9.8c-1 < 0.9.8g-9 (Debian and De | linux/remote/5632.rb
OpenSSL 0.9.8c-1 < 0.9.8g-9 (Debian and De | linux/remote/5720.py
OpenSSL 0.9.8k/1.0.0-beta2 - DTLS Remote M | multiple/dos/8720.c
OpenSSL 0.9.x - CBC Error Information Leak | linux/remote/22264.txt
OpenSSL 1.0.1f TLS Heartbeat Extension - ' | multiple/remote/32764.py
OpenSSL 1.1.0 - Remote Client Denial of Se | multiple/dos/41192.c
OpenSSL 1.1.0a/1.1.0b - Denial of Service  | linux/dos/40899.py
OpenSSL < 0.9.7l/0.9.8d - SSLv2 Client Cra | multiple/dos/4773.pl
OpenSSL < 0.9.8i - DTLS ChangeCipherSpec R | multiple/dos/8873.c
OpenSSL ASN.1 < 0.9.6j/0.9.7b - Brute Forc | multiple/dos/146.c
OpenSSL SSLv2 - Null Pointer Dereference C | multiple/dos/28726.pl
OpenSSL TLS Heartbeat Extension - 'Heartbl | multiple/remote/32745.py
OpenSSL TLS Heartbeat Extension - 'Heartbl | multiple/remote/32791.c
OpenSSL TLS Heartbeat Extension - 'Heartbl | multiple/remote/32998.c
PHP - 'openssl_x509_parse()' Memory Corrup | php/dos/30395.txt
PHP 6.0 - 'openssl_verify()' Local Buffer  | windows/dos/19963.txt
PHP < 5.3.6 'OpenSSL' Extension - 'openssl | php/dos/35486.php
PHP < 5.3.6 'OpenSSL' Extension - 'openssl | php/dos/35487.php
------------------------------------------
---------------------------------
------------------------------------------
---------------------------------
 Shellcode Title                         |  Path
------------------------------------------
---------------------------------
Linux/x86 - OpenSSL Encrypt (aes256cbc) Fi | linux_x86/46791.c
------------------------------------------
---------------------------------
```

```
-----------------------------------------
---------------------------------
 Paper Title                              |  Path
-----------------------------------------
---------------------------------
OpenSSL - Weak KDF                        | english/41019-openssl---
weak-kdf
OPENSSLDIR Privilege Escalation CVE-2021-2 | docs/english/50747-
openssldir-pr
-----------------------------------------
---------------------------------
```

A saída do comando mostra resultados relacionados a exploits, shellcodes e papers que envolvem a biblioteca OpenSSL. Veja algumas explicações sobre as partes da saída:

- Exploit Titles: A tabela lista vários exploits relacionados ao OpenSSL, incluindo detalhes como o título do exploit e o caminho para o arquivo de exploit. Por exemplo, "Apache 2.4.7 + PHP 7.0.2 - 'openssl_seal()" é um exploit PHP remoto que envolve a função openssl_seal(). Cada linha representa um exploit específico, identificado pelo título, e o caminho indica onde o arquivo do exploit está localizado no sistema.
- Shellcode Title: Esta seção mostra um shellcode relacionado ao OpenSSL, especificamente um shellcode para Linux/x86 que envolve a criptografia OpenSSL usando o algoritmo aes256cbc.
- Paper Title: A última seção lista documentos (papers) relacionados ao OpenSSL, como "OpenSSL - Weak KDF" e "OPENSSLDIR Privilege Escalation CVE-2021-2". Esses documentos podem conter informações detalhadas sobre fraquezas, métodos de ataque ou problemas de segurança específicos no OpenSSL.

6. Realize a busca de Exploits relacionada ao Openssh:

