

RECONNAISSANCE

NOT ALWAYS ABOUT RESOURCES

ABSTRACT

Reconnaissance is the most important part of information security assessment activities, especially in Black-Box Penetration Testing and Bug Hunting activities. The collection of this information can be used as a target in testing to find the possibility that one of the subdomains has information related to the main domain, thereby increasing the chances of finding vulnerabilities.

REDHO MALAND

Information Security Consultant / Penetration Tester



KEY

Information is your key to success
when you become Bug hunter and
Penetration tester

2020.idsecconf.org /

Information Of Everything [IoE]

At this time information is everything and also the key to success when doing penetration testing and bug bounty activities. The first step performed by penetration tester and bug hunter is to get as much information as possible related to the target and learn through them or can be called as Information Gathering / Reconnaissance.



RECONNAISSANCE

NOT ALWAYS ABOUT RESOURCES

REDHO MALAND ARESTA

For information, in this case i will use the term reconnaissance or we can say recon. The recon and scanning procedures are repetitive, monotone, and time-consuming. Therefore we must optimize this to be more effective and efficient, to minimize the effort we spend.

But how or in what ways (?)

Do we use methodology or workflow that exist on internet ? Do we use a lot of resource, like combine all of engine/tools recon on internet? and it's that enough if we just copy and paste one liners command which we obtain through bug bounty tips from twitter?

In this post, we will further discuss all the above things and I am going to share my point of view towards those.

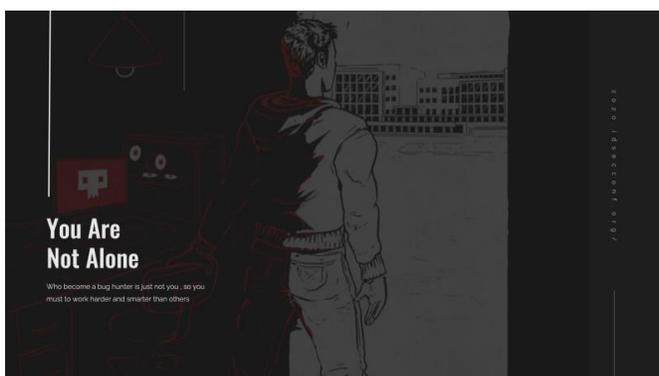
04:20

Fight Against Time

- Penetration Testing - Timebox
- Bug bounty - Race

Fight Against Time [00:00]

When we talk about penetration testing and bug bounty, the only thing we want to try to beat is the time. For example, usually penetration testing tied to a predetermined time (Time box) and also bug bounty is almost the same. There is no second place in bug bounty, even if there was, the report would have been a duplicate. So it's like a race, *The Early Bird Catches The Worm*.



You Are Not Alone

Keep (it) in mind, who become a bug hunter is not just you, so you must work harder and smarter than the others. Well, as I said earlier, the first thing they do is recon. This step is very important, because, if we don't do it well might be difficult to find a vulnerability or security

issues. And if we overdo it, will only waste our time. So it's important for us to maintain the balance in here. Before going there, first i will explain more the concept of recon, so that all of us have the same fundamental knowledge of understanding the meaning of recon.

Reconnaissance [rə'känəsəns]

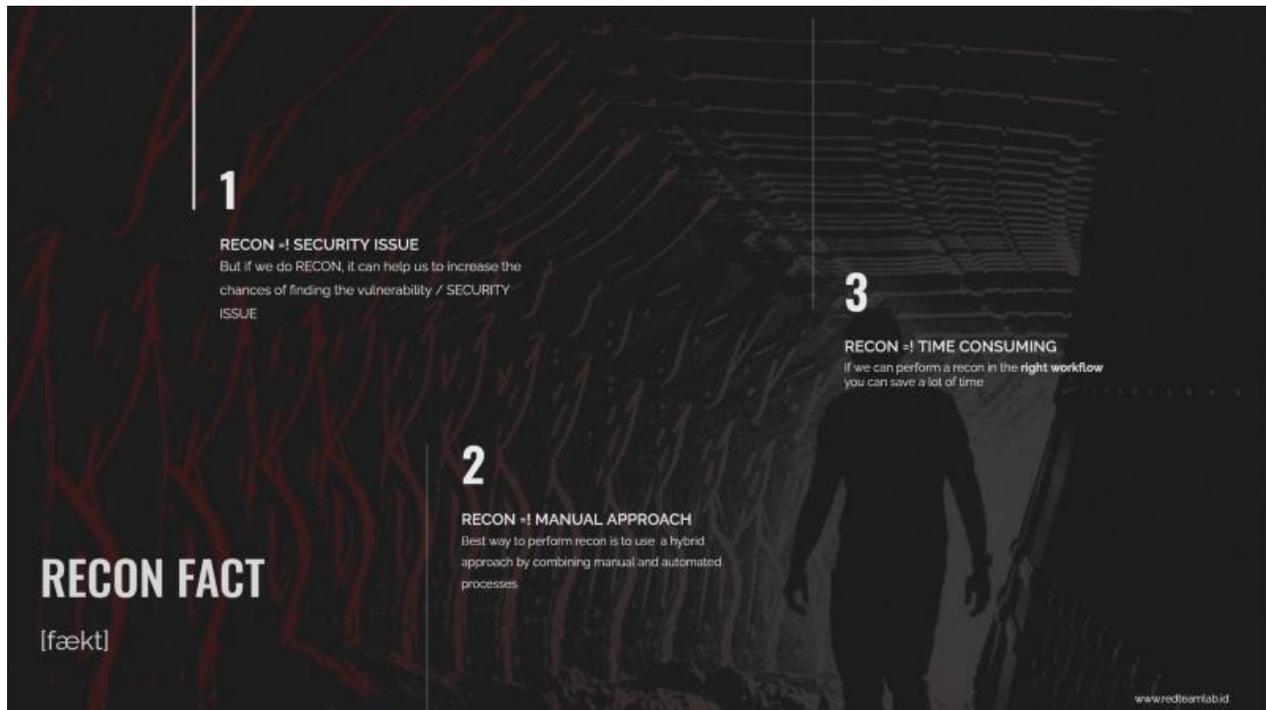


In short reconnaissance is a method used by pentester and bug hunters to get as much information as possible related to the target and learn through them. The two different stages of recon are active and passive.

During the passive reconnaissance stage, an attacker will use indirect methods to gather information from publicly available sources in internet like using third-party sites such as Google, Webarchive, Shodan, Zoomeye, Spyse, and etc, to get information. Once an attacker has collected as much public information as possible then move on to active recon. Active Reconnaissance the opposite of passive reconnaissance, which means we interact directly with the target or organization like port scanning, ping sweep, bruteforce attack and etc.

Reconnaissance Fact & Myths

Here are the most common facts and myths about the recon.



Reconnaissance != Security Issue

The first one recon is not same as security issues, but if we do recon it can help us to increase the chances of finding the vulnerability or security issues. For example finding hidden endpoints or sensitive files.

