



Red Team Recon - Tryhackme

Reconnaissance

- It can be defined as a preliminary survey or observation of our target (client) without alerting them to our activities.
- Reconnaissance can be broken down into two parts — Passive Reconnaissance and Active Reconnaissance.

Taxonomy of Reconnaissance

- **Passive Recon:**
 - Can be carried out by watching passively.
 - Doesn't require interacting with the target i.e. we aren't sending any packets or requests to the target.
 - It relies on publicly available information that is collected and maintained by a third party.
 - Open Source Intelligence (OSINT) is used to collect information about the target.
 - For example - We might collect Domain names, IP address blocks, email addresses, employee names, and job posts.
- **Active Recon:**
 - Requires interacting with the target by sending requests and packets and observing how it responds.
 - An example of Active recon- Using Nmap to scan target subnets and live hosts.
 - Information that we would want to discover - Live hosts, running servers, listening services and version numbers.
 - It can be classified as:

- **External Recon:** Conducted outside the target network. Example - Running Nikto from outside the target company network.
- **Internal Recon:** Conducted from within the target company's network. Example - Using Nessus to scan the internal network using one of the target's computers.

Built-In Tools

WHOIS

- It is a request and response protocol that follows the [RFC 3912](#) specification.
- A WHOIS server listens on TCP port 43 for incoming requests.
- The domain registrar is responsible for maintaining the WHOIS records for the domain names.
- `whois` queries the WHOIS server to provide all saved records.
- `whois` provides us with:
 - Registrar WHOIS server
 - Registrar URL
 - Record creation date
 - Record update date
 - Registrant contact info and address (unless withheld for privacy)
 - Admin contact info and address (unless withheld for privacy)
 - Tech contact info and address (unless withheld for privacy)
 - At the end of the `whois` query, we find the authoritative name servers.

```
(anishroy@linuxmint) - [~]
$ whois thmredteam.com
Domain Name: THMREDTEAM.COM
Registry Domain ID: 2643258257_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.namecheap.com
Registrar URL: http://www.namecheap.com
Updated Date: 2022-09-26T15:22:32Z
Creation Date: 2021-09-24T14:04:16Z
Registry Expiry Date: 2023-09-24T14:04:16Z
Registrar: NameCheap, Inc.
Registrar IANA ID: 1068
Registrar Abuse Contact Email: abuse@namecheap.com
Registrar Abuse Contact Phone: +1.6613102107
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Name Server: KIP.NS.CLOUDFLARE.COM
Name Server: UMA.NS.CLOUDFLARE.COM
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2023-04-11T05:54:05Z <<<

For more information on Whois status codes, please visit https://icann.org/epp

NOTICE: The expiration date displayed in this record is the date the
registrar's sponsorship of the domain name registration in the registry is
currently set to expire. This date does not necessarily reflect the expiration
date of the domain name registrant's agreement with the sponsoring
registrar. Users may consult the sponsoring registrar's Whois database to
view the registrar's reported date of expiration for this registration.

TERMS OF USE: You are not authorized to access or query our Whois
database through the use of electronic processes that are high-volume and
automated except as reasonably necessary to register domain names or
modify existing registrations; the Data in VeriSign Global Registry
Services' ("VeriSign") Whois database is provided by VeriSign for
information purposes only, and to assist persons in obtaining information
about or related to a domain name registration record. VeriSign does not
guarantee its accuracy. By submitting a Whois query, you agree to abide
by the following terms of use: You agree that you may use this Data only
for lawful purposes and that under no circumstances will you use this Data
```

nslookup

- It uses the default DNS server to get the A and AAAA records related to domain.

```
(anishroy@linuxmint) - [~]  
$ nslookup clinic.thmredteam.com  
Server:      127.0.0.53  
Address:     127.0.0.53#53  
  
Non-authoritative answer:  
Name:   clinic.thmredteam.com  
Address: 172.67.212.249  
Name:   clinic.thmredteam.com  
Address: 104.21.93.169  
Name:   clinic.thmredteam.com  
Address: 2606:4700:3034::6815:5da9  
Name:   clinic.thmredteam.com  
Address: 2606:4700:3034::ac43:d4f9
```

dig (Domain Information Groper)

