Report: Penetration Testing di Autenticazione con Hydra

Data: 9 Maggio 2025

Autore: Stefano Gugliotta

1. Introduzione

Questo report documenta una prova di penetration testing che si è concentrato sull'analisi della sicurezza dell'autenticazione di servizi di rete, utilizzando lo strumento Hydra. L'obiettivo principale di questo test è stato duplice: da un lato, acquisire familiarità con l'utilizzo di Hydra per il cracking delle credenziali e, dall'altro, rafforzare la comprensione di come i servizi di rete sono configurati.

L'esercizio è stato strutturato in due fasi distinte. Nella prima fase, abbiamo esaminato da vicino il servizio SSH, attivandolo e poi tentando di craccarne l'autenticazione con Hydra. Nella seconda fase, l'attenzione si è spostata su un altro servizio di rete, in questo caso FTP, che è stato configurato e successivamente sottoposto a un tentativo di cracking dell'autenticazione.

2. Strumenti Utilizzati

Per portare a termine questo esercizio, sono stati utilizzati i seguenti strumenti:

- **Kali Linux:** Questo sistema operativo è stato la piattaforma di elezione per l'esecuzione di tutti i test.
- **Hydra:** Questo strumento è stato impiegato per tentare di craccare le credenziali di autenticazione dei servizi di rete.
- **vsftpd:** Questo server FTP è stato utilizzato per configurare il servizio FTP che è stato poi oggetto del tentativo di cracking.

3. Procedura

L'esercizio è stato condotto in due fasi differenti, la prima utilizzando l’autenticazione SSH e successivamente utilizzando l’autenticazione FTP.

3.1 Fase 1: Cracking dell'autenticazione SSH

1. Creazione di un nuovo utente: Inizialmente, è stato creato un nuovo utente di

sistema, denominato "*test_user*", sulla macchina Kali Linux. A questo utente è stata assegnata una password iniziale, "*testpass*", utilizzando il comando `adduser`.

```
┌──(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
```

2. Abilitazione del servizio SSH: Successivamente, è stato attivato il servizio SSH sulla macchina Kali Linux. Questo è stato realizzato tramite il comando `service ssh start`. Il file di configurazione del demone SSH, situato in `/etc/ssh/sshd_config`, è stato intenzionalmente lasciato con le impostazioni predefinite per questo esercizio.

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

┌──(kali㉿kali)-[~]
└─$ sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: disabled)
     Active: active (running) since Fri 2025-05-09 03:54:06 EDT; 1min 6s ago
 Invocation: 1eb8744955e44552b740ad8b60756594
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 11602 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 11605 (sshd)
      Tasks: 1 (limit: 2214)
     Memory: 2.3M (peak: 2.7M)
        CPU: 18ms
     CGroup: /system.slice/ssh.service
             └─11605 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

May 09 03:54:06 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
May 09 03:54:06 kali sshd[11605]: Server listening on 0.0.0.0 port 22.
May 09 03:54:06 kali sshd[11605]: Server listening on :: port 22.
May 09 03:54:06 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.

┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b4:a1:05 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.83/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
       valid_lft 85064sec preferred_lft 85064sec
    inet6 fe80::b6a3:7108:4943:58ee/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

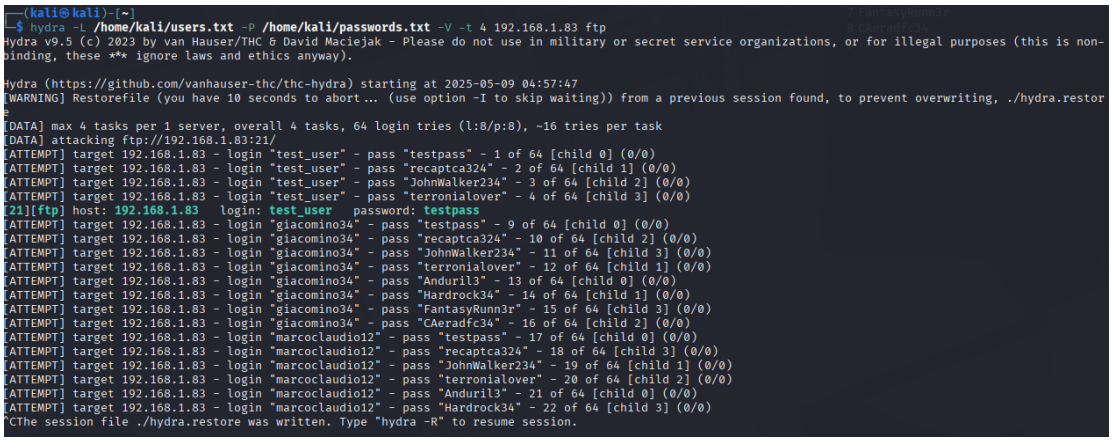
3. Test della connessione SSH: A questo punto, è stata verificata la capacità di connettersi al server SSH utilizzando le credenziali dell'utente appena creato. Il comando utilizzato per questo scopo è stato `ssh test_user@192.168.1.83`. La connessione è stata stabilita con successo, confermando che l'utente e la password erano validi.