Reconnaissance != Time Consuming

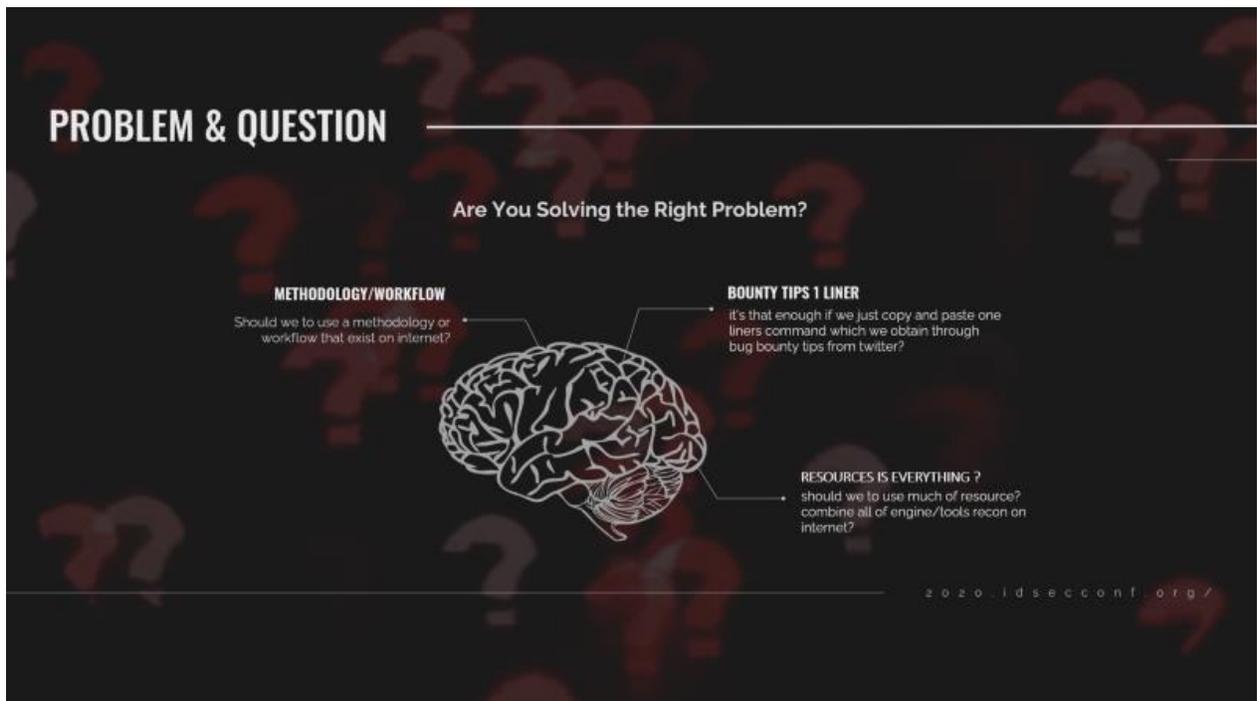
Recon is not a complete waste of time, if we do it with the right workflow and on target. So that we can save a lot of time and be more efficient

Reconnaissance != Manual Approaching

When perform recon, we don't always have to do it manually. Best way to perform recon is to use a hybrid approach by combining manual and automated processes

Are You Solving The Right Problem [?]

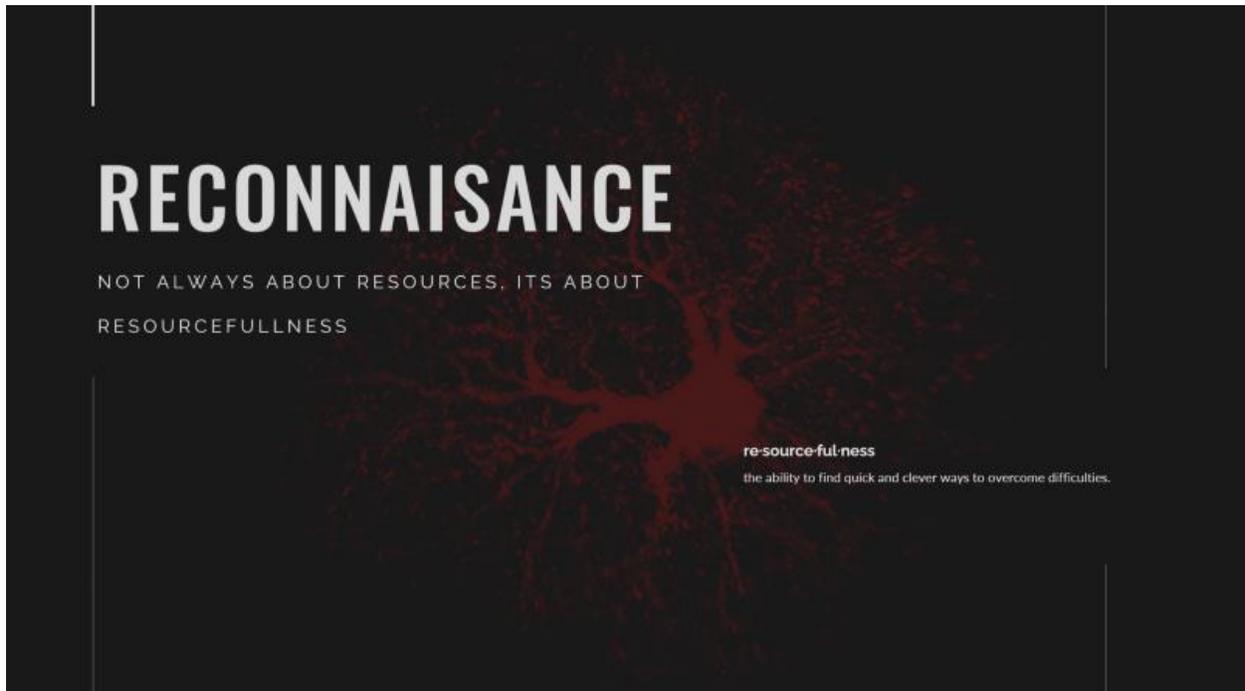
So at the beginning, I mentioned it. Do we have to use the methodology / recon workflow a, b, c, d on the internet? From my subjective, the answer is no, because there is no standard methodology for bug hunting especially when we doing reconnaissance.



Then is it enough, if we just copy and paste, the one liners command obtained through the bug bounty tips from twitter? The answer is not enough, because if we just copy and paste the command everybody is doing that. So we're not unique here and sometimes most people run the tool blindly which is where they do not understand the output given and will make them confused. It's also a waste of time.

Resources is Everything [?]

When performing reconnaissance, should we to use much of resource? like combine **all** of engine and tools recon on internet?. The answer is no, because use all of engine and tools is not always a good idea and is sometimes a waste of time.



And should be underlined in performing reconnaissance, it does not invariably depend on the plenty of resources. Here, we require creativity (resourcefulness) to overcome a hurdle and figure out how to turn the resources into a better results, so

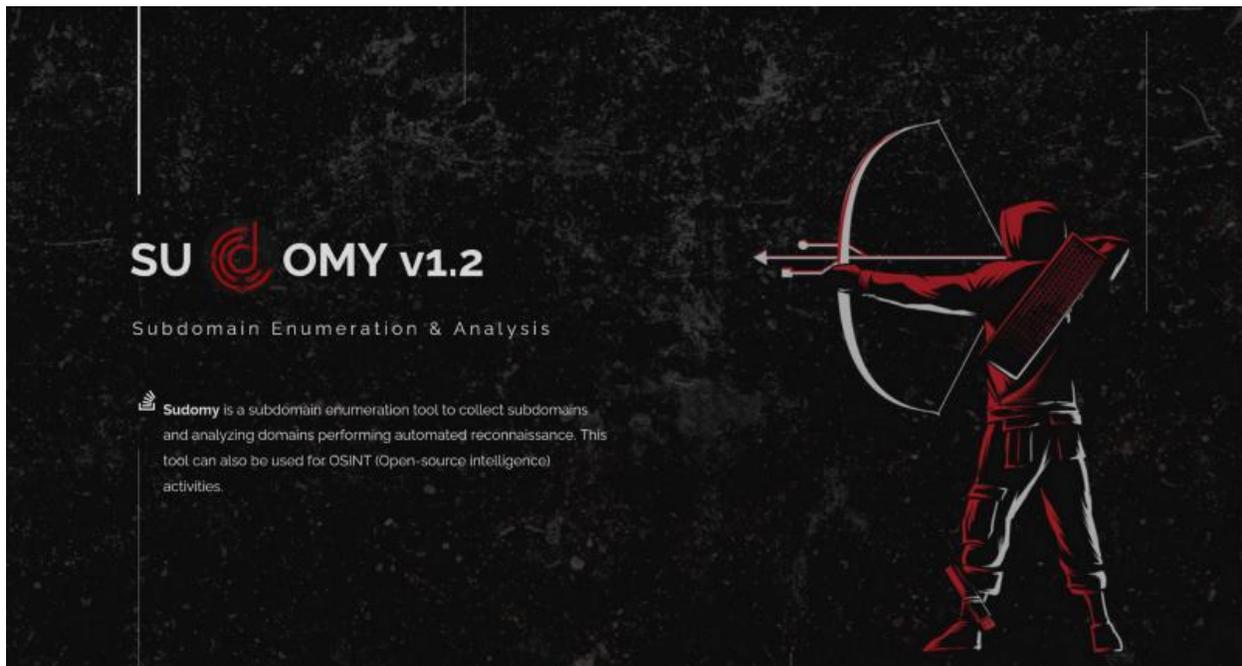
“Reconnaissance not always about Resources,
it is about Resourcefulness”

The pre-eminent way to optimize the recon process is to create a recon workflow and automation tool on your own, by optimizing the utilization of the resources. Hence, we can minimize the effort when doing recon.



