

## Steps to setup and use sherlock OSINT tool

1. Setup Kali Linux virtual machine on Virtualbox / VMWare and login into the box. Open terminal inside virtual machine and switch to root user with below command.

**sudo su - root**

```
(kali@kali)-[~]
$ sudo su - root
[sudo] password for kali:
(root@kali)-[~]
```

2. Now update the OS with below command to make sure all libraries and modules are up to date.

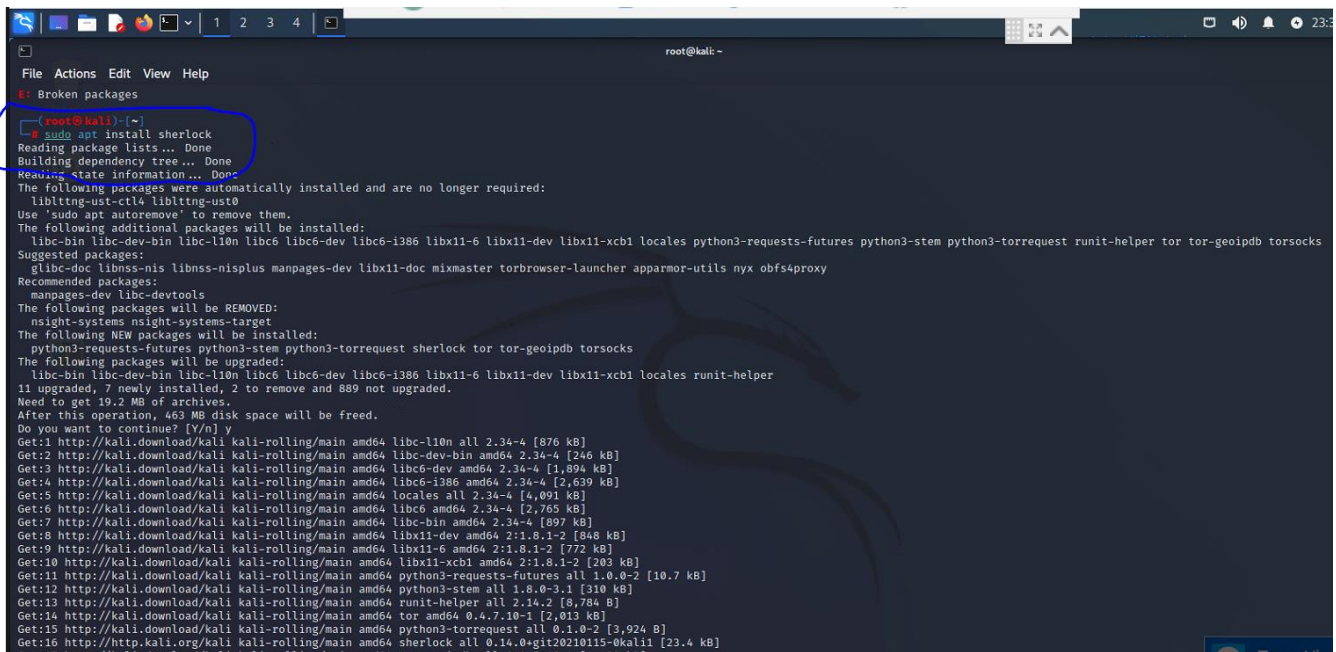
**sudo apt install update**

```
(root@kali)-[~]
# sudo apt update
Get:1 http://kali.download/kali kali-rolling InRelease [30.6 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [18.4 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [4
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [110
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb)
Get:6 http://kali.download/kali kali-rolling/non-free amd64 Packages [221
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb)
```

3. Now install sherlock OSINT tool with below command.

Installation documentation is provided at this link <https://www.kali.org/tools/sherlock/>