4. **Cracking dell'autenticazione SSH con Hydra:**

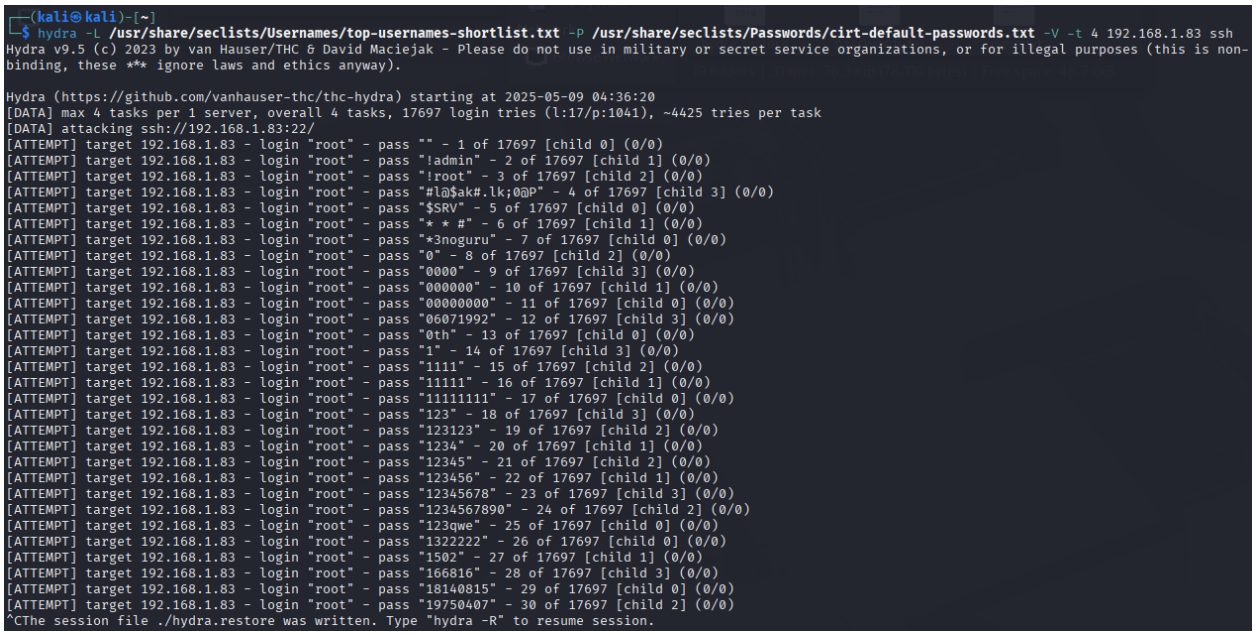
Una volta verificata la connettività SSH, l'attenzione si è spostata sull'utilizzo di Hydra per tentare di craccare l'autenticazione.

- Sono stati creati due file di testo personalizzati, "users.txt" e "passwords.txt". Il file "users.txt" conteneva un elenco di 8 possibili nomi utente, mentre il file "passwords.txt" conteneva 8 possibili password. È importante notare che solo una combinazione di username e password in questi file era corretta.
- Hydra è stato quindi impiegato per eseguire un attacco a dizionario contro il servizio SSH. L'attacco ha specificato i file "users.txt" e "passwords.txt" come

input per Hydra, indicando le possibili credenziali da testare, trovando username e password “*testuser*” e “*testpass*” come match.