What I am trying to highlight here is that does not mean I am a creative type of person and have a good methodology or workflow. I personally still in the progress of learning, however, my count is this might be able to be a new insight to all of my friends. Accordingly, I would like to share about my recon workflow, tools, point of view, and some ideas. Leastwise, this might work and help to increase the effectiveness when doing pentest & bug hunting. Therefore, the tool that I will introduce (on this page) is Subdomain Enumeration & Analysis or so can be called Sudomy.

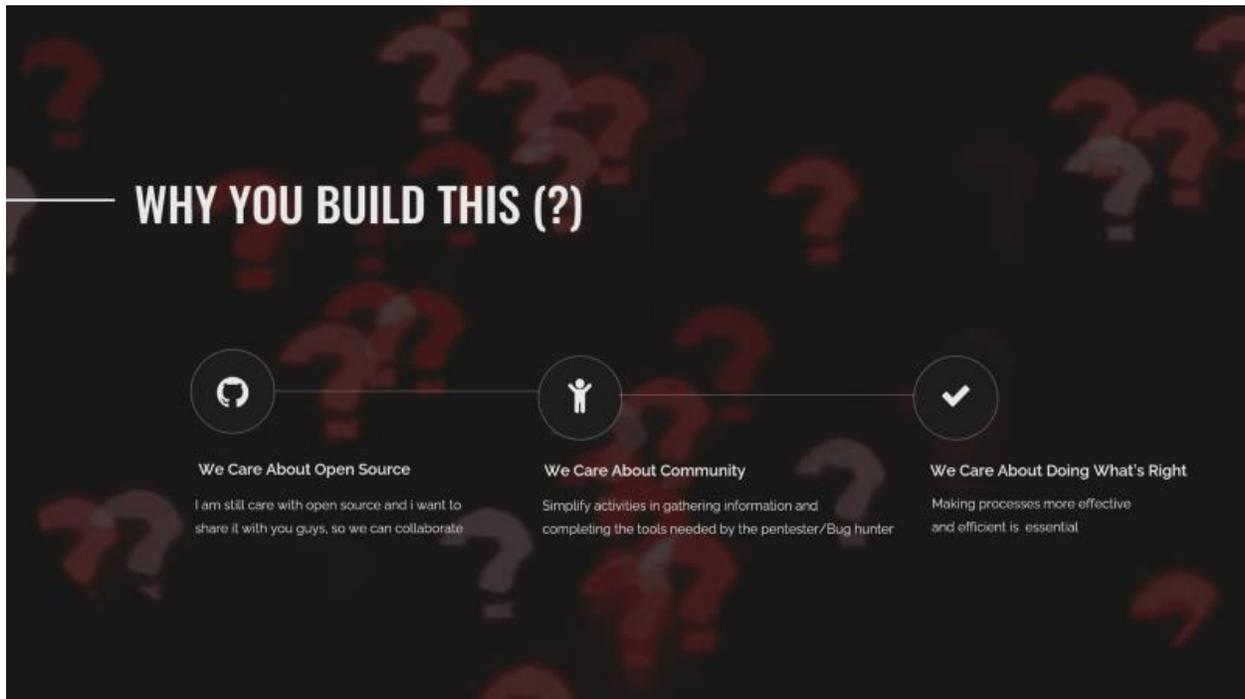
Subdomain Enumeration & Analysis



Sudomy (Subdomain Enumeration & Analysis) is a subdomain enumeration tool to collect subdomains and analyzing domains performing automated reconnaissance. This tool can also be used for OSINT (Open-source intelligence) activities.

Development of Information Gathering applications follows Information Systems Security Assessment Framework (*ISSAF*) rules by applying two techniques, namely passive and active. Passive techniques obtain information through a number of ways by utilizing third-party resources such as using Web APIs, Information Gathering libraries or through OSINT Source with scraping processes. While the active technique uses applications that are installed with similar features, namely the Information Gathering function either by brute force, word lists or other new methods like httpprobe & httpx for validation and more.

Why You Build This [?]



Firstly **We Care About** Open Source, i am still care with open source and i want to share it with you guys, so we can collaborate. Secondly, **We Care About** Community, Simplify activities in gathering information and completing the tools needed by the Penetration Tester & Bug hunter. And the last but not least, **We Care About** Doing What's Right to making processes more effective and efficient is essential

Comparison [/kəm'parɪs(ə)n/]



It's natural who want to compare this tool with other tools, because i also understand that there are many impressive tools out there that are similar like Sublist3r and Subfinder.



Minimize More Resources When Use Third-Party Sites

By evaluating and **selecting** the **good** third-party sites/resources, so the enumeration process can be **optimized**.

2020.1dsecconf.org/

The main difference between Subfinders and Sublister is in the utilization of resources. Here Sudomy minimizes more resources when using Third-Party Sites by evaluating and selecting good third-party sites/resources, so the enumeration process can be optimized. For example Sudomy does not use third party resources such as Google, Baidu, Ask Yahoo and Bing. Because the results obtained from these third-party resources are not optimal and there are also other factors such as being hampered by the captcha. This is the result of collecting subdomains use third party resources such as Google, Baidu, Ask Yahoo and Bing.



In fact, the result obtained from the search engines such as Yahoo and Bing have been collected in one third party resources such as SecurityTrails, dnsdb or webarchive. Therefore, I did not include search engines as a resource in collecting subdomain data. So that, the results obtained are still optimal and also less time consuming.

In testing Information Gathering tools using several supporting devices both hardware and software, here we use the E2C AWS server with the following specifications. For example the target used in testing is domain `tiket.com`.

Requirements

1- Target
 Domain: `tiket.com`

2- Connection Speed
 Server: E2C AWS
 Download: 945.02 Mbit/s
 Upload: 914.69 Mbit/s

3- Materials
 Name: Ubuntu
 Version: 16.04.7 LTS (Xenial Xerus)
 Model name: Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz
 Architecture: x86_64
 CPU op-mode(s): 32-bit, 64-bit
 CPU & Core(s): 8

COMPARISON

RESOURCES

Totals	Resources	Treads/Multiprocess	Reals Times	Results
Sudomy	22 Engine	Multiprocessing	0m7.724s	662 subdomains
Subfinder	35 Engine	Concurrency (Threads 200)	0m7.844s	615 subdomains
Sublist3r	11 Engine	Multithreads (Threads 200)	0m12.434s	94 Subdomains

Testing is done by comparing the sudomy application with other applications such as *Sublist3r v1.1.0*, *Subfinder v2.4.5*, and *Sudomy v1.2.0* with the target domain `tiket.com`. The Subfinder application uses 35 resources/engine, Sublist3r uses 11 engine/resources, and Sudomy uses 22 engine/resources. The time needed to search for a subdomain from `tiket.com`, the Subfinder application takes 0 minute 7,844s, sublist3r takes 0 minutes 12,434s, and sudomy takes 0 minutes 7,724s. The subdomain results from `tiket.com` found by the subfinder application are 615 subdomains, sublist3r are 94 subdomains, and sudomy is 662 subdomains.