- It provides a lot of query options and even allows us to specify a different DNS server to use.
- For example - We can use Cloudflare DNS server:

```
$ dig @1.1.1.1 tryhackme.com
```

```
(anishroy@linuxmint) - [~]
$ dig cafe.thmredteam.com @1.1.1.1

; <<>> DiG 9.18.12-0ubuntu0.22.04.1-Ubuntu <<>> cafe.thmredteam.com @1.1.1.1
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29952
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;cafe.thmredteam.com.      IN      A

;; ANSWER SECTION:
cafe.thmredteam.com.      300     IN      A      172.67.212.249
cafe.thmredteam.com.      300     IN      A      104.21.93.169

;; Query time: 172 msec
;; SERVER: 1.1.1.1#53(1.1.1.1) (UDP)
;; WHEN: Tue Apr 11 15:21:05 IST 2023
;; MSG SIZE rcvd: 80
```

host

- It is another useful alternative for querying DNS servers for DNS records.
- Example-

```
(anishroy@linuxmint) - [~]
$ host cafe.thmredteam.com
cafe.thmredteam.com has address 104.21.93.169
cafe.thmredteam.com has address 172.67.212.249
cafe.thmredteam.com has IPv6 address 2606:4700:3034::ac43:d4f9
cafe.thmredteam.com has IPv6 address 2606:4700:3034::6815:5da9
```

traceroute

- It traces the route taken by the packets from our system to the target host.
- Some routers don't respond to the packets sent by `traceroute`, as a result, we don't see their IP addresses.

```
(anishroy@linuxmint) - [~]
$ traceroute cafe.thmredteam.com
traceroute to cafe.thmredteam.com (103.224.182.242), 30 hops max, 60 byte packets
 1 * * *
 2 192.168.60.9 (192.168.60.9) 1.042 ms 1.029 ms 1.018 ms
 3 117.254.129.130 (117.254.129.130) 1.005 ms 0.994 ms 0.983 ms
 4 172.24.223.54 (172.24.223.54) 2.009 ms 1.998 ms 1.986 ms
 5 * * *
 6 * * *
 7 182.73.147.245 (182.73.147.245) 6.383 ms 7.427 ms 6.360 ms
 8 182.79.243.34 (182.79.243.34) 258.381 ms 182.79.247.94 (182.79.247.94) 248.988 ms 116.119.112.98 (116.119.112.98) 234.871 ms
 9 * * *
10 be3271.ccr41.lax01.atlas.cogentco.com (154.54.42.101) 282.178 ms be3360.ccr42.lax01.atlas.cogentco.com (154.54.25.149) 284.083 ms 300.324 ms
11 be2276.rcr52.san01.atlas.cogentco.com (154.54.83.162) 280.229 ms be2275.rcr51.san01.atlas.cogentco.com (154.54.6.218) 277.128 ms
12 te0-0-2-3.nr11.b022887-0.san01.atlas.cogentco.com (154.24.7.230) 281.562 ms te0-0-2-0.nr11.b022887-0.san01.atlas.cogentco.com (154.24.7.226) 283.646 ms te0-0-2-3.nr11.b022887-0.san01.a
tlas.cogentco.com (154.24.7.230) 284.714 ms
13 38.140.111.50 (38.140.111.50) 278.213 ms 277.062 ms 277.049 ms
14 sw01-te02-san-trellian.com (103.224.213.214) 280.414 ms 298.499 ms 285.256 ms
15 lb-182-242.above.com (103.224.182.242) 279.861 ms 276.808 ms 282.630 ms
```

Advanced Searching

- Some popular search modifiers that work with many popular search engines.

Symbol / Syntax	Function
<code>"search phrase"</code>	Find results with exact search phrase
<code>OSINT filetype:pdf</code>	Find files of type <code>PDF</code> related to a certain term.
<code>salary site:blog.tryhackme.com</code>	Limit search results to a specific site.
<code>pentest -site:example.com</code>	Exclude a specific site from results
<code>walkthrough intitle:TryHackMe</code>	Find pages with a specific term in the page title.
<code>challenge inurl:tryhackme</code>	Find pages with a specific term in the page URL.

- Google provides a web interface for advanced searches: [Google Advanced Search](#).
- Other examples - [Google Refine Web Searches](#), [DuckDuckGo Search Syntax](#), [Bing Advanced Search Options](#).
- Search engines crawl the world wide web day and night which sometimes leads to indexing confidential information like:
 - Documents for internal company use
 - Confidential spreadsheets with usernames, email addresses and even passwords.
 - Files containing usernames.
 - Sensitive directories
 - Service version number
 - Error messages
- Websites such as [Google Hacking Database](#) (GHDB) collect Google advanced searches with specific search terms and are publicly available.
- GHDB contains queries under the following categories:

- **Footholds:** Consider GHDB-ID: 6364 as it uses the query `intitle:"index of" "nginx.log"` to discover Nginx logs and might reveal server misconfigurations that can be exploited.
- **File Containing Usernames:** Example- GHDB-ID: 7047 uses the search term `intitle:"index of" "contacts.txt"` to discover files that leak juicy information.
- **Sensitive Directories:** consider GHDB-ID: 6768, which uses the search term `inurl:/certs/server.key` to find out if a private RSA key is exposed.
- **Web Server Detection:** Consider GHDB-ID: 6876, which detects GlassFish server information using the query `intitle:"GlassFish Server - Server Running"`.
- **Vulnerable Files:** Example- We can try to locate PHP files using the query `intitle:"index of" "*.php"` as provided by GHDB-ID: 7786.
- **Vulnerable Servers:** For example- To discover Solarwinds Orion web consoles GHDB-ID: 6728 uses the query `intext:"user name" intext:"orion core" -solarwinds.com`.
- **Error Messages:** Useful information can be extracted from error messages. Example - GHDB-ID: 5963, uses the query `intitle:"index of" errors.log` to find log files related to errors.
- Additional sources that can provide us valuable information without interacting with our target:
 - Social Media
 - Job ads

Specialized Search Engines

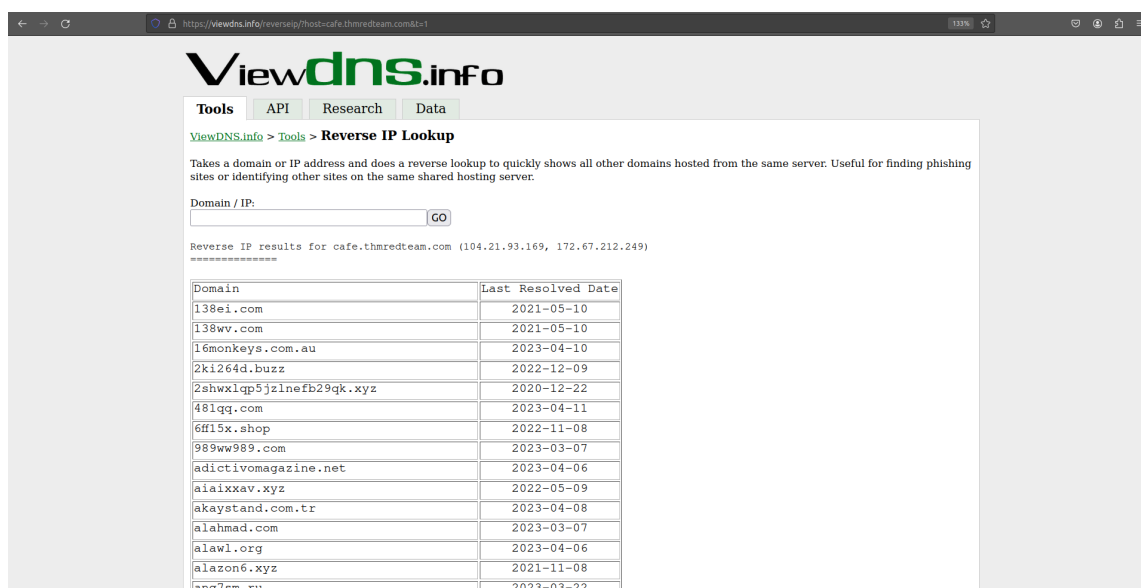
WHOIS and DNS Related

- There are third parties that offer paid services for historical WHOIS data for example - WHOIS history.
- It provides a history of WHOIS data and can come handy if domain registrant didn't use WHOIS privacy when they registered the domain.

- Some websites that offer advanced DNS services are:
 - [ViewDNS.info](https://viewdns.info)
 - [Threat Intelligence Platform](#)

ViewDNS.info

- It offers Reverse IP lookup.
- Today, it is common to come across shared hosting servers.
- In shared hosting, one IP address is shared among many different web servers with different domain names.
- With reverse IP lookup, we can find other domain names using a specific IP address (es).
- For example- Using reverse IP lookup to find other servers sharing the same IP address used by `cafe.thmredteam.com`



- So, IP address does not necessarily lead to a single website.

Threat Intelligence Platform

- It requires us to provide a domain name or an IP address.
- It launches a series of tests from malware checks to WHOIS and DNS queries.
- The WHOIS and DNS results are similar to the results we get using `whois` and `dig`

- Threat Intelligence Platform presents them in a more readable and visually appealing way.
- For example- On looking for thmredteam.com we also get the Name Server records resolved to their IPv4 and IPv6 addresses.

