

Skill Horizon Task2

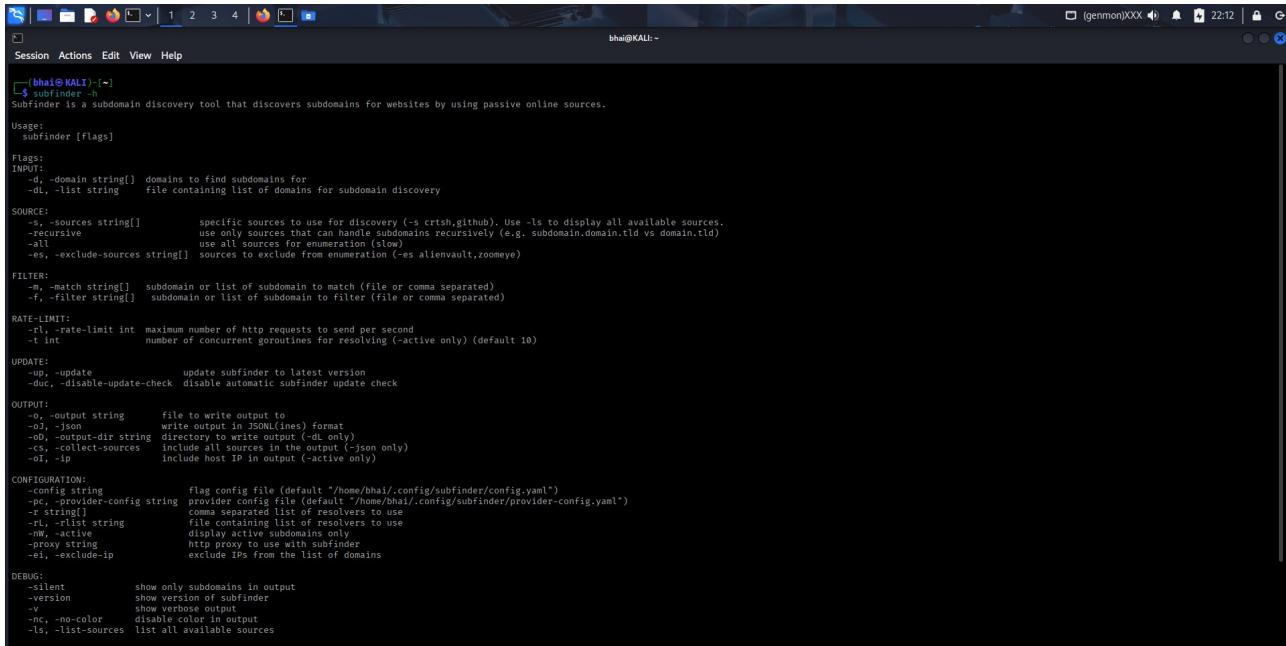
(Day 15-Nov-2025)

by:- Vikram Budania

Tasks

Install these scripts in the Kali Linux (go based)