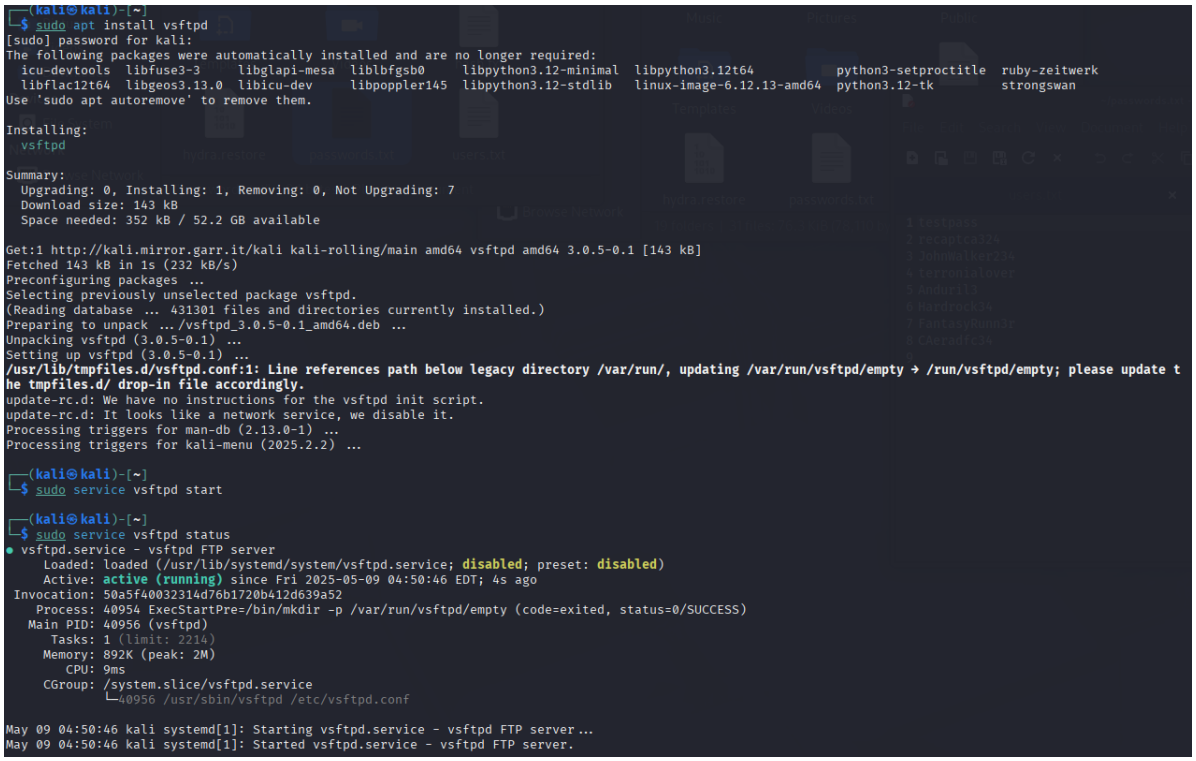
- Inoltre, è stato effettuato un tentativo di utilizzare le wordlist predefinite fornite dal pacchetto "seclists". Questo pacchetto, che contiene ampie raccolte di possibili nomi utente e password, è stato installato in precedenza tramite il comando `apt install seclists`. L’utilizzo di questo pacchetto di wordlists, però, non ha prodotto in questo caso risultati positivi in quanto l’username e la password ricercate non sono comuni all’interno delle wordlists, risultando in un interruzione del comando.