Malware	WHOIS	MX	NS												
<div> <div>Primary name server</div> <div>kip.ns.cloudflare.com</div> </div> <div> <div>Hostmaster (e-mail)</div> <div>dns.cloudflare.com</div> </div> <div> <div>Serial</div> <div>2298698524</div> </div> <div> <div>Refresh</div> <div>10000</div> </div> <div> <div>Retry</div> <div>2400</div> </div> <div> <div>Expire</div> <div>604800</div> </div> <div> <div>Minimum TTL</div> <div>3600</div> </div>															
<div>NS records</div> <div>NS records successfully fetched from the parent name server: d.gtld-servers.net.</div> <table> <tr> <th>NS server</th><th>IPv4</th><th>IPv6</th><th>TTL</th></tr> <tr> <td>kip.ns.cloudflare.com</td><td>108.162.193.128</td><td>2606:4700:58:adf5:3b80</td><td>172800</td></tr> <tr> <td>uma.ns.cloudflare.com</td><td>108.162.192.146</td><td>2803:f800:50::6ca2:c092</td><td>172800</td></tr> </table>				NS server	IPv4	IPv6	TTL	kip.ns.cloudflare.com	108.162.193.128	2606:4700:58:adf5:3b80	172800	uma.ns.cloudflare.com	108.162.192.146	2803:f800:50::6ca2:c092	172800
NS server	IPv4	IPv6	TTL												
kip.ns.cloudflare.com	108.162.193.128	2606:4700:58:adf5:3b80	172800												
uma.ns.cloudflare.com	108.162.192.146	2803:f800:50::6ca2:c092	172800												

- On looking for cafe.thmredteam.com we got a list of domain names on the same IP address as we got with ViewDNS.info.

IPs	WEB	SSL	Malware	WHOIS	MX	NS
<div>Other domains on the same IP</div> <div>16monkeys.com.au</div> <div>1xbet-dom1.top</div> <div>225i.shop</div> <div>246702.org</div> <div>24hourplumbercairns.com.au</div> <div>313star.com</div> <div>422231.com</div> <div>481qq.com</div> <div>4cyxj4.com</div> <div>4kf.cc</div>						

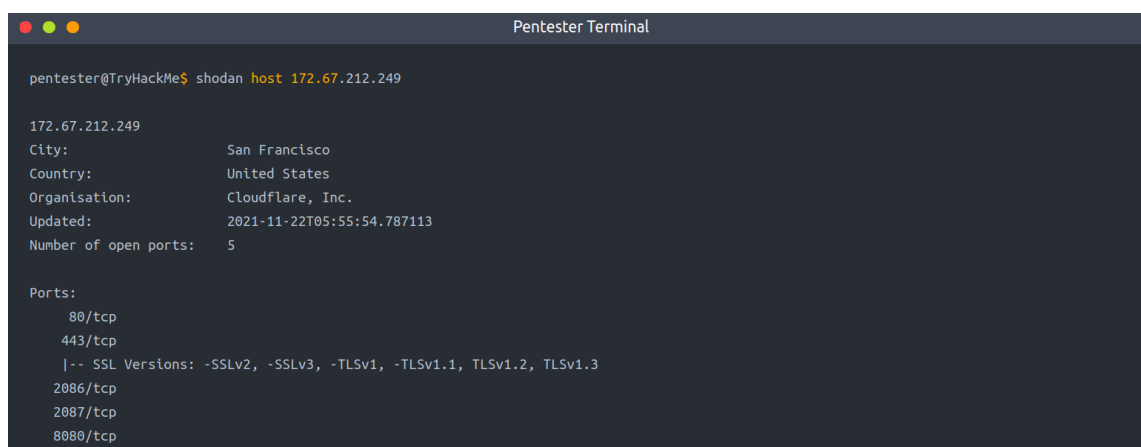
Specialized Search Engines

Censys

- Censys Search can provide a lot of information about IP addresses and domains.
- We can also get information related to ports when we lookup for a certain IP address.

Shodan

- It is a search engine, but we can also use it from command line.
- We need to create an account on Shodan, and then configure to use our API key using the command - `shodan init API_KEY`
- To learn more about shodan- <https://cli.shodan.io/>
- Example of looking up information about one of the IP addresses we got from `nslookup cafe.thmredteam.com`, using the command- `shodan host IP_ADDRESS`
- We can get the geographical location of the IP address and open ports as-



```

Pentester Terminal

pentester@TryHackMe$ shodan host 172.67.212.249

172.67.212.249
City:           San Francisco
Country:        United States
Organisation:   Cloudflare, Inc.
Updated:        2021-11-22T05:55:54.787113
Number of open ports: 5

Ports:
  80/tcp
  443/tcp
  |-- SSL Versions: -SSLv2, -SSLv3, -TLSv1, -TLSv1.1, TLSv1.2, TLSv1.3
  2086/tcp
  2087/tcp
  8080/tcp
  
```

Recon-ng

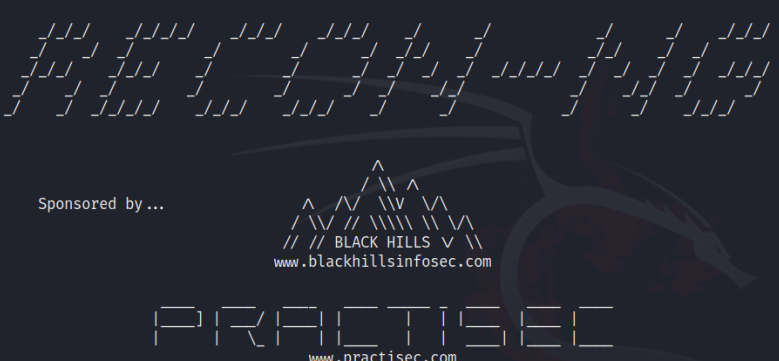
- It is a framework that helps automate the OSINT work.
- It uses modules from various authors and provides a multitude of functionality.
- Some modules require keys to work as the key allows the module to query the related online API.
- It can be used to find various bits and pieces of information that can aid in an operation or OSINT task.