• Subfinder



```
(bhai㉿KALI)-[~]
$ subfinder --help
Subfinder is a subdomain discovery tool that discovers subdomains for websites by using passive online sources.

Usage:
subfinder [Flags]

Flags:
INPUT:
-d, --domain string[] domains to find subdomains for
--dl, --list string file containing list of domains for subdomain discovery

SOURCE:
-s, --sources string[] specific sources to use for discovery (-s crtsh,github). Use -ls to display all available sources.
--source-only sources that can handle subdomains recursively (e.g. subdomain.domain.tld vs domain.tld)
-all use all sources for enumeration (slow)
--es, --exclude-sources string[] sources to exclude from enumeration (-es alienVault,zoomeye)

FILTER:
-m, --match string[] subdomain or list of subdomain to match (file or comma separated)
-f, --filter string[] subdomain or list of subdomain to filter (file or comma separated)

RATE-LIMIT:
-rL, --rate-limit int maximum number of http requests to send per second
-l int number of concurrent goroutines for resolving (-active only) (default 10)

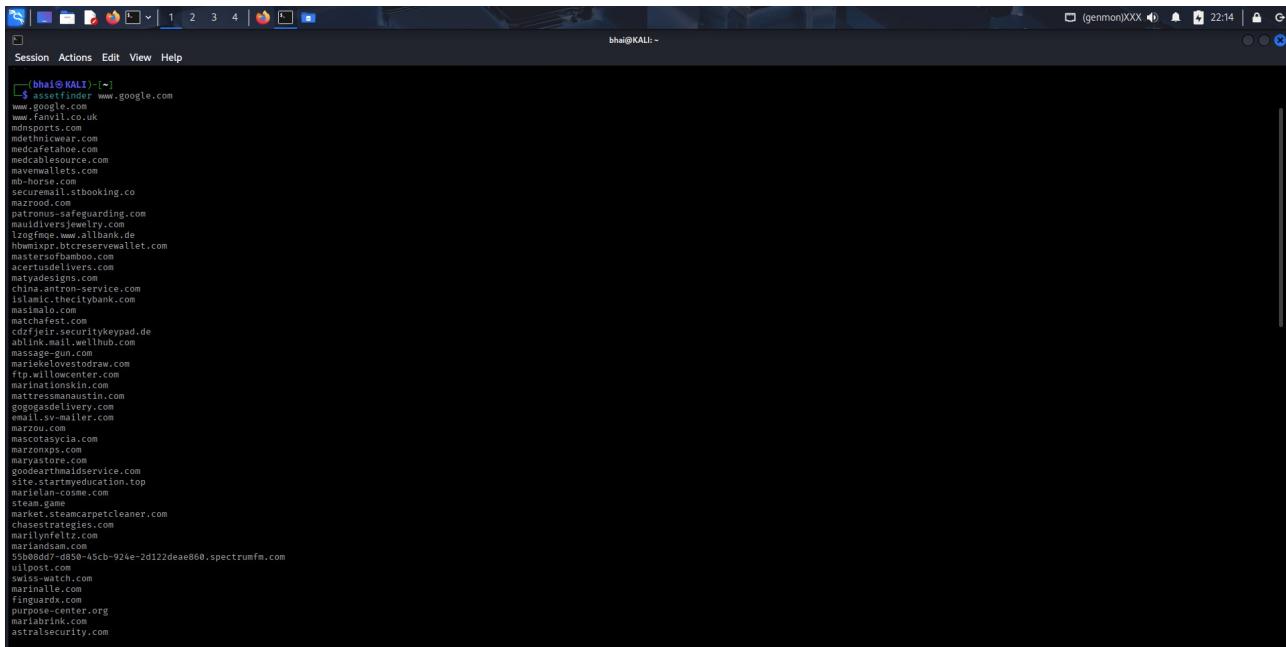
UPDATE:
-u, --update update subfinder to latest version
--duc, --disable-update-check disable automatic subfinder update check

OUTPUT:
-o, --output string file to write output to
-oJ, --json write output in JSONLINES format
-oD, --output-dir string directory to write output (-dl only)
-ss, --collect-sources include all sources in the output (-json only)
-oI, --ip include host IP in output (-active only)

CONFIGURATION:
--config string flag config file (default "/home/bhai/.config/subfinder/config.yaml")
--pc, --provider-config string provider config file (default "/home/bhai/.config/subfinder/provider-config.yaml")
--sr string[] comma separated list of resolvers to use
--rl, --list string file containing list of resolvers to use
--NW, --active display active subdomains only
--proxy string http proxy to use with subfinder
--ei, --exclude-ip exclude IPs from the list of domains

DEBUG:
-silent show only subdomains in output
--version Show version of subfinder
-v show verbose output
--nc, --no-color disable color in output
--ls, --list-sources list all available sources
```