But Golang is Powerfull & Faster [?]

Exactly, im agree with you. Use go programming language is quite effective for doing a lot of work simultaneously or we can say with concurrency. For example in tools subfinder, here subfinder is still classified **as very fast** for **collecting subdomains** by utilizing quite a **lot of resources**. Especially if the resources used have been optimized (?)

For compilation results and videos, you can check here:

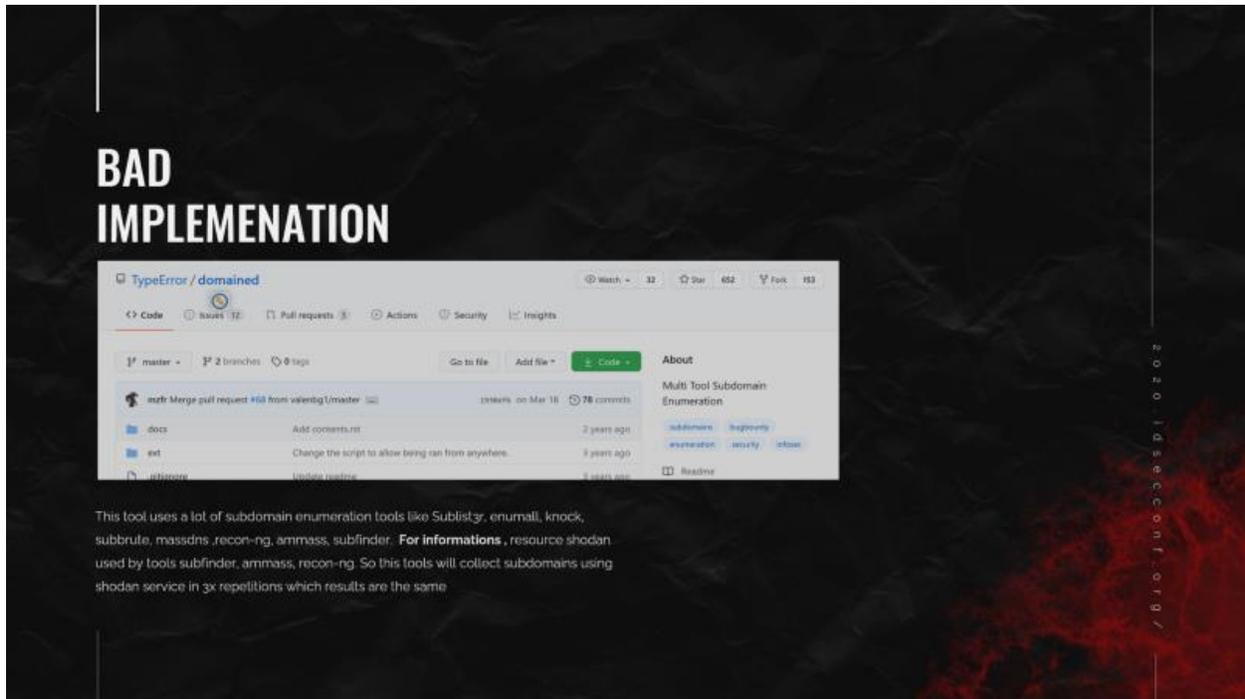
- [Sudomy](#)
- [Subfinder](#)
- [Sublist3r](#)

When I have free time. Maybe In the future, sudomy will use golang too. If you want to contributes, it's open to pull requests.



Bad Implementation

Well, this is an example of a tool that I think is less effective.

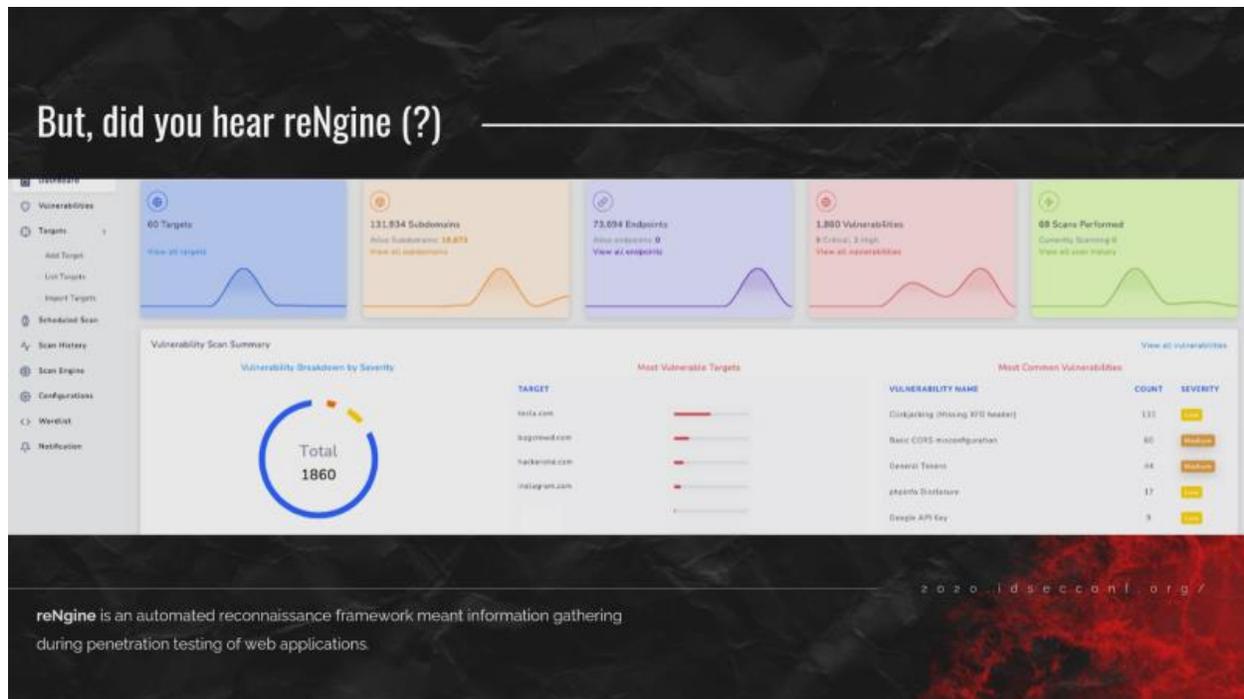


This tool uses a lot of subdomain enumeration tools like Sublist3r, enumall, knock, subbrute, massdns, recon-ng, ammass, subfinder. For informations, resource shodan used by tools subfinder, ammass, recon-ng. So this tools will collect subdomains using shodan service in 3x repetitions which results are the same

Domained collects subdomains by combining all subdomain enumeration tools such as sublist3r, enumall, knock, subbrute, massdns, recon-ng, ammass & subfinder. I think this is less effective and wastes too much time, because these tools use the same third-party resources for example resource shodan used by tools subfinder, ammass, recon-ng. So these tools will collect subdomains using shodan service in 3x repetitions which results are the same.

But, did you hear reNginer (?)

Of course, this tool is really cool and powerful, if you don't know what is reNginer. reNginer is an automated reconnaissance framework meant for information gathering during penetration testing of web applications. The beauty of reNginer is that it gathers everything in one place. It has a pipeline of reconnaissance, which is highly customizable and have web application interface to control that.



reNginer also collects subdomains by combining all subdomain enumeration tools such as sublist3r, subfinder, amass, assetfinder and etc. Another interesting feature of reNginer is port scanning and collecting endpoints or urls for each subdomain automatically. But in my opinion, the reconnaissance workflows are still not effective and efficient, so I'll try to explain it why.

reNgin Capabilities

reNgin capabilities

- **Subdomain Discovery:** Discovers all the subdomains using tools like sublist3r, subfinder, amass, assetfinder, etc.
- **Port Scan:** Use to identify the open ports on the subdomains that have been discovered. Currently reNgin uses **naabu** to check for open ports. We have plans to use **masscan** in the future.
- **Directory and File Search:** Uses **dirsearch** to discover the directories and files.
- **Fetch all Endpoints:** Fetches all the urls for each subdomains from various sources like Open Threat Exchange, Wayback machine, common crawl etc. reNgin uses **gau**, **hakrawler** to fetch the endpoints.
- **Vulnerability Scan (Beta):** reNgin uses **nuclei** to perform the Vulnerability Scans on the targets.