- All the data collected is automatically saved in the database related to our workspace.
- For example- We might discover host addresses to later port-scan etc.
- We can start recon-ng using the command `recon-ng`
- Then we need to select the installed module we want to use, or install the module(s) we need.

Creating a Workspace

- After starting recon-ng, we run the command- `workspaces create thmredteam`.
- It creates a workspace of the name *thmredteam*.

```
(kali@kali)-[~]
$ recon-ng
[*] Version check disabled.
```



```
[recon-ng v5.1.2, Tim Tomes (@lanmaster53)]
[*] No modules enabled/installed.
[recon-ng][default] > workspaces create thmredteam
[recon-ng][thmredteam] > workspaces list
```

```
[recon-ng][thmredteam] > workspaces list
```

Workspaces	Modified
default	2023-04-12 14:29:15
thmredteam	2023-04-12 14:36:19

```
[recon-ng][thmredteam] > db schema
```

domains	
domain	TEXT
notes	TEXT
module	TEXT

companies	
company	TEXT
description	TEXT
notes	TEXT
module	TEXT

- We can start recon-ng with the specific workspace using the command- `recon-ng -w WORKSPACE_NAME`.

Seeding The Database

- Here, we know the target's domain name, so we will feed it into the recon-ng database related to the current workspace to gather more information about it.
- We can check the tables in our database, using the command - `db schema`.
- To insert thmredteam.com into the domains table, we can use the command- `db insert domains`.

```
[recon-ng][thmredteam] > db insert domains
domain (TEXT): thmredteam.com
notes (TEXT):
[*] 1 rows affected.
```

Recon-ng Marketplace

- Our next step would be to search for a module that transforms domains into other types of information.
- We will search for suitable modules from the marketplace.
- Some useful commands related to marketplace usage are:
 - `marketplace search KEYWORD` to search for available modules with *keyword*.
 - `marketplace info MODULE` to provide information about the module in question.
 - `marketplace install MODULE` to install the specified module into Recon-ng.
 - `marketplace remove MODULE` to uninstall the specified module.
- The modules are grouped under multiple categories, such as discovery, import, recon and reporting.
- To get a list of all available modules, we run - `marketplace search`.


```
[recon-ng][thmredteam] > marketplace search
```

Path	Version	Status	Updated	D	K
discovery/info_disclosure/cache_snoop	1.1	not installed	2020-10-13		
discovery/info_disclosure/interesting_files	1.2	not installed	2021-10-04		
exploitation/injection/command_injector	1.0	not installed	2019-06-24		
exploitation/injection/xpath_bruter	1.2	not installed	2019-10-08		
import/csv_file	1.1	not installed	2019-08-09		
import/list	1.1	not installed	2019-06-24		
import/masscan	1.0	not installed	2020-04-07		
import/nmap	1.1	not installed	2020-10-06		
recon/companies-contacts/bing_linkedin_cache	1.0	not installed	2019-06-24		*
recon/companies-contacts/censys_email_address	2.0	not installed	2021-05-11	*	*
recon/companies-contacts/pen	1.1	not installed	2019-10-15		
recon/companies-domains/censys_subdomains	2.0	not installed	2021-05-10	*	*
recon/companies-domains/pen	1.1	not installed	2019-10-15		
recon/companies-domains/viewdns_reverse_whois	1.1	not installed	2021-08-24		
recon/companies-domains/whoxy_dns	1.1	not installed	2020-06-17		*
recon/companies-hosts/censys_org	2.0	not installed	2021-05-11	*	*
recon/companies-hosts/censys_tls_subjects	2.0	not installed	2021-05-11	*	*

- After that, we search for modules *domains-*.

```
[recon-ng][thmredteam] > marketplace search domains-
[*] Searching module index for 'domains-'...
```

