hydra | Kali Linux Tools

Archived: 2026-04-05 21:46:46 UTC

Tool Documentation:

hydra Usage Example

Attempt to login as the root user (`-l root`) using a password list (`-P /usr/share/wordlists/metasploit/unix_passwords.txt`) with 6 threads (`-t 6`) on the given SSH server (`ssh://192.168.1.123`):

```
 root@kali:~# hydra -l root -P /usr/share/wordlists/metasploit/unix_passwords.txt -t 6 ssh://192.168.1.123
Hydra v7.6 (c)2013 by van Hauser/THC & David Maciejak - for legal purposes only

Hydra (http://www.thc.org/thc-hydra) starting at 2014-05-19 07:53:33
[DATA] 6 tasks, 1 server, 1003 login tries (l:1/p:1003), ~167 tries per task
[DATA] attacking service ssh on port 22
```

pw-inspector Usage Example

Read in a list of passwords (`-i /usr/share/wordlists/nmap.lst`) and save to a file (`-o /root/passes.txt`), selecting passwords of a minimum length of 6 (`-m 6`) and a maximum length of 10 (`-M 10`):

```
 root@kali:~# pw-inspector -i /usr/share/wordlists/nmap.lst -o /root/passes.txt -m 6 -M 10
root@kali:~# wc -l /usr/share/wordlists/nmap.lst
5086 /usr/share/wordlists/nmap.lst
root@kali:~# wc -l /root/passes.txt
4490 /root/passes.txt
```

Packages and Binaries:

hydra

Very fast network logon cracker
Hydra is a parallelized login cracker which supports numerous protocols to attack. It is very fast and flexible, and new modules are easy to add.

This tool makes it possible for researchers and security consultants to show how easy it would be to gain unauthorized access to a system remotely.

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.

Installed size: `964 KB`
How to install: `sudo apt install hydra`

▶ Dependencies:

dpl4hydra

Generates a (d)efault (p)assword (l)ist as input for THC hydra

```
 root@kali:~# dpl4hydra -h
dpl4hydra v0.9.9 (c) 2012 by Roland Kessler (@rokessler)

Syntax: dpl4hydra [help] | [refresh] | [BRAND] | [all]

This script depends on a local (d)efault (p)assword (l)ist called
/root/.dpl4hydra/dpl4hydra_full.csv. If it is not available, regenerate it with
'dpl4hydra refresh'. Source of the default password list is
http://open-sez.me

Options:
  help        Help: Show this message
  refresh     Refresh list: Download the full (d)efault (p)assword (l)ist
              and generate a new local /root/.dpl4hydra/dpl4hydra_full.csv file. Takes time!
  BRAND       Generates a (d)efault (p)assword (l)ist from the local file
              /root/.dpl4hydra/dpl4hydra_full.csv, limiting the output to BRAND systems, using
              the format username:password (as required by THC hydra).
              The output file is called dpl4hydra_BRAND.lst.
  all         Dump list of all systems credentials into dpl4hydra_all.lst.

Example:
# dpl4hydra linksys
File dpl4hydra_linksys.lst was created with 20 entries.
# hydra -C ./dpl4hydra_linksys.lst -t 1 192.168.1.1 http-get /index.asp
```