```
┌──(root㉿kali)-[/usr/share/exploitdb]
└─# searchsploit openssh
------------------------------------------------------
---------------------------------
 Exploit Title                                      |  Path
------------------------------------------------------
---------------------------------
Debian OpenSSH - (Authenticated) Remote SELinux Privil | linux/
remote/6094.txt
Dropbear / OpenSSH Server - 'MAX_UNAUTH_CLIENTS' Denia | multiple/
dos/1572.pl
FreeBSD OpenSSH 3.5p1 - Remote Command Execution       | freebsd/
remote/17462.txt
glibc-2.2 / openssh-2.3.0p1 / glibc 2.1.9x - File Read | linux/
local/258.sh
Novell Netware 6.5 - OpenSSH Remote Stack Overflow     | novell/
dos/14866.txt
OpenSSH 1.2 - '.scp' File Create/Overwrite             | linux/
remote/20253.sh
OpenSSH 2.3 < 7.7 - Username Enumeration               | linux/
remote/45233.py
OpenSSH 2.3 < 7.7 - Username Enumeration (PoC)         | linux/
remote/45210.py
OpenSSH 2.x/3.0.1/3.0.2 - Channel Code Off-by-One      | unix/
remote/21314.txt
OpenSSH 2.x/3.x - Kerberos 4 TGT/AFS Token Buffer Over | linux/
remote/21402.txt
OpenSSH 3.x - Challenge-Response Buffer Overflow (1)   | unix/
remote/21578.txt
OpenSSH 3.x - Challenge-Response Buffer Overflow (2)   | unix/
remote/21579.txt
OpenSSH 4.3 p1 - Duplicated Block Remote Denial of Ser | multiple/
dos/2444.sh
OpenSSH 6.8 < 6.9 - 'PTY' Local Privilege Escalation   | linux/
local/41173.c
OpenSSH 7.2 - Denial of Service                        | linux/
dos/40888.py
OpenSSH 7.2p1 - (Authenticated) xauth Command Injectio | multiple/
remote/39569.py
OpenSSH 7.2p2 - Username Enumeration                   | linux/
remote/40136.py
OpenSSH < 6.6 SFTP (x64) - Command Execution           | linux_x86-64/
remote/45000.c
OpenSSH < 6.6 SFTP - Command Execution                 | linux/
remote/45001.py
OpenSSH < 7.4 - 'UsePrivilegeSeparation Disabled' Forw | linux/
local/40962.txt
OpenSSH < 7.4 - agent Protocol Arbitrary Library Loadi | linux/
```

```
OpenSSH < 7.4 - agent Protocol Arbitrary Library Loadi | linux/
remote/40963.txt
OpenSSH < 7.7 - User Enumeration (2)                   | linux/
remote/45939.py
OpenSSH SCP Client - Write Arbitrary Files             | multiple/
remote/46516.py
OpenSSH/PAM 3.6.1p1 - 'gossh.sh' Remote Users Ident    | linux/
remote/26.sh
OpenSSH/PAM 3.6.1p1 - Remote Users Discovery Tool      | linux/
remote/25.c
OpenSSHd 7.2p2 - Username Enumeration                  | linux/
remote/40113.txt
Portable OpenSSH 3.6.1p-PAM/4.1-SuSE - Timing Attack   | multiple/
remote/3303.sh
---------------------------------------------------------
---------------------------------
Shellcodes: No Results
---------------------------------------------------------
---------------------------------
 Paper Title                                           |  Path
---------------------------------------------------------
---------------------------------
Roaming Through the OpenSSH Client: CVE-2016-0777 and  | english/39247-
roaming-through-th
---------------------------------------------------------
---------------------------------
```

A saída do comando mostrado acima apresenta:

- Exploit Titles: A tabela lista vários exploits relacionados ao OpenSSH, incluindo detalhes como o título do exploit e o caminho para o arquivo de exploit. Por exemplo, "OpenSSH 2.3 < 7.7 - Username Enumeration" é um exploit remoto que se concentra na enumeração de nomes de usuário no OpenSSH nas versões de 2.3 a 7.7. Cada linha representa um exploit específico, identificado pelo título, e o caminho indica onde o arquivo do exploit está localizado no sistema.
- Shellcodes: Esta seção indica que não há resultados de shellcodes relacionados ao OpenSSH na saída atual. Shellcodes são pequenos códigos que geralmente são injetados e executados para realizar determinadas ações.
- Paper Title: A última seção lista documentos (papers) relacionados ao OpenSSH, como "Roaming Through the OpenSSH Client: CVE-2016-0777 and CVE-2016-0778." Esses documentos podem conter informações detalhadas sobre vulnerabilidades específicas, métodos de exploração ou outras descobertas relacionadas ao OpenSSH.

7. Finalmente, vamos efetuar a busca de Exploits relacionados ao Windows 10 (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO! Somente as 20 primeiras linhas**):

```
┌──(root㉿kali)-[/usr/share/exploitdb]
└─# searchsploit -t windows 10
------------------------------------------------------------------------
- ---------------------------------
 Exploit Title
|  Path
------------------------------------------------------------------------
- ---------------------------------
Adobe Flash Player < 10.1.53.64 - Action Script Type Confusion (ASLR +
| windows/remote/17187.txt
Apple iTunes 8.1.1.10 (Windows) - 'itms/itcp' Remote Buffer Overflow
| windows/remote/8934.py
Autodesk Backburner Manager 3 < 2016.0.0.2150 - Null Dereference Denial
| windows/dos/41160.py
Avast Anti-Virus < 19.1.2360 - Local Credentials Disclosure
| windows/local/46345.py
Avaya IP Office (IPO) < 10.1 - 'SoftConsole' Remote Buffer Overflow (SE
| windows/remote/43121.txt
Avaya IP Office (IPO) < 10.1 - ActiveX Buffer Overflow
| windows/dos/43120.txt
Cisco WebEx Meetings < 33.6.6 / < 33.9.1 - Privilege Escalation
| windows/local/46479.txt
DameWare Remote Controller < 12.0.0.520 - Remote Code Execution
| windows/remote/43059.py
EA Origin < 10.5.38 - Remote Code Execution
| windows/remote/47019.txt