Path	Version	Status	Updated	D	K
recon/domains-companies/censys_companies	2.0	not installed	2021-05-10	*	*
recon/domains-companies/pen	1.1	not installed	2019-10-15		
recon/domains-companies/whoxy_whois	1.1	not installed	2020-06-24		*
recon/domains-contacts/hunter_io	1.3	not installed	2020-04-14		*
recon/domains-contacts/metacrawler	1.1	not installed	2019-06-24	*	
recon/domains-contacts/pen	1.1	not installed	2019-10-15		
recon/domains-contacts/pgp_search	1.4	not installed	2019-10-16		
recon/domains-contacts/whois_pocs	1.0	not installed	2019-06-24		
recon/domains-contacts/wikileaker	1.0	not installed	2020-04-08		
recon/domains-credentials/pwnedlist/account_creds	1.0	not installed	2019-06-24	*	*
recon/domains-credentials/pwnedlist/api_usage	1.0	not installed	2019-06-24		*
recon/domains-credentials/pwnedlist/domain_creds	1.0	not installed	2019-06-24	*	*
recon/domains-credentials/pwnedlist/domain_ispwned	1.0	not installed	2019-06-24		*
recon/domains-credentials/pwnedlist/leak_lookup	1.0	not installed	2019-06-24		
recon/domains-credentials/pwnedlist/leaks_dump	1.0	not installed	2019-06-24		*
recon/domains-domains/brute_suffix	1.1	not installed	2020-05-17		
recon/domains-hosts/binaryedge	1.2	not installed	2020-06-18		*
recon/domains-hosts/bing_domain_api	1.0	not installed	2019-06-24		*
recon/domains-hosts/bing_domain_web	1.1	not installed	2019-07-04		
recon/domains-hosts/brute_hosts	1.0	not installed	2019-06-24		
recon/domains-hosts/builtwith	1.1	not installed	2021-08-24		*

- We see many subcategories under recon, such as **domains-companies**, **domains-contacts** and **domains-hosts**.
- The naming tells us what kind of new information we will get from that transformation.
- Example- **domains-hosts**
- Modules like **whoxy_whois**, requires a key, as we can tell from the  under the **K** column.
- This indicates that this module is not usable unless we have a key to use the related service.

- Other modules have dependencies, indicated by a `*` under the **D** column.
- To learn more about any module, we can run - `marketplace info MODULE`
- For example-

```
[recon-ng][thmredteam] > marketplace info google_site_web
```

path	recon/domains-hosts/google_site_web
name	Google Hostname Enumerator
author	Tim Tomes (@lanmaster53)
version	1.0
last_updated	2019-06-24
description	Harvests hosts from Google.com by using the 'site' search operator. Updates the 'hosts' table with the results
required_keys	[]
dependencies	[]
files	[]
status	not installed

- We can install the module we want

```
[recon-ng][thmredteam] > marketplace install google_site_web
[*] Module installed: recon/domains-hosts/google_site_web
[*] Reloading modules...
[recon-ng][thmredteam] > modules search
```

Recon
recon/domains-hosts/google_site_web

Working with Installed Modules

- To get a list of all modules, we can use - `modules search`
- To load a specific module to memory, we can use - `modules load MODULE`

```
[recon-ng][thmredteam] > modules search
```

Recon
recon/domains-hosts/google_site_web

```
[recon-ng][thmredteam] > modules load google_site_web
[recon-ng][thmredteam][google_site_web] > █
```

- To run it, we need to set the required options.

- To list the options that we can set for the loaded module, we can use- `options list`.
- To set the value for the option, we can use - `options set <option> <value>`.
- We run the module using- `run`.

```
[recon-ng][thmredteam][google_site_web] > options list
```

Name	Current Value	Required	Description
SOURCE	default	yes	source of input (see 'info' for details)

```
[recon-ng][thmredteam][google_site_web] > info
```

```

Name: Google Hostname Enumerator
Author: Tim Tomes (@lanmaster53)
Version: 1.0

Description:
Harvests hosts from Google.com by using the 'site' search operator. Updates the 'hosts' table with
the results.

Options:

```

Name	Current Value	Required	Description
SOURCE	default	yes	source of input (see 'info' for details)

```

Source Options:
default      SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
<string>     string representing a single input
<path>       path to a file containing a list of inputs
query <sql>  database query returning one column of inputs

```

```
[recon-ng][thmredteam][google_site_web] > run
```

```

THMREDTEAM.COM
[*] Searching Google for: site:thmredteam.com
[*] Country: None
[*] Host: cafe.thmredteam.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*]
[*] Country: None
[*] Host: clinic.thmredteam.com
[*] Ip_Address: None
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*]
[*] Searching Google for: site:thmredteam.com -site:cafe.thmredteam.com -site:clinic.thmredteam.com
[*] No New Subdomains Found on the Current Page. Jumping to Result 201.
[*] Searching Google for: site:thmredteam.com -site:cafe.thmredteam.com -site:clinic.thmredteam.com