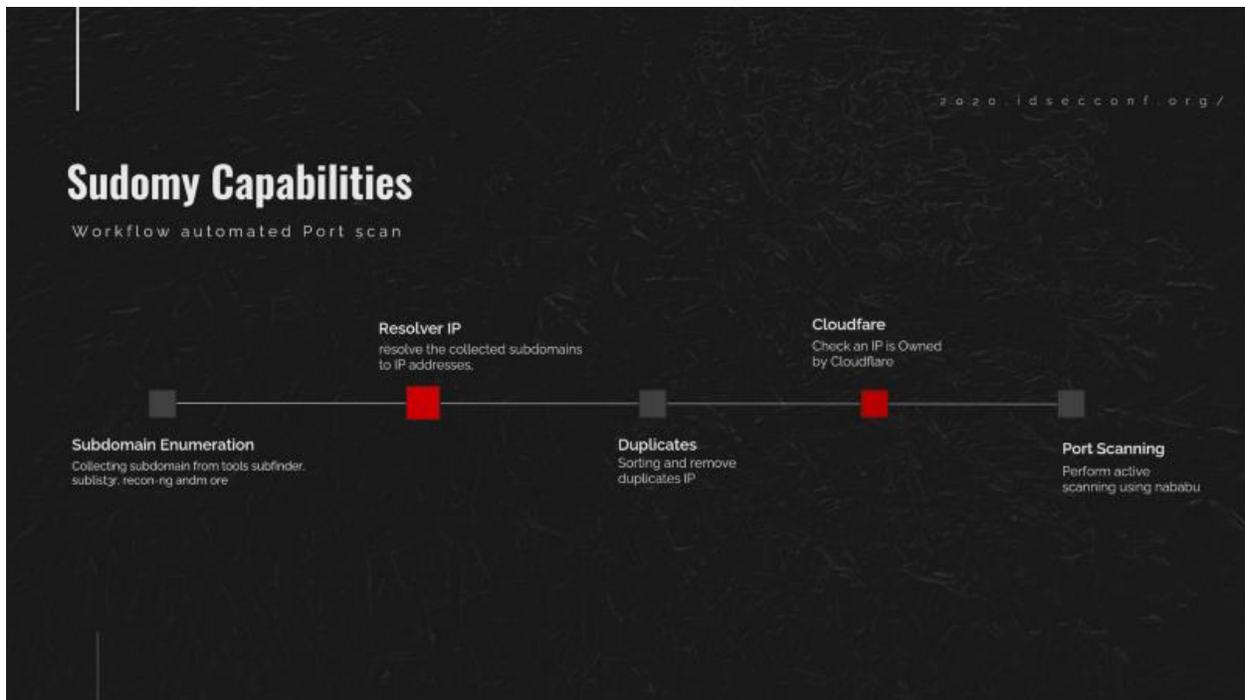
Workflow automated Port scan



reNgin Capabilities (Port scan), reNgin performs an active scan to get open ports using naabu and masscan. When performing port scanning reNgin doesn't check whether the collected IP address (Resolve) is protected by Cloudflare or not. So if the IP address Cloudflare is still included in the scanning list, it will take a lot of time, which shouldn't be necessary.

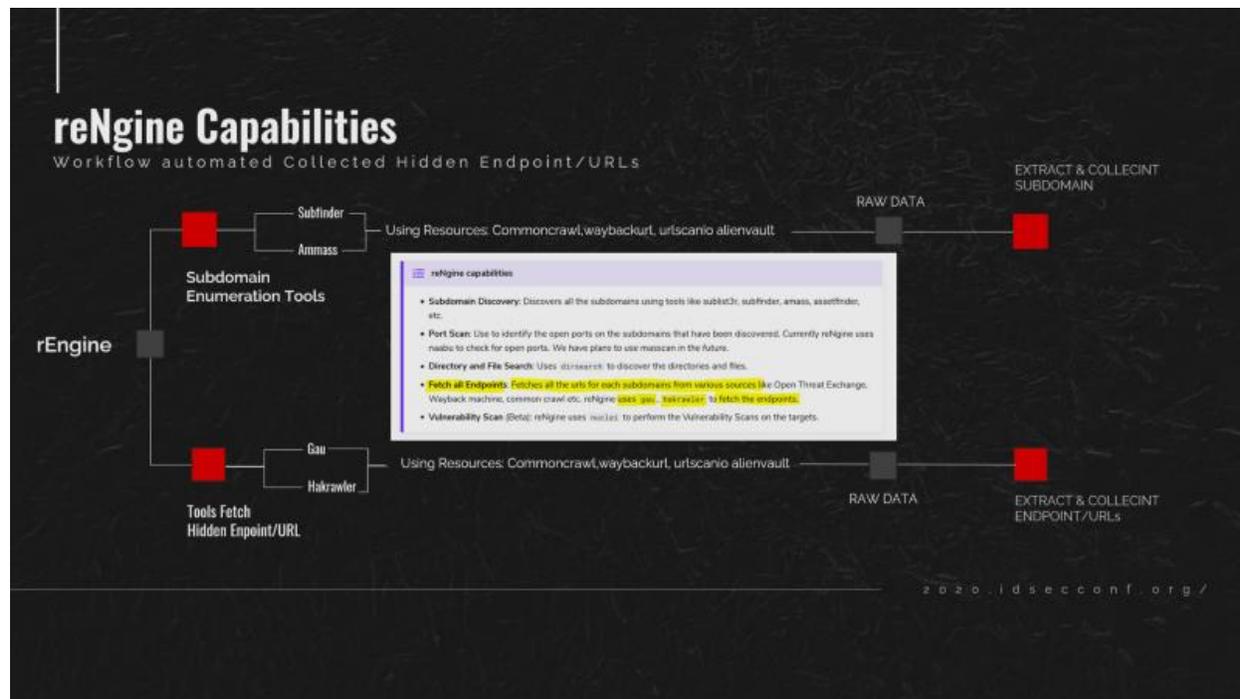


Meanwhile, Sudomy uses two methods to perform port scanning. Sudomy performs an active scan to get open ports using nmap and passive scans use third-party sites such as shodan. For right now the main source is shodan, in the future there will definitely be added other resources such as censys and zoomeye.



So, this are sudomy capabilities in port scanning, what makes Sudomy different from rEngine is that Sudomy will first check whether the IP address is protected by Cloudflare or not. Then if there is an IP address in the list that is protected by Cloudflare, the ip will not be scanned at all, So sudomy will only perform port scanning from unprotected IP Addresses. Then, the specialty of sudomy is in the passive scan which uses Shodan to collect open ports and all we have to do is taking advantage from that and validate it.

Next, for reNgin capabilities to fetch all the urls for each subdomains from various sources, i think this workflows are still not effective and efficient. Because to collect the endpoint / url, reNgin directly uses additional tools such as gau and hakrawler.



Which is where these tools gau and hakrawler, use same third-party site resources that are also used by Subfinder, and Masscan like Waybackurl, commoncrawl and urlscan. If you do not know what gau and hakwaler are, they are crawling tools that are used to collect information such as endpoints and urls from a domain, by utilizing third-party sites as previously explained. So here we can see that, the recon process here has two repetitions in resource utilization.

