Tool of the Week

...

THC Hydra

By Arslan and Brian

Introduction to THC Hydra

What is THC Hydra?

- THC Hydra is a parallelized login cracker
- Originally created to help researchers and security professionals demonstrate unauthorized remote access
- Supports attacks through a number of services/protocols
- Can target single or multiple machines connected online
- Dictionary and brute-force attacks, with parallel attacking option

Background Information - What is a Password attack?

- One type of software attack where attacker tries
  - To guess and check passwords
  - To crack encrypted passwords either manually or through the use of scripts and other tools

Types of Password Attacks

- Brute Force Attack: Simplest method, generating all possible combinations to crack the password
- Dictionary Attack: A common password cracking method, using a list of potential passwords
- Password Phishing: User disguised as a trustworthy entity to gain access to the systems

Interesting Fact - To crack an 8 character password composed from a set of 96 characters with a speed of 1 billion passwords per second, it would take at least 83.5 days

Power of THC Hydra

- Protocols/services supported by THC Hydra
  - POP3
  - FTP
  - HTTP-GET, HTTP-POST-FORM, HTTP-GET-FORM...
  - Firebird
  - Subversion
  - Telnet
  - Postgres
  - SSH
  - Teamspeak
  - MySQL
  - rexec
  - SOCKS5
  - SNMP
  - NNTP
  - And many more...
- Use `hydra -U protocol_name_here` to find the parameters needed by THC Hydra to attack through a specified protocol/service

Power of THC Hydra

- Types of attacks that THC Hydra can handle
  - Brute force attacks
  - Dictionary attacks
  - Parallel attacks (16 threads by default, `-t` option)
  - Check for null, login as password, reversed characters (`-e` option)
  - Attack several different servers

Online Vs Offline Password Cracking

Offline

- Only possible with access to password hashes
- Time to crack depends on password strength, processing power, and time
- Essentially unlimited number of password cracking attempts

Online

- No access to password hashes
- Possible account lockout
- Several password cracking attempts can be logged and deemed suspicious
- Time to crack can be slow due to dependence on network speeds
- Small dictionary attacks are most common

Hydra Syntax and Options

- General format of command
  - `hydra -l user -p pass [other options] server service`

- `-l`, indicates a single username to test, where `user` is the username
- `-p`, indicates a single password to test, where `pass` is the password
- `server`, the IP address of the targeted machine
- `service`, the service to attack (Ex. svn, ssh, xmpp, etc.)
- Ex. `hydra -l admin -p password 246.252.2.220 telnet`

Hydra Syntax and Options

- Dictionary attacks
  - `hydra -L user_file -P pass_file [other options] server service`

- `-L`, indicates the name of the file containing usernames to test
- `-P`, indicates the name of the file containing passwords to test
- File should have one username or password per line
- Ex. `hydra -L users.txt -P passwords.txt 246.252.2.220 smtp`

Hydra Syntax and Options

- Brute force attack option
  - `-x min:max:charset`

- `min`, the minimum length of password to test
- `max`, the maximum length of password to test
- `charset`, the types of characters to test
  - `a`, indicates all lowercase letters from a to z
  - `A`, indicates all uppercase letters from A to Z
  - `1`, indicates numeric characters from 0 to 9
  - Special characters must be input individually to be checked. Ex. `!`, `?`, `$`, etc.
- Ex. `hydra -l abc -x 1:5:aA1#@ 150.69.211.33 ssh`

Hydra Options of Interest

- `-t n`, run attacks using `n` number of threads
- `-C file`, replaces `-L` and `-P` by having entries in `file` with a colon separated format: “username:password”
- `-M file`, attack servers indicated inside `file` in parallel
- `-o file`, output to `file` user-password pairs that are found to be valid
- Check `man hydra` for more

Hydra Web Form Attacks

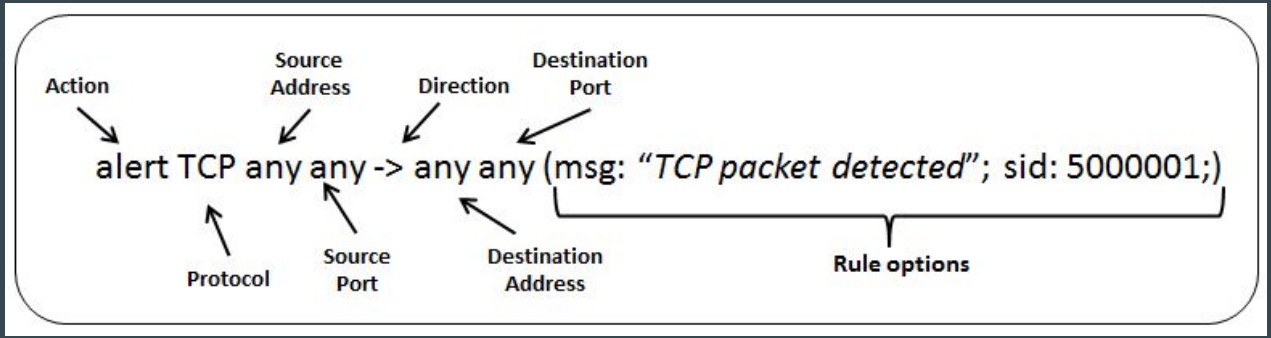
- Use the HTTP-POST-FORM protocol
  - `hydra -l user -p pass server http-form-post "server_page:POST_variables:fail_response"`

- `server_page` is the page on the server to POST to
- `POST_variables` are the variables to test (dependent on website)
  - Use `^USER^` and `^PASS^` in this section to indicate where to input usernames and passwords that are being attempted
- `fail_response` is a string that the website responds with to signal a failed login
- Ex. `hydra -l u -p p 192.168.0.29 http-form-post "/login.php:username=^USER^&password=^PASS^:Invalid Username"`

Demo Time

Ip Logging tools

- Snort
  - Free and Open Source Network Intrusion Prevention System (IPS) and Intrusion Detection System (IDS)
  - Rule-based language combining signature, protocol and anomaly inspection methods
  - Can detect malicious activity such as DOS attacks, Buffer overflows, stealth port scans and more
  - Capable of performing real time traffic analysis and packet logging on IP networks



Difference Between Snort & Wireshark

- Wireshark
  - Reads packets and decodes them in human readable text
  - Then you inspect what's in the packet
- Snort
  - As mentioned It is IDS and IPS
  - Scans for malicious activities in packets & ALERTS user

Sources

- https://github.com/vanhauser-thc/thc-hydra/blob/master/web/README
- https://tools.kali.org/password-attacks/hydra
- https://www.automatetheplanet.com/thc-hydra-password-cracking-by-examples/
- http://rohanhande.blogspot.com/2012/03/password-cracking-part-4-online-vs.html
- https://security.stackexchange.com/questions/37020/why-does-hydra-return-16-valid-passwords-when-none-are-valid
- http://insidetrust.blogspot.com/2011/08/using-hydra-to-dictionary-attack-web.html