```

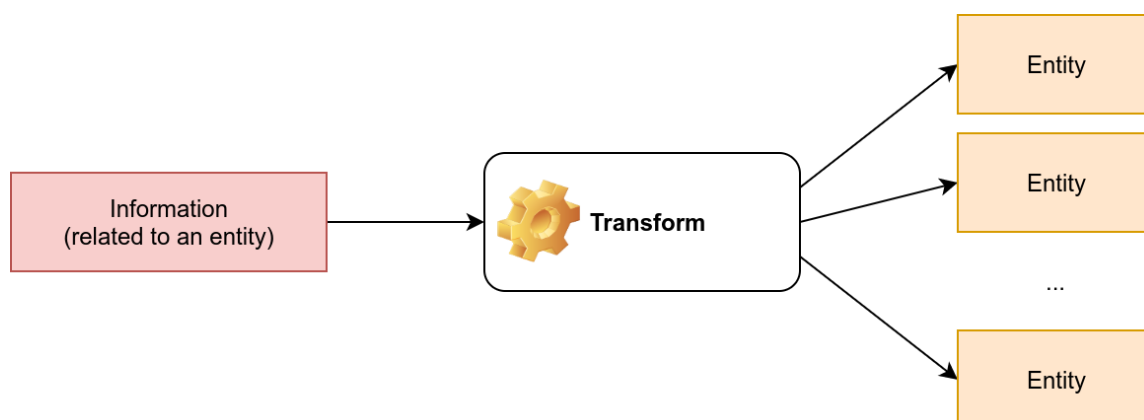
Keys

- Some modules cannot be used without a key for the respective service API.
- **K** indicates that we need to provide the relevant service key to use the module in question.
- To list the keys- `keys list`.

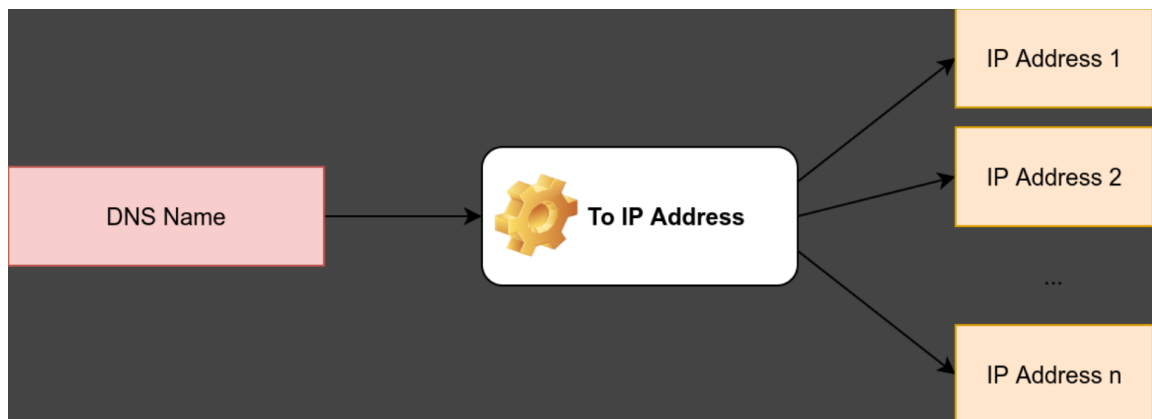
- To add a key- `keys add KEY_NAME KEY_VALUE` .
- To remove a key- `keys remove KEY_NAME` .