...
```

O argumento -t é usado para especificar um termo de pesquisa. Neste caso, o termo de pesquisa é "windows 10", indicando que o usuário está interessado em encontrar exploits relacionados ao sistema operacional Windows 10.

8. No canto superior esquerdo, clique em “Aplicativos” → “Navegador web” para abrir o Mozilla Firefox. Explore também o seguinte site para mais informações:

```
https://www.exploit-db.com/
```

9. Feche o Firefox e o Terminal.

Parabéns! Agora você sabe como usar o Exploitdb para explorar Exploits atualizados!

Atividade 3.2 – Explorando varreduras com o Kali Linux

Nesta atividade, vamos explorar a ferramenta Nmap para fazer varreduras de dispositivos em uma rede com o Kali Linux. O Nmap é uma poderosa ferramenta de código aberto usada para mapear redes e descobrir hosts, serviços e informações detalhadas sobre sistemas em uma rede. Ele permite que os administradores de segurança e pentesters avaliem a segurança de uma rede, identificando portas abertas, serviços em execução e até mesmo características específicas do sistema operacional. Todo material apresentado aqui deve ser usado somente para fins acadêmicos.

Vamos começar inicializando nosso Kali Linux via RDP ao IP: 192.168.98.40, com usuário “aluno” e senha “rnpesr”.

1. Abra o Terminal e execute o seguinte comando (com senha “rnpesr”) para ser super usuário:

```
┌──(aluno㉿kali)-[~]
└─$ sudo -i
[sudo] senha para aluno:
```

2. Veja as interfaces da sua máquina virtual:

```
┌──(root㉿kali)-[~]
└─# ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:df:26:13:80  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 192.168.98.40  netmask 255.255.255.0  broadcast
192.168.98.255
        inet6 fe80::1005:27ff:fe44:1631  prefixlen 64  scopeid
0x20<link>
        ether 12:05:27:44:16:31  txqueuelen 1000  (Ethernet)
        RX packets 1818194  bytes 2694820836 (2.5 GiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 61082  bytes 77348531 (73.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Loopback Local)
        RX packets 23  bytes 1937 (1.8 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 23  bytes 1937 (1.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

3. Explore o nmap para listar os IPs ativos das redes na interface de rede eth0:

```
┌──(root㉿kali)-[~]
└─# nmap -sn 192.168.98.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-06 10:00 -03
Nmap scan report for ip-192-168-98-1.ec2.internal (192.168.98.1)
Host is up (0.00020s latency).
MAC Address: 12:BF:8B:B8:B0:87 (Unknown)
Nmap scan report for ip-192-168-98-2.ec2.internal (192.168.98.2)
Host is up (0.00019s latency).
MAC Address: 12:BF:8B:B8:B0:87 (Unknown)
Nmap scan report for ip-192-168-98-20.ec2.internal (192.168.98.20)
Host is up (0.00024s latency).
MAC Address: 12:89:91:6B:9F:03 (Unknown)
Nmap scan report for ip-192-168-98-30.ec2.internal (192.168.98.30)
Host is up (0.00034s latency).
MAC Address: 12:33:11:FB:DE:ED (Unknown)
Nmap scan report for ip-192-168-98-201.ec2.internal (192.168.98.201)
Host is up (0.00018s latency).
MAC Address: 12:CA:37:E3:5B:9B (Unknown)
Nmap scan report for ip-192-168-98-40.ec2.internal (192.168.98.40)
Host is up.
Nmap done: 256 IP addresses (6 hosts up) scanned in 1.95 seconds
```

Na varredura nmap mostrada acima, a rede alvo foi o da interface 192.168.98.0/24. No seu laboratório é possível que sejam diferentes, confira as redes que você tem presente conforme o passo 2.

O parâmetro “sn” especifica a técnica de "ping scan", que é utilizada para varrer uma rede e descobrir quais hosts estão online sem realizar varreduras de portas ou descoberta de serviços. Em outras palavras, o parâmetro -sn indica ao nmap para realizar uma varredura de hosts online sem realizar uma varredura de portas.