Sudomy Capabilities

Workflow automated Collected Hidden Endpoint/URLs



While sudomy doesn't do that., the first thing that sudomy will do is to make a requests and extract raw data from third-party resources such as Commoncrawl, Waybackurl, URLscanio. Then, sudomy will extract what information will be retrieved. Here sudomy will retrieve the subdomain information and url endpoint. Then stored into a separate files. Therefore, there is no repeating resources and requests here. Other than that, in terms of features, Sudomy is more complete in performing automated reconections to gather other information

Is The Same With as Findomain?

Almost similar, some of the features in findomain are almost covered by sudomy as well. What sets them apart is their commercial side.

IS THE SAME WITH AS FINDOMAIN?
COMMERCIAL VS NON-COMMERCIAL

Sudomy
is a subdomain enumeration tool to collect subdomains and analyzing domains performing automated reconnaissance. This tool can also be used for OSINT (Open-source intelligence) activities.

Findomain
is a complete reconnaissance solution for enterprises and cybersecurity specialists that uses cutting edge technology, able to send alerts about new subdomains, their HTTP status, HTTP content size, HTTP content type & more

Tool	Price	Features
Sudomy	- \$ / lifetime	Validation Subdomain HTTP Check [status code, title, content] Subdomain Screenshots Detection Virtualhost Identify technologies Webhook alerts
Findomain	59 \$ / Mo	Validation Subdomain HTTP Check [status code, title, content] Subdomain Screenshots Detection Virtualhost Identify technologies Webhook alerts

Free

Premium/VIP

Feature Sudomy

For recent time, *Sudomy* has these 20 features:

- Easy, light, fast and powerful. Bash script (controller) is available by default in almost all Linux distributions. By using bash script multiprocessing feature, all processors will be utilized optimally.
- Subdomain enumeration process can be achieved by using **active** method or **passive** method
- **Active Method**

Sudomy utilize Gobuster tools because of its highspeed performance in carrying out DNS Subdomain Bruteforce attack (wildcard support). The wordlist that is used comes from combined SecList (Discover/DNS) lists which contains around 3 million entries

- **Passive Method**

By evaluating and **selecting** the **good** third-party sites/resources, the enumeration process can be **optimized**. More results will be obtained with less time required. *Sudomy* can collect data from these well-curated 22 third-party sites:

```
https://censys.io
https://developer.shodan.io
https://dns.bufferover.run
https://index.commoncrawl.org
https://riddler.io
https://api.certspotter.com
https://api.hackertarget.com
https://api.threatminer.org
https://community.riskiq.com
https://crt.sh
https://dnsdumpster.com
https://docs.binaryedge.io
https://securitytrails.com
https://graph.facebook.com
https://otx.alienvault.com
https://rapiddns.io
https://spyse.com
https://urlscan.io
https://www.dnsdb.info
https://www.virustotal.com
https://threatcrowd.org
https://web.archive.org
```

- Test the list of collected subdomains and probe for working http or https servers. This feature uses a third-party tool, [httpprobe](#).
- Subdomain availability test based on Ping Sweep and/or by getting HTTP status code.
- The ability to detect virtualhost (several subdomains which resolve to single IP Address). *Sudomy* will resolve the collected subdomains to IP addresses, then classify them if several subdomains resolve to single IP address. This feature will be very useful for the next penetration testing/bug bounty process. For instance, in port scanning, single IP address won't be scanned repeatedly
- Performed port scanning from collected subdomains/virtualhosts IP Addresses
- Testing Subdomain TakeOver attack (CNAME Resolver, DNSLookup, Detect NXDomain, Check Vuln)

- Taking Screenshots of subdomains default using gowitness or you can choice another screenshot tools, like (-ss webscreenshot)
- Identify technologies on websites (category,application,version)
- Detection urls, ports, title, content-length, status-code, response-body probbing.
- Smart auto fallback from https to http as default.
- Data Collecting/Scraping open port from 3rd party (Default::Shodan), For right now just using Shodan [Future::Censys,Zoomeye]. More efficient and effective to collecting port from list ip on target [[Subdomain > IP Resolver > Crawling > ASN & Open Port]]
- Collecting Juicy URL & Extract URL Parameter (Resource Default::WebArchive, CommonCrawl, UrlScanIO)
- Collect interesting path (api|.git|admin|etc), document (doc|pdf), javascript (js|node) and parameter
- Define path for outputfile (specify an output file when completed)
- Check an IP is Owned by Cloudflare
- Generate & make wordlist based on collecting url resources (wayback,urlscan,commoncrawl. To make that, we Extract All the paramater and path from our domain recon
- Generate Network Graph Visualization Subdomain & Virtualhosts
- Report output in HTML & CSV format
- Sending notifications to a slack channel

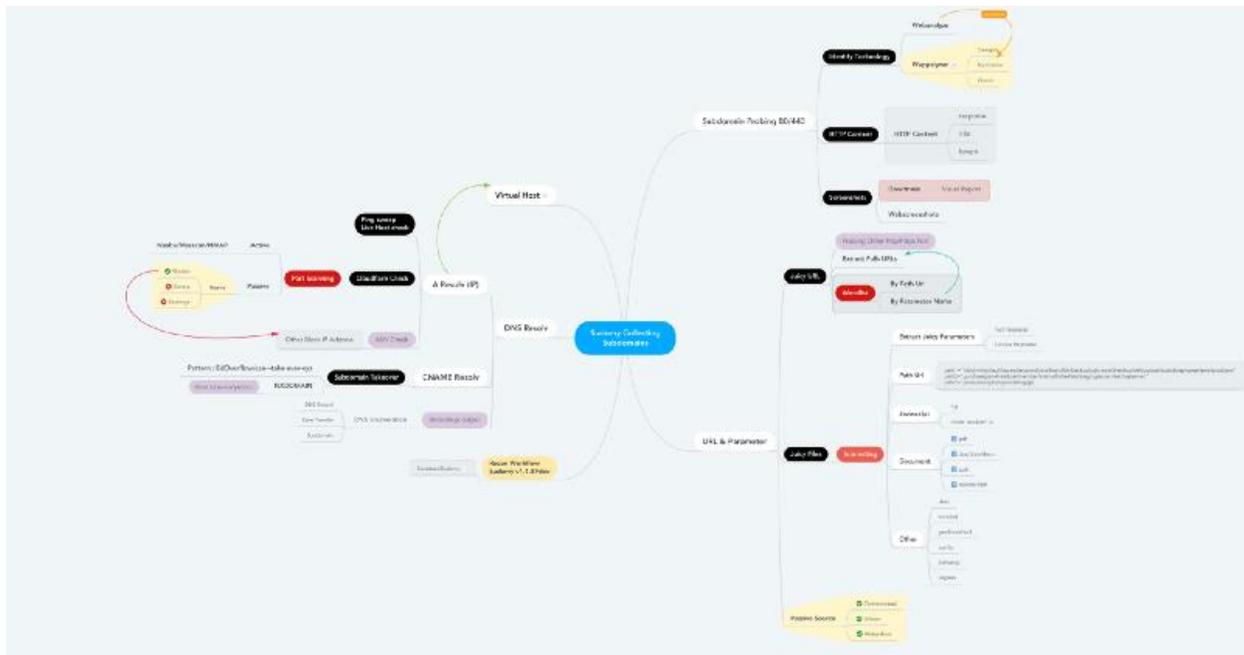
How Sudomy Works

How sudomy works or recon flow, when you run the best arguments to collect subdomains and analyze by doing automatic recon.

```
root@maland: ./sudomy -d bugcrowd.com -dP -eP -rS -cF -pS -tO -gW --httpx --dnsprobe -aI webanalyze -sS
```

Recon Workflow

This Recon Workflow Sudomy v1.1.8#dev



First thing, sudomy will collect subdomains from the main domain using a passive method through the selected third-party resources such as censys, shodan, bufferover, commoncrawl, riddler, certspotter, hackertarget, threatminer, riskiq, crtsh, dnsdumpster, binaryedge, securitytrails, facebookm, alienvault, rapiddns, spyse, urlscan, dnsdb, virustotal, threatcrowd and webarchive. To improve the enumeration result, the sudomy application needs to add the API Key for Shodan, Censys, Virus Total, BinaryEdge, and SecurityTrails in the configuration file sudomy.api.