• Assetfinder



```
(bhai㉿KALI)-[~]
$ assetfinder www.google.com
www.google.com
www.Fanvi1.co.uk
mdnsports.com
mdnsports.com
medcablestv.com
medcablestv.com
mavenwallets.com
mb-horse.com
secureoutbooking.co
mazrood.com
patronus-safeguarding.com
mauidiversjewelry.com
taglogistics.de
humixpr.brcrservewallet.com
mastersofbamboo.com
acertusdelivers.com
matyadesigns.com
elitelyhostingservice.com
islamic.thecitybank.com
masimalo.com
matchafest.com
clicksecuritykeypad.de
atlink.mail.wellhub.com
massage-gun.com
mariekelovesraw.com
firstrulecenter.com
marinationsskin.com
matressmanastin.com
gogogadelivery.com
email-mailer.com
marzou.com
mascotasyria.com
marzonxps.com
maryastore.com
goodwillservice.com
site.startmyeducation.top
marieilan-cosme.com
Steam.gq
markstamcarpetcleaner.com
chasestrategies.com
marilynfeltz.com
marilandsam.com
S3000000000-45cb-924e-2d122deae860.spectrumfm.com
uiiost.com
swiss-watch.com
marinale.com
finguard.com
purple-crown.org
mariahrink.com
astralsecurity.com
```

• Alterx

```
(vikram@VIKRAM:~)$ alterx -h
Fast and customizable subdomain wordlist generator using DSL.

Usage:
  alterx [flags]

Flags:
INPUT:
  -l, -list string[]    subdomains to use when creating permutations (stdin, comma-separated, file)
  -p, -pattern string[] custom permutation patterns input to generate (comma-separated, file)
  -pp, -payload value   custom payload pattern input to replace/use in key=value format (-pp 'word=words.txt')

OUTPUT:
  -es, -estimate         estimate permutation count without generating payloads
  -o, -output string    output file to write altered subdomain list
  -ms, -max-size value  Max export data size (kb, mb, gb, tb) (default mb)
  -v, -verbose           display verbose output
  -silent                display results only
  -version               display alterx version

CONFIG:
  -config string          alterx cli config file (default '$HOME/.config/alterx/config.yaml')
  -en, -enrich             enrich wordlist by extracting words from input
  -ac string               alterx permutation config file (default '$HOME/.config/alterx/permutation_v0.0.6.yaml')
  -limit int                limit the number of results to return (default 0)

UPDATE:
  -up, -update              update alterx to latest version
  -duc, -disable-update-check  disable automatic alterx update check

(vikram@VIKRAM:~)$ []
```

- **Httpx**

```
bhai@KALI:~$ httpx -h
Error: Option '-h' requires 2 arguments.

[bhai@KALI]:~$ httpx --help
HTTPX v
A next generation HTTP client.

Usage: httpx <URL> [OPTIONS]

-m, --method METHOD      Request method, such as GET, POST, PUT, PATCH, DELETE, OPTIONS, HEAD.
                          [Default: GET, or POST if a request body is included]
-p, --params <NAME VALUE> ...
                          Query parameters to include in the request URL.
-c, --content TEXT       Content type to include in the request body.
-d, --data <NAME VALUE> ...
                          Form data to include in the request body.
-f, --files <NAME FILENAME> ...
                          Form files to include in the request body.
-j, --json TEXT          JSON data to include in the request body.
-h, --headers <NAME VALUE> ...
                          Include additional HTTP headers in the request.
--cookies <NAME VALUE> ...
                          Cookies to include in the request.
--auth <USER PASS>       Username and password to include in the request. Specify '--' for the password to use a password prompt. Note that using --verbose/-v will expose the Authorization header, including the password encoding in a trivially reversible format.
--proxy URL              Send the request via a proxy. Should be the URL giving the proxy address.
--timeout FLOAT           Timeout value to use for network operations, such as establishing the connection, reading some data, etc ... [Default: 5.0]
--follow-redirects        Automatically follow redirects.
--no-verify               Disable SSL verification.
--http2                  Send the request using HTTP/2, if the remote server supports it.
--download FILE           Save the response content as a file, rather than displaying it.
-v, --verbose             Verbose output. Show request as well as response.
--help                   Show this message and exit.

[bhai@KALI]:~$
```