4. Agora, vamos coletar os IPs da rede 192.168.98.0/24 em um arquivo de texto na pasta Documentos:

```
┌──(root㉿kali)-[~]
└─# cd /home/aluno/Documentos/

┌──(root㉿kali)-[/home/aluno/Documentos]
└─# ls

┌──(root㉿kali)-[/home/aluno/Documentos]
└─# nmap -sn 192.168.98.0/24 | grep 192 | cut -d ' ' -f 5 > ips.txt

┌──(root㉿kali)-[/home/aluno/Documentos]
└─# ls
ips.txt

┌──(root㉿kali)-[/home/aluno/Documentos]
└─# cat ips.txt
ip-192-168-98-1.ec2.internal
ip-192-168-98-2.ec2.internal
ip-192-168-98-20.ec2.internal
ip-192-168-98-30.ec2.internal
ip-192-168-98-201.ec2.internal
ip-192-168-98-40.ec2.internal
```

O comando mostrado acima conta com os seguintes parâmetros:

- **nmap -sn 192.168.98.0/24**: Realiza um escaneamento de hosts na faixa de endereços IP de 192.168.98.1 a 192.168.98.254 sem realizar o escaneamento de portas (opção -sn para ping scan).
- **|**: O operador de pipe, que direciona a saída do comando anterior para o próximo comando.
- **grep 192**: Filtra as linhas que contêm o número 192 na saída do comando anterior.
- **cut -d ' ' -f 5**: Divide cada linha usando o espaço como delimitador (-d ' ') e extrai o quinto campo (-f 5), que geralmente é o endereço IP do host.
- **> ips.txt**: Redireciona a saída final para um arquivo chamado "ips.txt", cujo conteúdo foi visualizado e apresenta 9 endereços IPs (neste exemplo).

5. Agora, efetuemos a varredura para detectar algum IP que conta com algum tipo de proteção de Firewall:

```
┌──(root㉿kali)-[/home/aluno/Documentos]
└─# nmap -sA 192.168.98.0/24
nmap -sA 192.168.98.0/24
```

O comando mostrado acima usa a técnica de análise stealth (opção -sA) que envia pacotes TCP SYN/ACK (acknowledgment) para determinar se as portas estão abertas, fechadas ou filtradas por firewalls. Esse tipo de escaneamento é conhecido como TCP ACK Scan.

6. O seguinte comando é para encontrar as portas que estão abertas na rede (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**):

```
┌──(root㉿kali)-[/home/aluno/Documentos]
└─# nmap --open 192.168.98.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-06 10:02 -03
Nmap scan report for ip-192-168-98-1.ec2.internal (192.168.98.1)
Host is up (0.000064s latency).
All 1000 scanned ports on ip-192-168-98-1.ec2.internal (192.168.98.1)
are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 12:BF:8B:B8:B0:87 (Unknown)

Nmap scan report for ip-192-168-98-2.ec2.internal (192.168.98.2)
Host is up (0.00013s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT   STATE      SERVICE
53/tcp unfiltered domain
MAC Address: 12:BF:8B:B8:B0:87 (Unknown)

Nmap scan report for ip-192-168-98-20.ec2.internal (192.168.98.20)
Host is up (0.000062s latency).
All 1000 scanned ports on ip-192-168-98-20.ec2.internal (192.168.98.20)
are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 12:89:91:6B:9F:03 (Unknown)

Nmap scan report for ip-192-168-98-30.ec2.internal (192.168.98.30)
Host is up (0.000039s latency).
All 1000 scanned ports on ip-192-168-98-30.ec2.internal (192.168.98.30)
are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 12:33:11:FB:DE:ED (Unknown)

Nmap scan report for ip-192-168-98-201.ec2.internal (192.168.98.201)
Host is up (0.00031s latency).
All 1000 scanned ports on ip-192-168-98-201.ec2.internal
(192.168.98.201) are in ignored states.
Not shown: 1000 unfiltered tcp ports (reset)
MAC Address: 12:CA:37:E3:5B:9B (Unknown)

Nmap scan report for ip-192-168-98-40.ec2.internal (192.168.98.40)
Host is up (0.000016s latency).
All 1000 scanned ports on ip-192-168-98-40.ec2.internal (192.168.98.40)
are in ignored states.
Not shown: 1000 unfiltered tcp ports (reset)

Nmap done: 256 IP addresses (6 hosts up) scanned in 12.73 seconds
```

Veja o que foi encontrado:

