Tools included in the hydra package

hydra – Very fast network logon cracker

root@kali:~# hydra -h  
Hydra v7.6 (c)2013 by van Hauser/THC & David Maciejak - for legal purposes only  
  
Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e nsr] [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-W TIME] [-f] [-s PORT] [-x MIN:MAX:CHARSET] [-SuvV46] [service://server[:PORT][/OPT]]  
  
Options:  
  -R        restore a previous aborted/crashed session  
  -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  
  -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 be attacked in parallel, one entry per line  
  -o FILE   write found login/password pairs to FILE instead of stdout  
  -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 host, default: 16)  
  -w / -W TIME  waittime for responses (32s) / between connects per thread  
  -4 / -6   prefer IPv4 (default) or IPv6 addresses  
  -v / -V / -d  verbose mode / show login+pass for each attempt / debug mode  
  -U        service module usage details  
  server    the target server (use either 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: asterisk afp cisco cisco-enable cvs firebird ftp ftps http[s]-{head|get} http[s]-{get|post}-form http-proxy http-proxy-urlenum icq imap[s] irc ldap2[s] ldap3[-{cram|digest}md5][s] mssql mysql ncp nntp oracle-listener oracle-sid pcanywhere pcnfs pop3[s] postgres rdp rexec rlogin rsh s7-300 sip smb smtp[s] smtp-enum snmp socks5 ssh sshkey svn teamspeak telnet[s] vmauthd vnc xmpp  
  
Hydra is a tool to guess/crack valid login/password pairs - usage only allowed  
for legal purposes. This tool is licensed under AGPL v3.0.  
The newest version is always available at http://www.thc.org/thc-hydra  
These services were not compiled in: sapr3 oracle.  
  
Use HYDRA\_PROXY\_HTTP or HYDRA\_PROXY - and if needed HYDRA\_PROXY\_AUTH - environment for a proxy setup.  
E.g.:  % export HYDRA\_PROXY=socks5://127.0.0.1:9150 (or socks4:// or connect://)  
       % export HYDRA\_PROXY\_HTTP=http://proxy:8080  
       % export HYDRA\_PROXY\_AUTH=user:pass  
  
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://[fe80::2c:31ff:fe12:ac11]:143/TLS:DIGEST-MD5

pw-inspector – Reads passwords in and prints those which meet the requirements

root@kali:~# pw-inspector  
PW-Inspector v0.2 (c) 2005 by van Hauser / THC vh@thc.org [http://www.thc.org]  
  
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/-n/-p, e.g. $,!,/,(,\*, etc.)  
  -s         special characters - all others not withint 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.

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