Maltego

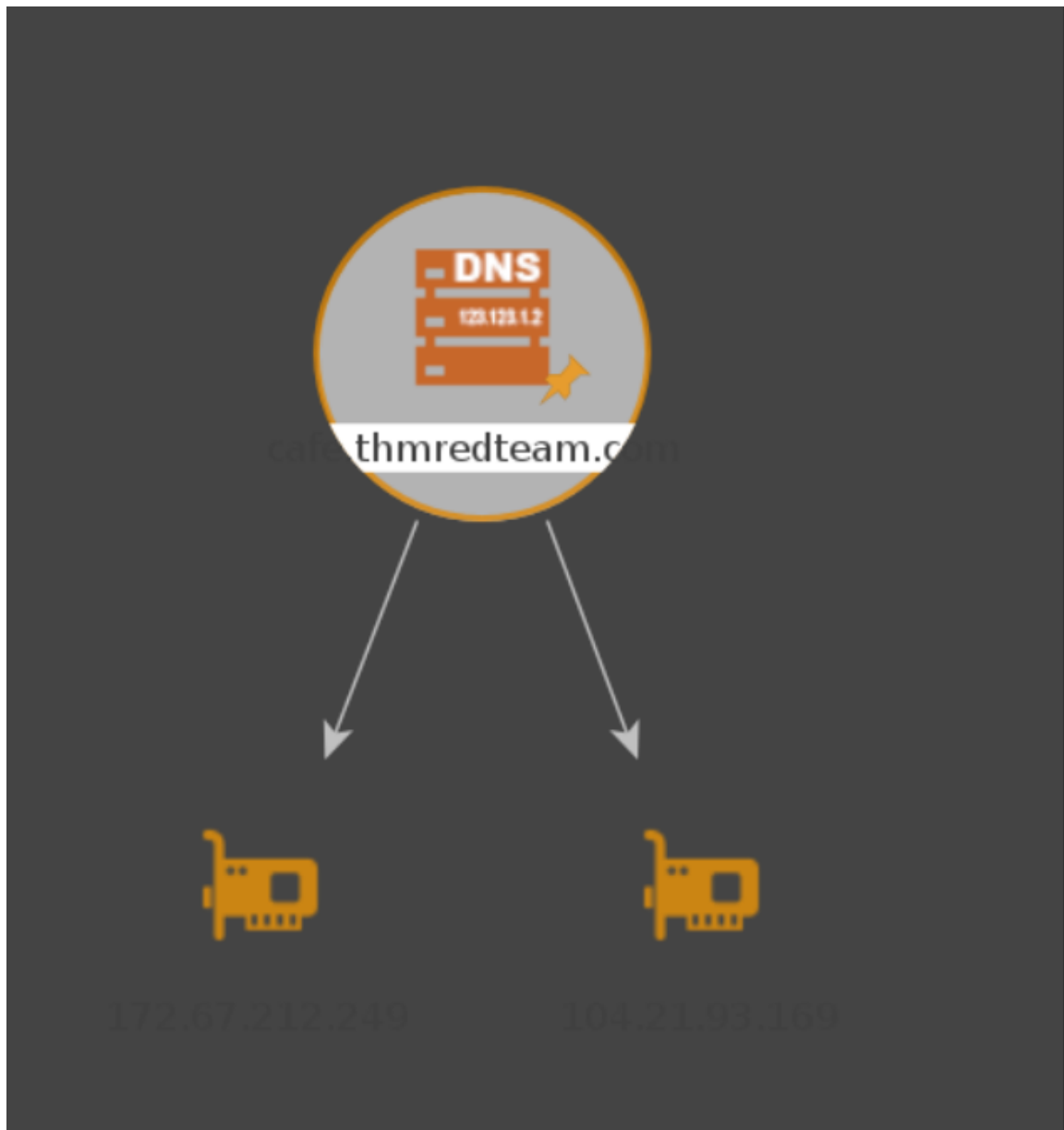
- It is an application that blends mind-mapping with OSINT.
- The information collected in Maltego can be used for later stages.
- For example - company information, contact names, and email addresses collected can be used to create very legitimate-looking phishing emails.
- In Maltego's terminology, a **transform** is a piece of code that would query an API to retrieve information related to a specific entity.
- Information related to an entity goes via a transform to return zero or more entities.



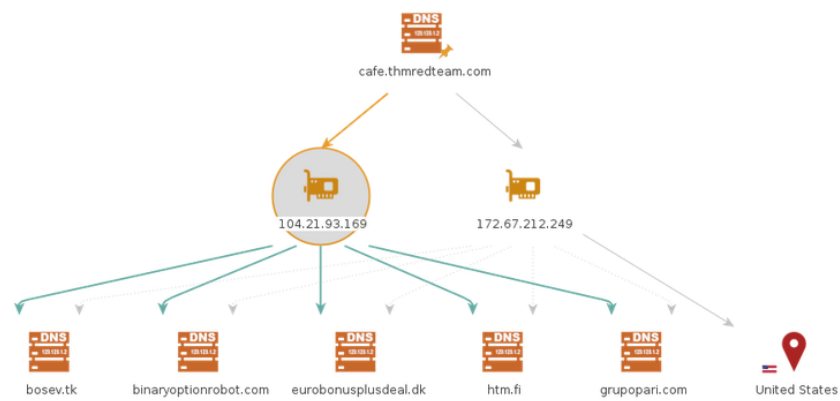
- Some of the transforms available in Maltego might actively connect to the target system.
- Every transform might lead to several new values.
- For example - Starting from the DNS name [cafe.thmredteam.com](https://www.thmredteam.com) , we expect to get new kinds of entities based on the transform we use.
- Example - "To IP Address" is expected to return IP addresses.



- We can right click on the DNS name - "cafe.thmredteam.com" and choose:
 - Standard Transforms
 - Resolve to IP
 - To IP Addresses (DNS)
- After executing this transform, we would get one or more IP addresses.



- Then we can choose to apply another transform for one of the IP addresses, like:
 - DNS from IP
 - To DNS Name from passive DNS



- The results of `whois` and `nslookup` are shown visually in the following Maltego graph.