- **Host 192.168.98.1**: Estado: Online. Latência: 0.000064 segundos. Serviços: Nenhuma porta aberta detectada. As 1000 portas escaneadas estão em estado filtrado (não houve resposta), o que sugere a presença de um firewall ou filtro de pacotes. Endereço MAC: 12:BF:8B:B8:B0:87 (Desconhecido).
- **Host 192.168.98.2**: Estado: Online. Latência: 0.00013 segundos. Serviços: Porta 53/tcp: Estado unfiltered (não filtrado), serviço domain (DNS). Apesar de a porta não estar classificada como "open", o estado unfiltered indica que a porta é acessável, mas não foi possível confirmar com certeza absoluta se está aberta. Isso pode ocorrer devido à falta de resposta conclusiva no protocolo. Endereço MAC: 12:BF:8B:B8:B0:87 (Desconhecido).
- **Host 192.168.98.20**: Estado: Online. Latência: 0.000062 segundos. Serviços: Nenhuma porta aberta detectada. Todas as 1000 portas escaneadas estão em estado filtrado (no-response), indicando que o host pode estar protegido por firewall ou não respondeu aos pacotes de varredura. Endereço MAC: 12:89:91:6B:9F:03 (Desconhecido).
- **Host 192.168.98.30**: Estado: Online. Latência: 0.000039 segundos. Serviços: Nenhuma porta aberta detectada. Assim como no host anterior, todas as 1000 portas escaneadas estão marcadas como filtradas, sem resposta, o que dificulta a detecção de serviços ativos. Endereço MAC: 12:33:11:FB:DE:ED (Desconhecido).
- **Host 192.168.98.201**: Estado: Online. Latência: 0.00031 segundos. Serviços: Nenhuma porta aberta detectada. As 1000 portas escaneadas estão em estado unfiltered com resposta reset, o que indica que as portas estão acessíveis, mas fechadas (o sistema responde com pacotes RST). Isso sugere que não há serviços ativos nessas portas ou que apenas um subconjunto limitado está em uso, mas não detectado como "aberto". Endereço MAC: 12:CA:37:E3:5B:9B (Desconhecido).
- **Host 192.168.98.40**: Estado: Online. Latência: 0.000016 segundos. Serviços: Nenhuma porta aberta detectada. Todas as 1000 portas escaneadas estão em estado unfiltered e respondem com reset, indicando que o host está ativo e respondendo, mas nenhuma das portas está aberta ou aceitando conexões. Isso é comum em máquinas com firewall ativo bloqueando conexões externas.

7. Finalmente, para pegar os pacotes enviados na rede temos o seguinte comando:

```
┌──(root㉿kali)-[/home/aluno/Documentos]
└─# nmap --packet-trace 192.168.98.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-09-06 10:24 -03
SENT (0.0460s) ARP who-has 192.168.98.1 tell 192.168.98.40
SENT (0.0461s) ARP who-has 192.168.98.2 tell 192.168.98.40

...


RCVD (13.9826s) TCP 192.168.98.40:50849 > 192.168.98.40:18988 S ttl=39
id=48268 iplen=44  seq=2937192391 win=1024 <mss 1460>
RCVD (13.9826s) TCP 192.168.98.40:18988 > 192.168.98.40:50849 RA ttl=64
id=0 iplen=40  seq=0 win=0
Nmap scan report for ip-192-168-98-40.ec2.internal (192.168.98.40)
Host is up (0.000013s latency).
Not shown: 998 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
3389/tcp open  ms-wbt-server

Nmap done: 256 IP addresses (6 hosts up) scanned in 14.03 seconds
```

A opção --packet-trace exibe informações detalhadas sobre os pacotes enviados e recebidos durante o escaneamento. Isso inclui detalhes sobre cada pacote, como cabeçalhos e dados, proporcionando uma visão mais profunda do processo de comunicação entre o scanner (Nmap) e os hosts alvo. Essa opção é útil para análises avançadas e depuração, permitindo que os usuários examinem o tráfego de rede gerado durante o escaneamento.

8. Apague o arquivo “ips.txt” e feche o Terminal:

```
┌──(root㉿kali)-[/home/aluno/Documentos]
└─# ls
ips.txt

┌──(root㉿kali)-[/home/aluno/Documentos]
└─# rm ips.txt
```

Parabéns! Agora você conhece o nmap e os comandos básicos que pode usar com ele. Para maiores informações do nmap, acesse:

```
https://nmap.org/
```

Atividade 3.3 – Interceptando tráfego de navegador com o Burp Suite no Kali Linux

Nesta atividade, vamos explorar a ferramenta Burp Suite para interceptar conteúdo acessado pelo Mozilla Firefox no Kali Linux. O Burp Suite é uma ferramenta de teste de segurança amplamente utilizada por profissionais de segurança cibernética e testadores de penetração. Desenvolvido pela PortSwigger, o Burp Suite oferece um conjunto abrangente de recursos para avaliação de segurança em aplicações web. Ele inclui um proxy interceptador que permite analisar e modificar o tráfego entre o navegador e o servidor, scanners de vulnerabilidades automáticos para identificar falhas de segurança, e diversas ferramentas para realizar testes manuais de segurança, como intrusão de dados e manipulação de requisições. Todo material apresentado aqui deve ser usado somente para fins acadêmicos.

Vamos começar inicializando nosso Kali Linux via RDP ao IP: 192.168.98.40, com usuário “aluno” e senha “rnpesr”.