- Katana

```
(vikram@VIKRAM:~)$ katana
/ \____ / \____ / \____ / \____ / \____
/ \ \ \ \ \ , / \ \ \ \ \ , / \ \ \ \ \ , / \ \ \ \ \
projectdiscovery.io

[INF] Current katana version v1.2.1 (outdated)
[FTL] could not create runner: [:RUNTIME] could not validate options <- [:RUNTIME] no inputs specified for crawler
(vikram@VIKRAM:~)$ 
```

• Gau

```
(vikram@VIKRAM:~)$ gau --help
Usage of gau:
--blacklist strings    list of extensions to skip
--config string        location of config file (default $HOME/.gau.toml or %USERPROFILE%\gau.toml)
--fc strings           list of status codes to filter
--fp                  remove different parameters of the same endpoint
--from string          fetch urls from date (format: YYYYMM)
--ft strings           list of mime-types to filter
--json                output as json
--mc strings           list of status codes to match
--mt strings           list of mime-types to match
--o string             filename to write results to
--providers strings   list of providers to use (wayback,commoncrawl,otx,urlscan)
--proxy string         http proxy to use
--retries uint         retries for HTTP client
--subs                include subdomains of target domain
--threads uint         number of workers to spawn (default 1)
--timeout uint         timeout (in seconds) for HTTP client (default 45)
--to string            fetch urls to date (format: YYYYMM)
--verbose              show verbose output
--version              show gau version
(vikram@VIKRAM:~)$ 
```

• Fuzzuli

```
Mon Nov 17 10:25:40 PM 0%
0% 21% ↑ 0.6KB ↓ 53KB Terminal 100%
(vikram@VIKRAM:~)$ fuzzuli -h
Fuzzuli is a url fuzzing tool that aims to find critical backup files by creating a dynamic wordlist based on the domain.

Usage:
  fuzzuli [flags]

Flags:
GENERAL OPTIONS:
  -w int    worker count (default 16)
  -f string  input file containing list of host/domain
  -pt string  paths. separate with commas to use multiple paths. e.g. ./db/,old/ (default "/")
  -p          print urls that is sent request
  -sl          silent mode
  -v          print version

WORDLIST OPTIONS:
  -nt string  methods. available methods: regular, withoutdots, withoutvowels, reverse, mixed, withoutdv, shuffle
  -sf string  suffix
  -pf string  prefix
  -ex string  file extension. default (rar, zip, tar.gz, tar, gz, jar, 7z, bz2, sql, backup, war)
  -rp string  replace specified char
  -rm string  remove specified char
  -jw          just generate wordlist do not http request

DOMAIN OPTIONS:
  -es string  exclude domain that contains specified string or char. e.g. for OR operand google|bing@yahoo (default "#")
  -dl int     match domain length that specified. (default 40)

MATCHER OPTIONS:
  -ct string  match response with specified content type
  -sc int     match response with specified status code (default 200)
  -cl int     match response with specified minimum content length. e.g. >100 (default 100)

HTTP OPTIONS:
  -to int     timeout in seconds. (default 10)
  -ua string  user agent (default "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:100.0) Gecko/20100101 Firefox/100.0")
  -hm string  HTTP Method. (default "HEAD")
  -px string  http proxy to use

(vikram@VIKRAM:~)$ 
```

• Waybackurls

```
(vikram@VIKRAM:~)$ waybackurls -h
Usage of waybackurls:
  -dates
    show date of fetch in the first column
  -get-versions
    list URLs for crawled versions of input URL(s)
  -no-subs
    don't include subdomains of the target domain
(vikram@VIKRAM:~)$ █
```