

Relazione Attacco alle Web App (SQL Injection)

Nell'immagine seguente entro come amministratore di sistema (**root**) con il comando **sudo -s**, e con il comando **adduser**, aggiungo un nuovo utente:

- **adduser test_user**

```
(root@kali)-[~]
# 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 []: test_user
    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' ...
```

Nella seguente immagine con il comando **sudo service ssh start**, attiviamo il servizio **ssh**:

- **sudo service ssh start**

Successivamente utilizzo il comando **ssh test_user@192.168.50.154**, con il quale mi colleghero all'indirizzo IP specifico, come si può vedere da comando, illustrato

```
(root@kali)-[~]
# sudo service ssh start

(root@kali)-[~]
# ssh test_user@192.168.50.154
test_user@192.168.50.154'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 Sep 20 03:51:25 2024 from 192.168.50.154
```

Prima di descrivere il prossimo passaggio, vado ad utilizzare il comando **nano** per la creazione e la modifica del nostro/nostri file, nel nostro caso creiamo due uno chiamato **admin.txt** che contiene gli admin, ed un file **password.txt**, contenente le password degli admin, come da seguente immagine:

```
(kali@kali)-[~]  
$ nano usernames.txt  
  
(kali@kali)-[~]  
$ nano password.txt
```

Nella seguente immagine e successivo passaggio al comando che ci collega all'IP **192.168.50.154**, utilizziamo il comando **hydra -L admin.txt -P password.txt -t 4 ssh**, per poter estrapolare i dati contenuti nei **file.txt (user e password)**, con i permessi del nuovo utente creato, **test_user@kali**

```
(test_user@kali)-[~]  
$ hydra -L admin.txt -P password.txt 192.168.50.154 -t 4 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 2024-09-20 05:04:23  
[DATA] max 4 tasks per 1 server, overall 4 tasks, 36 login tries (1:6/p:6), ~9 tries per task  
[DATA] attacking ssh://192.168.50.154:22/  
[22][ssh] host: 192.168.50.154 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 2024-09-20 05:04:50
```

Nella seguente immagine e successivo passaggio al comando che ci collega all'IP **192.168.50.154**, utilizziamo il comando **hydra -L admin.txt -P password.txt -t 4 ssh**, per poter estrapolare i dati contenuti nei **file.txt (user e password)**, questa volta con i permessi dell'utente **kali_kali@**

Eseguendo il comando hydra, avrò la possibilità di visualizzare il **log in (test_user)** e la **password (testpass)**

```
(kali@kali)-[~]  
$ hydra -L admin.txt -P password.txt 192.168.50.154 -t 4 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 2024-09-20 05:19:11  
[DATA] max 4 tasks per 1 server, overall 4 tasks, 30 login tries (1:5/p:6), ~8 tries per task  
[DATA] attacking ssh://192.168.50.154:22/  
[22][ssh] host: 192.168.50.154 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 2024-09-20 05:19:32
```

Nella seguente immagine ed ultimo passaggio, creiamo altri **file.txt**, uno contenente gli user, **usernames.txt** ed un file, contenente le password, **password.txt**.

- utilizziamo il comando **hydra -L usernames.txt -P password.txt ftp://127.0.0.1**

Con il comando precedentemente illustrato, mi collego al mio IP localhost (127.0.0.1), con il servizio **ftp**

Eseguendo il comando hydra, avrò la possibilità di visualizzare il **log in (test_user)** e la **password (testpass)**

```
(kali@kali)-[~]
$ service vsftpd start

(kali@kali)-[~]
$ nano usernames.txt

(kali@kali)-[~]
$ nano password.txt

(kali@kali)-[~]
$ hydra -L usernames.txt -P password.txt ftp://127.0.0.1
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 2024-09-20 06:58:38
[DATA] max 16 tasks per 1 server, overall 16 tasks, 42 login tries (l:6/p:7), ~3 tries per task
[DATA] attacking ftp://127.0.0.1:21/
[21][ftp] host: 127.0.0.1  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 2024-09-20 06:58:49
```