1. Abra o Terminal e execute o seguinte comando com o usuário “aluno” para abrir o aplicativo Burp Suite (clique em “OK” no aviso de versão do JRE):

```
┌──(aluno㉿kali)-[~]
└─$ burpsuite
```

2. Aceite o aviso referente à versão do JRE marcando o campo "*Don´t show again for the JRE*" e clicando em “OK”. Aceite os Termos e Condições clicando em “I Accept”.
3. Verifique que "Temporary Project in memory" está selecionado e clique em "Next".
4. Selecione "Use Burp defaults", clique em "Start Burp".
5. Clique na aba superior "Proxy". Veja que a interceptação está desabilitada “Intercept is off”.

6. Clique em "Proxy Settings". Veja que em "Proxy listeners" há uma interface selecionada como "Running" na interface 127.0.0.1:8080.
7. Abra o Mozilla Firefox (em “Aplicativos” → “Navegador web)”. Acesse o seguinte site "http://burp", veja que não abre nenhuma página.
8. Ainda no Firefox, no canto superior direito, clique no menu de 3 linhas e em "Settings".
9. Clique no campo "Find in Settings" e escreva "Proxy".
10. Clique em "Settings" de “Network Settings”.
11. Selecione "Manual proxy configuration" e no campo "HTTP Proxy", insira o IP da interface:

```
127.0.0.1
```

12. No campo “Port” insira “8080”. Ative a caixa “Also use this proxy for HTTPS”. Clique em "OK".
13. Ainda no Mozila, abra uma nova aba e tente novamente acessar "http://burp" (cole burp exatamente com o http://). Desta vez, veja que o site do Burp Suite é aberto:

```
Burp Suite Community Edition    CA Certificate
Welcome to Burp Suite Community Edition.
```

14. Clique no botão “CA Certificate” e veja que é baixado o arquivo “cacert.der” na pasta “Downloads”.
15. Volte à aba “Settings” do Firefox e clique em “Privacy & Security” (coluna esquerda) → no campo Certificates “View Certificates” → “Import” → selecione o arquivo “cacert.der” → “Abrir”.
16. Na janela aberta “Downloading Certificate”, ative as duas caixas que começam com “Trust this CA to...” e clique em “OK” 2 vezes.
17. Retomando o passo 6 (no programa Burp Suite), feche a janela "Settings" do Burp Suite e volte a janela principal do programa Burp Suite.
18. Veja que na aba "Proxy" o botão "Intercept is off" está sendo mostrado (canto superior esquerdo). Clique nele e veja que agora o botão está como "Intercept is on", agora a interceptação está funcionando.

19. Volte ao Mozilla Firefox, abra uma nova aba e acesse o site “https://casasbahia.com.br/”. Veja que você é direcionado ao programa Burp Suite e que o site não abre no Firefox! Isso é porque o Burp Suite interceptou a comunicação para que você avalie o conteúdo.
20. No Burp Suite, veja que os detalhes da conexão estão sendo mostrados, como Host, Cooikie, User-Agent, etc.
21. Clique na aba "HTTP history" e veja como foi interceptada a requisição do Mozilla Firefox (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**).
22. Clique na aba “Intercept” e em seguida no botão “Forward” para que seja liberado o acesso ao site no Firefox. Veja no Firefox que agora sim o site abre!
23. Refaça os passos 8, 9 e 10. Selecione “No Proxy” e “OK”.
24. Feche o Burp Suite, o Firefox e o Terminal.

Parabéns! Agora você sabe como usar o Burp Suite para interceptar tráfego de navegador via Proxy!

Atividade 3.4 – Escutando requisições com o Netcat no Kali Linux

Nesta atividade, vamos verificar requisições com a ferramenta Netcat (nc) no Kali Linux. O Netcat, também conhecido como "nc", é uma poderosa ferramenta de linha de comando utilizada para comunicação de dados através de redes, seja TCP ou UDP. Originalmente desenvolvido para Unix, o Netcat permite a criação de conexões de rede, transferência de dados, e até mesmo a criação de servidores simples. Sua versatilidade o torna uma ferramenta valiosa em testes de segurança, programação de redes e outras atividades relacionadas à manipulação de dados em ambientes de rede. Todo material apresentado aqui deve ser usado somente para fins acadêmicos.

Vamos começar inicializando nosso Kali Linux via RDP ao IP: 192.168.98.40, com usuário “aluno” e senha “rnpesr”.