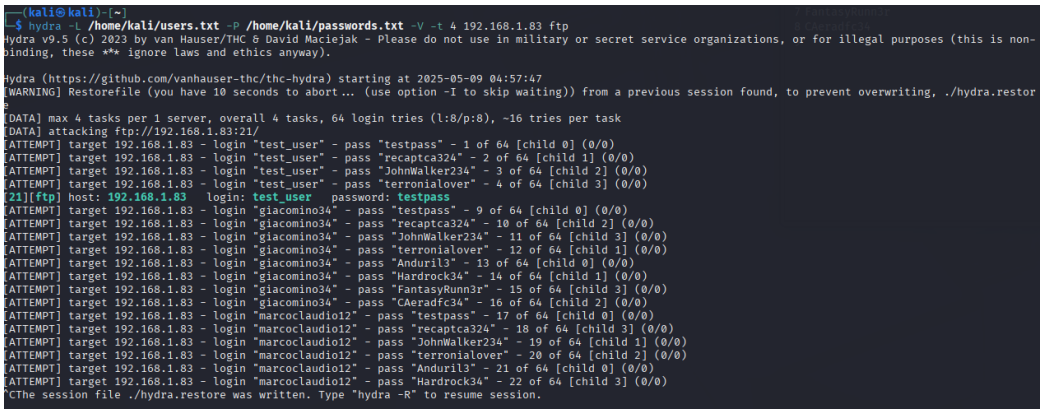
3.2 Fase 2: Cracking dell'autenticazione FTP

1. Installazione del servizio FTP: Per questa fase, è stato installato il server FTP "vsftpd" sulla macchina Kali Linux. L'installazione è stata eseguita utilizzando il comando `apt install vsftpd`, e il servizio è stato successivamente avviato tramite il

comando `service vsftpd start`.



2. Configurazione del servizio FTP: Dopo l'installazione, è stata verificata la configurazione del servizio FTP. In particolare, è stato controllato che l'accesso anonimo fosse disabilitato. Questo controllo è stato effettuato esaminando il file di configurazione `/etc/vsftpd.conf` e verificando che l'opzione `anonymous_enable` fosse impostata su `NO`.
3. Test della connessione FTP: Prima di procedere con il tentativo di cracking, è stata verificata la capacità di connettersi al server FTP locale. Questo test è stato eseguito utilizzando un client FTP.
4. Cracking dell'autenticazione FTP con Hydra: Infine, Hydra è stato impiegato per eseguire un attacco a dizionario contro il servizio FTP. Per questo attacco, sono stati riutilizzati gli stessi file "users.txt" e "passwords.txt" che erano stati precedentemente creati per la fase di cracking SSH, evitando di utilizzare le wordlists di seclists in quanto non avrebbero prodotto alcun risultato.



4. Risultati

L'esercizio ha prodotto i seguenti risultati:

- SSH: Quando Hydra è stato configurato per utilizzare i file "users.txt" e "passwords.txt" personalizzati, è stato in grado di identificare con successo la combinazione corretta di username e password ("test_user"/"testpass"). Tuttavia, l'utilizzo delle wordlist predefinite fornite da "seclists" non ha prodotto risultati positivi in un lasso di tempo ragionevole.
- FTP: Analogamente a quanto osservato con SSH, Hydra è stato in grado di identificare la combinazione corretta di username e password per il servizio FTP quando sono stati forniti i file "users.txt" e "passwords.txt" personalizzati.

5. Conclusioni

In conclusione, questo esercizio ha dimostrato l'efficacia di Hydra come strumento per il cracking dell'autenticazione di servizi di rete, in particolare SSH e FTP. Un'osservazione importante emersa da questo esercizio è che la creazione di wordlist mirate e di dimensioni ridotte può migliorare significativamente la velocità e l'efficacia del processo di cracking rispetto all'utilizzo di wordlist generiche e di grandi dimensioni.

6. Raccomandazioni

Sulla base dei risultati di questo esercizio, si raccomandano le seguenti misure per migliorare la sicurezza dei sistemi:

- Sicurezza delle password: È fondamentale che gli utenti utilizzino password complesse e univoche per proteggere i propri account e i servizi di rete a cui accedono.
- Configurazione dei servizi: I servizi di rete devono essere configurati con attenzione, disabilitando l'accesso anonimo quando non è necessario e implementando meccanismi per limitare il numero di tentativi di accesso falliti.

- Monitoraggio e rilevamento: È essenziale implementare sistemi di monitoraggio efficaci per rilevare attività di cracking sospette o altri comportamenti anomali.
- Valutazione continua: I test di penetrazione dovrebbero essere eseguiti regolarmente per identificare e correggere tempestivamente eventuali vulnerabilità nei sistemi.