---

hydra

A very fast network logon cracker which supports many different services

```
 root@kali:~# hydra -h
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organiz

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e nsr] [-o FILE] [-t TASKS] [-M FILE [-T TA

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
  -r        use a non-random shuffling method for option -x
  -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
  -D XofY   Divide wordlist into Y segments and use the Xth segment.
  -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)
  -v / -V / -d  verbose mode / show login+pass for each attempt / debug mode
  -O        use old SSL v2 and v3
  -K        do not redo failed attempts (good for -M mass scanning)
  -q        do not print messages about connection errors
  -U        service module usage details
  -m OPT    options specific for a module, see -U output for information
  -h        more command line options (COMPLETE HELP)
  server    the target: DNS, IP or 192.168.0.0/24 (this OR the -M option)
  service   the service to crack (see below for supported protocols)
  OPT       some service modules support additional input (-U for module help)

Supported services: adam6500 asterisk cisco cisco-enable cobaltstrike cvs firebird ftp[s] http[s]-{head|get|post

Hydra is a tool to guess/crack valid login/password pairs.
Licensed under AGPL v3.0. The newest version is always available at;
https://github.com/vanhauser-thc/thc-hydra
Please don't use in military or secret service organizations, or for illegal
purposes. (This is a wish and non-binding - most such people do not care about
```

```
laws and ethics anyway - and tell themselves they are one of the good ones.)
These services were not compiled in: afp ncp oracle sapr3.

Use HYDRA_PROXY_HTTP or HYDRA_PROXY environment variables for a proxy setup.
E.g. % export HYDRA_PROXY=socks5://l:p@127.0.0.1:9150 (or: socks4:// connect://)
     % export HYDRA_PROXY=connect_and_socks_proxylist.txt  (up to 64 entries)
     % export HYDRA_PROXY_HTTP=http://login:pass@proxy:8080
     % export HYDRA_PROXY_HTTP=proxylist.txt  (up to 64 entries)

Examples:
  hydra -l user -P passlist.txt ftp://192.168.0.1
  hydra -L userlist.txt -p defaultpw imap://192.168.0.1/PLAIN
  hydra -C defaults.txt -6 pop3s://[2001:db8::1]:143/TLS:DIGEST-MD5
  hydra -l admin -p password ftp://[192.168.0.0/24]/
  hydra -L logins.txt -P pws.txt -M targets.txt ssh
```

---

hydra-wizard

Wizard to use hydra from command line

```
 root@kali:~# man hydra-wizard
HYDRA-WIZARD(1)             General Commands Manual             HYDRA-WIZARD(1)

NAME
       HYDRA-WIZARD - Wizard to use hydra from command line

DESCRIPTION
       This  script  guide  users  to use hydra, with a simple wizard that will
       make the necessary questions to launch hydra from command  line  a  fast
       and easily
       1. The wizard ask for the service to attack
       2. The target to attack
       3. The username o file with the username what use to attack
       4. The password o file with the passwords what use to attack
       5.  The wizard ask if you want to test for passwords same as login, null
       or reverse login
       6. The wizard ask for the port number to attack
       Finally, the wizard show the resume information of attack,  and  ask  if
       you want launch attack

SEE ALSO
       hydra(1), dpl4hydra(1),

AUTHOR
       hydra-wizard was written by Shivang Desai <shivang.ice.2010@gmail.com>.
```

```
       This  manual page was written by Daniel Echeverry <epsilon77@gmail.com>,
       for the Debian project (and may be used by others).

                                   19/01/2014                   HYDRA-WIZARD(1)
```

---

pw-inspector

A tool to reduce the password list

```
 root@kali:~# pw-inspector -h
PW-Inspector v0.2 (c) 2005 by van Hauser / THC vh@thc.org [https://github.com/vanhauser-thc/thc-hydra]

Syntax: pw-inspector [-i FILE] [-o FILE] [-m MINLEN] [-M MAXLEN] [-c MINSETS] -l -u -n -p -s

Options:
  -i FILE    file to read passwords from (default: stdin)
  -o FILE    file to write valid passwords to (default: stdout)
  -m MINLEN  minimum length of a valid password
  -M MAXLEN  maximum length of a valid password
  -c MINSETS the minimum number of sets required (default: all given)
Sets:
  -l         lowcase characters (a,b,c,d, etc.)
  -u         upcase characters (A,B,C,D, etc.)
  -n         numbers (1,2,3,4, etc.)
  -p         printable characters (which are not -l/-u/-n, e.g. $,!,/,(,*, etc.)
  -s         special characters - all others not within the sets above

PW-Inspector reads passwords in and prints those which meet the requirements.
The return code is the number of valid passwords found, 0 if none was found.
Use for security: check passwords, if 0 is returned, reject password choice.
Use for hacking: trim your dictionary file to the pw requirements of the target.
Usage only allowed for legal purposes.
```

Updated on: 2025-Dec-09