During the subdomain collection process, sudomy also fetches raw data (without filtering) from certain resources such as CommonCrawl, UrlScan, Webarchive and Shodan. Because the raw data from these resources can be processed and used at the same time to obtain other information. For example port information, asn number, path, url, parameters and other interesting files such as api, git, admin, javascript (js, node_module) and documents (doc, pdf, pub, xlsx) which can be used for generate wordlists.

From the subdomain list, sudomy will validate the active subdomain by checking the http / https protocol automatically. This feature uses third party tools such as HTTProbe. Not only that, sudomy will also check the title, content-length, status-code and response-body on each active subdomain. Then from the list of active subdomains, sudomy will continue the enumeration process by identifying the web technologies used such as the Content Management System (CMS), Bootstrap, Web Server, Operating System and the database used by the website.

With a collection of subdomains successfully collected, sudomy will detect multiple subdomains that are divided into one IP address (virtualhost) and classify them and choose if multiple subdomains resolve to the same IP address.

```
- blog.bugcrowd.com
- www.bugcrowd.com
52.5.214.175
- stargate.a.bugcrowd.com
52.6.24.219
- tableau.a.bugcrowd.com
3.230.87.225
- production-sandbox.a.bugcrowd.com
104.20.4.239
- bugcrowd.com
- tracker.bugcrowd.com
104.20.61.51
- api.bugcrowd.com
- assetinventory.bugcrowd.com
- concourse.bugcrowd.com
- email.crowdcontrol.bugcrowd.com
- email.submit.bugcrowd.com
- hooks.bugcrowd.com
52.44.177.143
- gemstash-mattress.a.bugcrowd.com
192.28.152.174
- bounce.bugcrowd.com
66.220.12.139
- forum.bugcrowd.com
3.222.95.161
- proxilate.a.bugcrowd.com
- proxilate.bugcrowd.com
34.194.216.57
- collateral.bugcrowd.com
104.17.74.206
- go.bugcrowd.com
- ww2.bugcrowd.com
34.198.111.148
- tracker.production-sandbox.a.bugcrowd.com
104.20.60.51
- email.assetinventory.bugcrowd.com
- email.bugcrowd.com
- email.bugs.bugcrowd.com
- email.forum.bugcrowd.com
- gslink.bugcrowd.com
```

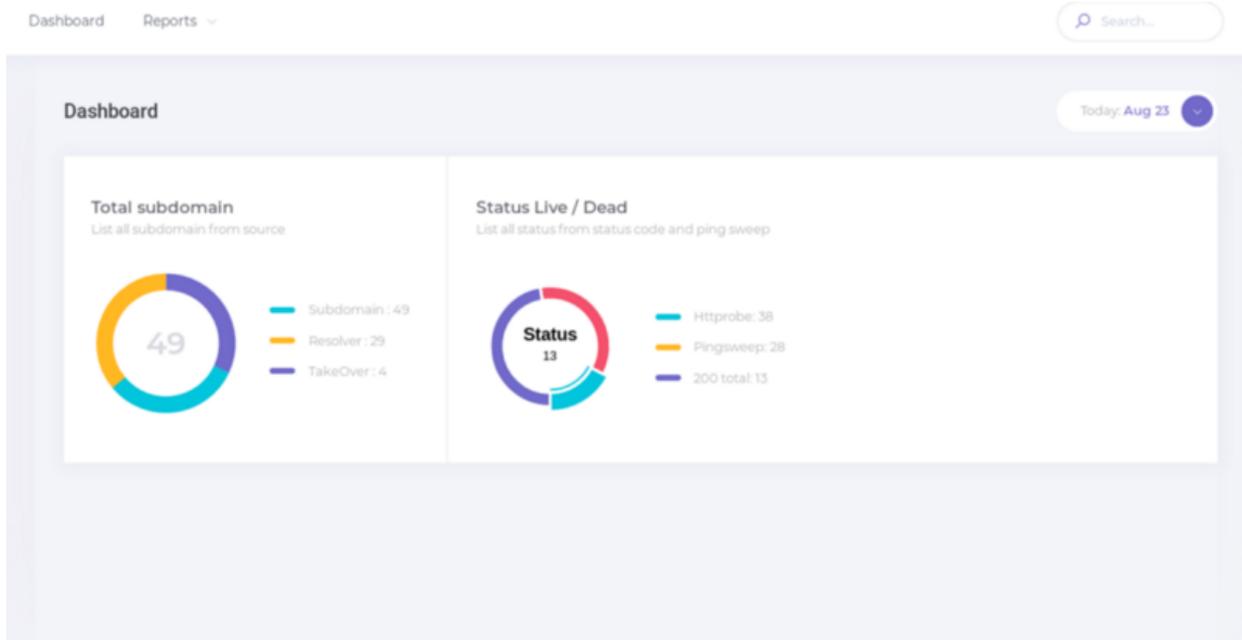
After the IP list is collected, sudomy will check the host based on the ping sweep and also check if the IP is owned / protected by Cloudflare. Then sudomy will performs port scanning through a list of the original ip addresses that have been filtered, here sudomy uses two methods to perform port scanning.

In active scanning sudomy uses nmap and for passive scanning, for now, the main source is shodan, in the future, other resources such as censys and zoomeye will be added.

```
[*] Collecting/Scraping open port from Engine
-----
o 104.17.71.206
  - 8080/tcp 80/tcp 443/tcp
o 104.17.72.206
  - 80/tcp 443/tcp
o 104.18.211.56
  - 443/tcp 80/tcp 8080/tcp 2086/tcp 8443/tcp
o 104.20.4.239
  - 2087/tcp 8080/tcp 8080/tcp 443/tcp 8443/tcp 2086/tcp 80/tcp 2083/tcp 2095/tcp 2052/tcp
o 104.20.5.239
  - 2087/tcp 443/tcp 2083/tcp 8080/tcp 80/tcp 2095/tcp 8080/tcp 2086/tcp 8443/tcp 2096/tcp 2082/tcp
o 104.20.60.51
  - 8080/tcp 8080/tcp 443/tcp 80/tcp 2082/tcp 2086/tcp 2083/tcp 2087/tcp 8443/tcp
o 104.20.61.51
  - 2082/tcp 443/tcp 2086/tcp 8080/tcp 2087/tcp 8080/tcp 2083/tcp 80/tcp 8443/tcp
o 192.20.152.174
```

Not only that, sudomy also checks the possible subdomain takeover and creates a custom wordlist based on the information that has been gathered. In creating a wordlist, sudomy uses resources such as CommonCrawl, UrlScan, Webarchive and Shodan to get Path, Url and Parameter information about the target. So the wordlist used is more specific, saves time and is right on target.