**sudo apt install sherlock**



```
(root@kali)-[~]
# sudo apt install sherlock
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libltdl-dev libltdl4 libltdl5
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libc-bin libc-dev-bin libc-l10n libc6 libc6-dev libc6-i386 libx11-6 libx11-dev libx11-xcb1 locales python3-requests-futures python3-stem python3-torrequest runit-helper tor tor-geoipdb torsocks
Suggested packages:
  glibc-doc libnss-nis libnss-nisplus manpages-dev libx11-doc mixmaster torbrowser-launcher apparmor-utils nxy obfs4proxy
Recommended packages:
  manpages-dev libc-devtools
The following packages will be REMOVED:
  nsight-systems nsight-systems-target
The following NEW packages will be installed:
  python3-requests-futures python3-stem python3-torrequest sherlock tor tor-geoipdb torsocks
The following packages will be upgraded:
  libc-bin libc-dev-bin libc-l10n libc6 libc6-dev libc6-i386 libx11-6 libx11-dev libx11-xcb1 locales runit-helper
11 upgraded, 7 newly installed, 2 to remove and 889 not upgraded.
Need to get 19.2 MB of archives.
After this operation, 463 MB disk space will be freed.
Do you want to continue? [Y/n] y
Get:1 http://kali.download/kali kali-rolling/main amd64 libc-l10n all 2.34-4 [876 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 libc-dev-bin amd64 2.34-4 [246 kB]
Get:3 http://kali.download/kali kali-rolling/main amd64 libc6-dev amd64 2.34-4 [1,894 kB]
Get:4 http://kali.download/kali kali-rolling/main amd64 libc6-i386 amd64 2.34-4 [2,639 kB]
Get:5 http://kali.download/kali kali-rolling/main amd64 locales all 2.34-4 [4,091 kB]
Get:6 http://kali.download/kali kali-rolling/main amd64 libc6 amd64 2.34-4 [2,765 kB]
Get:7 http://kali.download/kali kali-rolling/main amd64 libc-bin amd64 2.34-4 [897 kB]
Get:8 http://kali.download/kali kali-rolling/main amd64 libx11-dev amd64 2:1.8.1-2 [848 kB]
Get:9 http://kali.download/kali kali-rolling/main amd64 libx11-6 amd64 2:1.8.1-2 [772 kB]
Get:10 http://kali.download/kali kali-rolling/main amd64 libx11-xcb1 amd64 2:1.8.1-2 [203 kB]
Get:11 http://kali.download/kali kali-rolling/main amd64 python3-requests-futures all 1.0.0-2 [10.7 kB]
Get:12 http://kali.download/kali kali-rolling/main amd64 python3-stem all 1.8.0-3.1 [310 kB]
Get:13 http://kali.download/kali kali-rolling/main amd64 python3-torrequest all 2.34-2 [8,784 B]
Get:14 http://kali.download/kali kali-rolling/main amd64 tor amd64 0.4.7.10-1 [2,013 kB]
Get:15 http://kali.download/kali kali-rolling/main amd64 python3-requests all 0.1.0-2 [3,924 B]
Get:16 http://kali.download/kali kali-rolling/main amd64 sherlock all 0.14.0+git20210115-0kali1 [23.4 kB]
```

4. Give below command to get all instructions on how to use the tool for reference.

**sherlock -h**

```

(root@kali)-[~]
# sherlock -h
usage: sherlock [-h] [--version] [--verbose] [--folderoutput FOLDEROUTPUT] [--output OUTPUT] [--tor] [--unique-tor] [--csv] [--site SITE_NAME] [--proxy PROXY_URL] [--json]
               [--print-all] [--print-found] [--no-color] [--browse] [--local]
               USERNAMES [USERNAMES ...]

Sherlock: Find Usernames Across Social Networks (Version 0.14.0)

positional arguments:
  USERNAMES              One or more usernames to check with social networks.

options:
  -h, --help              show this help message and exit
  --version               Display version information and dependencies.
  --verbose, -v, -d, --debug
                        Display extra debugging information and metrics.
  --folderoutput FOLDEROUTPUT, -fo FOLDEROUTPUT
                        If using multiple usernames, the output of the results will be saved to this folder.
  --output OUTPUT, -o OUTPUT
                        If using single username, the output of the result will be saved to this file.
  --tor, -t               Make requests over Tor; increases runtime; requires Tor to be installed and in system path.
  --unique-tor, -u        Make requests over Tor with new Tor circuit after each request; increases runtime; requires Tor to be installed and in system path.
  --csv                   Create Comma-Separated Values (CSV) File.
  --site SITE_NAME        Limit analysis to just the listed sites. Add multiple options to specify more than one site.
  --proxy PROXY_URL, -p PROXY_URL
                        Make requests over a proxy. e.g. socks5://127.0.0.1:1080
  --json JSON_FILE, -j JSON_FILE
                        Load data from a JSON file or an online, valid, JSON file.
  --timeout TIMEOUT       Time (in seconds) to wait for response to requests. Default timeout is infinity. A longer timeout will be more likely to get results from slow sites.
  --print-all             Output sites where the username was not found.
  --print-found            Output sites where the username was found.
  --no-color              Don't color terminal output
  --browse, -b            Browse to all results on default browser.
  --local, -l             Force the use of the local data.json file.

```

- For example, I have used below keywords to search about [REDACTED] on sherlock tool for reference and obtained below results.

**sherlock <keyword>**

(Keyword can be person name or website domain name etc)

```

(root@kali)-[~]
# sherlock [REDACTED]
[*] Checking username [REDACTED] on:
[+] Academia.edu: [REDACTED]
[+] Coil: [REDACTED]
[+] Facebook: [REDACTED]
[+] Fiverr: [REDACTED]
[+] Instagram: [REDACTED]
[+] Kaggle: [REDACTED]
[+] [REDACTED]
[+] NICommunityForum: [REDACTED]
[+] SlideShare: [REDACTED]
[+] [REDACTED]
[+] Twitter: [REDACTED]
[+] [REDACTED]

(root@kali)-[~]
# sherlock [REDACTED]
[*] Checking username [REDACTED] on:
[+] Coil: [REDACTED]
[+] Fiverr: [REDACTED]
[+] Instagram: [REDACTED]
[+] Minecraft: [REDACTED]
[+] NICommunityForum: [REDACTED]
[+] [REDACTED]
[+] [REDACTED]

(root@kali)-[~]
# sherlock [REDACTED]
[*] Checking username [REDACTED] on:
[+] Coil: [REDACTED]
[+] Fiverr: [REDACTED]
[+] Minecraft: [REDACTED]
[+] NICommunityForum: [REDACTED]
[+] [REDACTED]
[+] [REDACTED]

```

6. From a hacker's point of view, we can search all these links and group all of them that belonging to one person. By using this information, we can track specific person pics, location and other sensitive info.