1. Abra o Terminal e execute o seguinte comando (com senha “rnpesr”) para ser super usuário:

```
┌──(aluno㉿kali)-[~]
└─$ sudo -i
[sudo] senha para aluno:
```

2. Veja os modos de funcionamento que o Netcat permite usar:

```
┌──(root㉿kali)-[~]
└─# nc -help
OpenBSD netcat (Debian patchlevel 1.229-1)
usage: nc [-46CDdFhklNnrStUuvZz] [-I length] [-i interval] [-M ttl]
          [-m minttl] [-O length] [-P proxy_username] [-p source_port]
          [-q seconds] [-s sourceaddr] [-T keyword] [-V rtable] [-W
recvlimit]
          [-w timeout] [-X proxy_protocol] [-x proxy_address[:port]]
          [destination] [port]
        Command Summary:
                -4              Use IPv4
                -6              Use IPv6

...
```

3. Descubra seu IP:

```
┌──(root㉿kali)-[~]
└─# ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:df:26:13:80  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 192.168.98.40  netmask 255.255.255.0  broadcast
192.168.98.255
        inet6 fe80::1005:27ff:fe44:1631  prefixlen 64  scopeid
0x20<link>
        ether 12:05:27:44:16:31  txqueuelen 1000  (Ethernet)
        RX packets 1818194  bytes 2694820836 (2.5 GiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 61082  bytes 77348531 (73.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Loopback Local)
        RX packets 23  bytes 1937 (1.8 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 23  bytes 1937 (1.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

4. Agora, vamos configurar o Netcat para que fique escutando requisições:

```
┌──(root㉿kali)-[~]
└─# nc -l -p 5555 -v
listening on [any] 5555 ...
```

O comando mostrado acima tem a seguinte saída:

- **nc**: Representa o próprio executável do Netcat.
- **-l**: Indica que o Netcat deve operar como um servidor de escuta.
- **-p 5555**: Especifica a porta (porta 5555) na qual o Netcat estará escutando conexões.

- **-v**: Ativa o modo verbose, exibindo informações detalhadas durante a execução.
- **A mensagem "listening on [any] 5555 ..."** indica que o Netcat está aguardando por conexões em qualquer endereço disponível no sistema, na porta 5555. Este comando é frequentemente usado para criar um servidor que escuta conexões em uma determinada porta para fins de comunicação ou testes de rede.

5. No canto superior esquerdo, clique em “Aplicativos” → “Navegador web” para abrir o Mozilla Firefox. Acesse o seguinte site referente ao IP da sua máquina:

```
http://192.168.98.40:5555
```

6. Volte ao Terminal e veja que o Netcat capturou a tentativa de conexão (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**):

```
┌──(root㉿kali)-[~]
└─# nc -l -p 5555 -v
Listening on 0.0.0.0 5555
Connection received on ip-192-168-98-40.ec2.internal 42396
GET / HTTP/1.1
Host: 192.168.98.40:5555
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101
Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Priority: u=0, i
```

O resultado da escuta é:

- **connect to [192.168.98.40] from ip-192-168-98-40.ec2.internal [192.168.98.40] 42396**: Indica que uma conexão foi estabelecida com o servidor na porta 5555 a partir do endereço IP 192.168.98.40. O número de porta do cliente é 42396.
- **GET / HTTP/1.1**: Este é um cabeçalho de solicitação HTTP enviado pelo cliente para o servidor. Ele indica que o cliente está solicitando o recurso raiz ("/") usando o método GET e a versão HTTP 1.1.

- **Host: 192.168.98.40:5555**: Este cabeçalho indica o host e a porta para os quais a solicitação está sendo feita. Neste caso, o cliente está se comunicando com o host 192.168.98.40 na porta 5555.
- **User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0**: Este cabeçalho contém informações sobre o navegador ou o agente do usuário que está fazendo a solicitação. Aqui, o agente do usuário é identificado como Firefox na plataforma Linux.
- **Accept, Accept-Language, Accept-Encoding, Connection, Upgrade-Insecure-Requests, Priority**: Estes são cabeçalhos HTTP adicionais que fornecem informações sobre as preferências do cliente em relação ao conteúdo que ele está disposto a aceitar, o idioma preferido, a codificação de conteúdo, a conexão e a indicação de que o cliente pode aceitar atualizações de segurança de protocolo.

7. Feche o Firefox e o Terminal.

Parabéns! Agora você conhece como escutar requisições que entram ao seu computador por uma determinada porta, isso inclui conexões de resposta de acesso a sites que você visite.

Atividade 3.5 – Redirecionando tráfego via portas com o Netcat no Kali Linux

Nesta atividade, vamos redirecionar tráfego, direcionado originalmente para a porta 80, para a porta 443 com a ferramenta Netcat (ncat) no Kali Linux. Todo material apresentado aqui deve ser usado somente para fins acadêmicos.

Vamos começar inicializando nosso Kali Linux via RDP ao IP: 192.168.98.40, com usuário “aluno” e senha “rnpesr”.

1. Abra o Terminal e execute o seguinte comando (com senha “rnpesr”) para ser super usuário:

```
┌──(aluno㉿kali)-[~]
└─$ sudo -i
[sudo] senha para aluno:
```

2. Execute o comando de redirecionamento de tráfego da porta 80 para a porta 443:

```
┌──(root㉿kali)-[~]
└─# ncat -vl 80 -c 'ncat -l 443'
Ncat: Version 7.95 ( https://nmap.org/ncat )
Ncat: Listening on [::]:80
Ncat: Listening on 0.0.0.0:80
```

O comando ncat -vl 80 -c 'ncat -l 443' utiliza o Ncat (uma versão avançada do Netcat) para criar uma situação em que as conexões recebidas na porta 80 serão redirecionadas para a porta 443. O comando é composto por:

- **ncat**: Chama o programa Ncat para iniciar a escuta e manipulação de conexões.
- **-vl 80**: As opções -v e -l indicam que o Ncat deve operar em modo de escuta e exibir informações verbosas. O 80 especifica a porta em que o Ncat estará escutando conexões.
- **-c 'ncat -l 443'**: Esta opção especifica o comando a ser executado quando uma conexão é recebida. No caso, está utilizando o Ncat novamente para escutar na porta 443.

O resultado exibido pelo comando indica:

- **Ncat: Version 7.95 (https://nmap.org/ncat)**: Informa a versão do Ncat utilizada.
- **Ncat: Listening on [::]:80**: Indica que o Ncat está escutando conexões na porta 80 em todos os endereços IPv6 disponíveis no sistema.
- **Ncat: Listening on 0.0.0.0:80**: Indica que o Ncat está escutando conexões na porta 80 em todos os endereços IPv4 disponíveis no sistema.

3. Abra um segundo terminal e estabeleça uma conexão na porta 80:

```
┌──(aluno㉿kali)-[~]
└─$ ncat -nv 127.0.0.1 80
Ncat: Version 7.95 ( https://nmap.org/ncat )
Ncat: Connected to 127.0.0.1:80.
```

O comando mostrado acima cria uma conexão para o servidor na porta 80 no próprio sistema (localhost) e exibe informações verbosas sobre a conexão bem-sucedida:

- **ncat**: Chama o programa Ncat para realizar operações de conexão de rede.
- **-nv**: As opções -n e -v indicam que o Ncat deve suprimir a resolução de DNS (evitando pesquisa de nome) e exibir informações verbosas sobre a conexão, respectivamente.
- **127.0.0.1**: Indica o endereço IP para o qual a conexão será estabelecida, neste caso, o localhost (o próprio sistema).
- **80**: Especifica a porta para a qual a conexão será feita, neste caso, a porta 80.

O resultado exibido pelo comando indica:

- **Ncat: Version 7.95 (https://nmap.org/ncat)**: Informa a versão do Ncat utilizada.
- **Ncat: Connected to 127.0.0.1:80**: Indica que o Ncat estabeleceu com sucesso uma conexão para o endereço IP 127.0.0.1 na porta 80.

4. Volte ao primeiro terminal e veja que a conexão do passo 3 foi identificada (a porta pode ser diferente):

```
┌──(root㉿kali)-[~]
└─# ncat -vl 80 -c 'ncat -l 443'
Ncat: Version 7.95 ( https://nmap.org/ncat )
Ncat: Listening on [::]:80
Ncat: Listening on 0.0.0.0:80
Ncat: Connection from 127.0.0.1:59584.
```

5. Abra um terceiro terminal e estabeleça conexão com a porta 443 do localhost:

```
┌──(aluno㉿kali)-[~]
└─$ ncat -v 127.0.0.1 443
Ncat: Version 7.95 ( https://nmap.org/ncat )
Ncat: Connected to 127.0.0.1:443.
```

6. Veja que o segundo terminal estabeleceu uma conexão com o localhost pela porta 80 e que o terceiro terminal estabeleceu uma conexão com o localhost pela porta 443. Pelo fato de haver encaminhamento entre as portas 80 e 443, o segundo terminal poderá estabelecer comunicação com o terceiro terminal. Vamos testar!
7. Continue no Terceiro terminal e escreva “Teste A” e aperte Enter:

```
┌──(aluno㉿kali)-[~]
└─$ ncat -v 127.0.0.1 443
Ncat: Version 7.94SVN ( https://nmap.org/ncat )
Ncat: Connected to 127.0.0.1:443.
Teste A
```

8. Volte ao segundo Terminal e veja que “Teste A” é mostrado estabelecendo um chat! Ainda no segundo Terminal, escreva “Teste B” e aperte Enter:

```
┌──(aluno㉿kali)-[~]
└─$ ncat -nv 127.0.0.1 80
Ncat: Version 7.94SVN ( https://nmap.org/ncat )
Ncat: Connected to 127.0.0.1:80.
Teste A
Teste B
```

9. Volte ao terceiro Terminal e veja que “Teste B” também é mostrado (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**).
10. Feche os Terminais.

Parabéns! Agora você sabe como redirecionar tráfego entre duas portas usando o Ncat!