When the information gathering stage has been completed, sudomy will taking screenshots of the subdomains list and generate a HTML & CSV report which can help Penetration Tester, Bug Hunter, Researchers and Cyber security analysis in analyzing



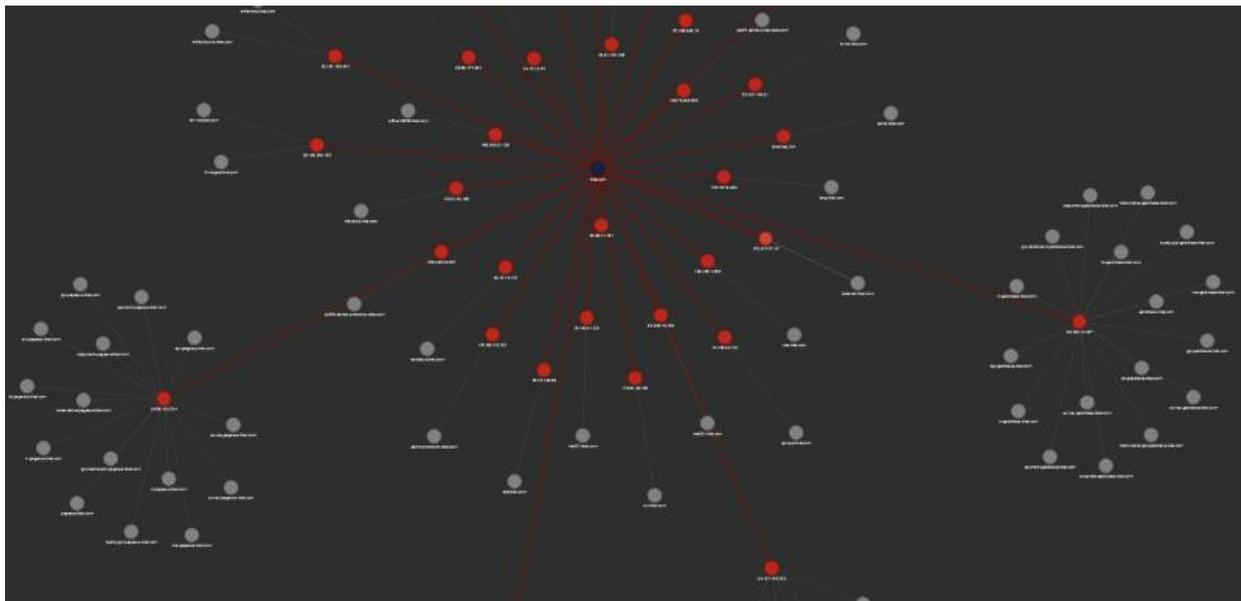
The dashboard shows a detailed view of subdomains and their hosts. The table below lists the subdomains and their IP addresses.

Subdomains	Hosts
1 alt3.aspxm.l.google.com.	209.85.200.26
2 alt4.aspxm.l.google.com.	64.233.177.26
3.aspxm.l.google.com.	74.125.200.26
4 alt1.aspxm.l.google.com.	74.125.28.26
5 alt2.aspxm.l.google.com.	64.233.179.26
6 api.bugcrowd.com	104.20.60.51
7 blog.bugcrowd.com	104.20.5.239
8 bounce.bugcrowd.com	192.28.152.174
9 bugcrowd.com	104.20.4.239
10 collateral.bugcrowd.com	3.219.144.242
11 concourse.bugcrowd.com	104.20.60.51

If the screenshot and report creation stages are complete, then the final stage will be entered, namely a notification that the scanning has been completed using the slack notification

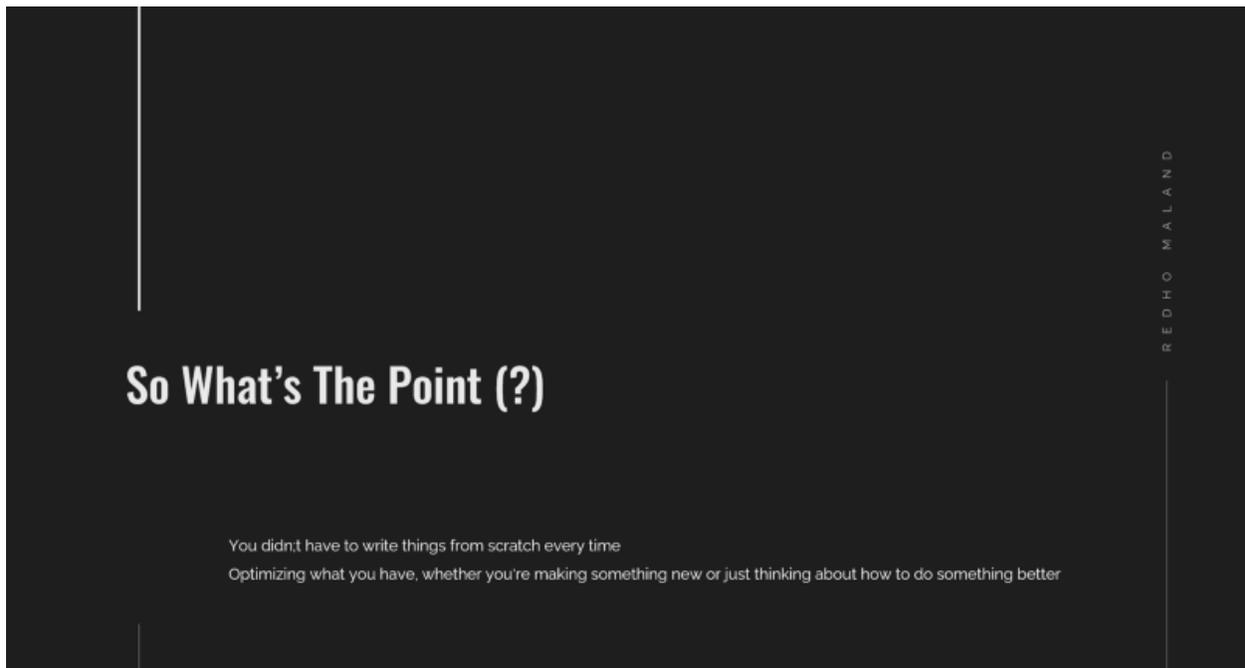


Besides that, sudomy also creates a statics html report to visualize subdomains and virtualhosts in graphical form as shown below:



So What's The Point [?]

The point here is to stay creative in solving problems / obstacles and being unique, creating your own version of the methodology and tools. You didn't have to write things from scratch every time.



Optimizing what you have, whether you're making something new or just thinking about how to do something better

References

Title	Reference URLs
Marsoni, T. U. Kalsum, and A. Kurniawan, "Analisa Implementasi Teknik Reconnaissance Pada Webserver (Studi Kasus: Upt Puskom Universitas Dehasen)," <i>J. Media Infotama Anal. Implementasi Tek. ISSN</i> , vol. 12, no. 1, pp. 1858–2680, 2016, [Online]	http://jurnal.unived.ac.id/index.php/jmi/article/viewFile/268/249 .
P. V. Mockapetris, "Domain names - implementation and specification," pp. 1–55, 1987, [Online].	http://tools.ietf.org/html/rfc1035
H. Yokosuka <i>et al.</i> , "Sinaci A A, Sehitoglu O T, Yondem M T, Fidan G and Tatli I 2010 SEMbySEM in Action: Domain Name Registry Service Through a Semantic Middleware eChallenges	https://www.istage.ist.go.jp/article/jfsr/20/3/20_193/_article/-char/ja
Institute for Security and Open Methodologies 2010 The Open Source Security Testing Methodology Manual	https://www.isecom.org/OSSTMM.3.pdf
Scarfone K, Souppaya M, Cody A and Orebaugh A 2015 NIST SP 800-42: Guideline on Network Security Testing vol 115 (Gaithersburg, MD)	https://www.nist.gov/publications/guideline-network-security-testing
Open Information Systems Security Group 2006 Information Systems Security Assessment Framework (ISSAF)	https://untrustednetwork.net/files/issaf0.2.1.pdf
A. Mendoza and G. Gu, "Mobile Application Web API Reconnaissance: Web-to-Mobile Inconsistencies & Vulnerabilities," <i>Proc. - IEEE Symp. Secur. Priv.</i> , vol. 2018-May, pp. 756–769, 2018,	https://ieeexplore.ieee.org/document/8418636/
Sudomy: Semi-automated Information Gathering Tools for	https://www.semanticscholar.org/paper/Sudomy%3A-Information-Gathering-Tools-for-Subdomain-Ramadhan-Aresta/919198cdb3d48d01e8845fc16d0a0c8a5f6522f9

Subdomain Enumeration and Analysis	
Awesome & Beautiful Artworks	https://darknetdiaries.com/artwork/
Subdomain Finder Tools (Sublister)	https://github.com/aboul3la/Sublist3r
Subdomain Finder Tools (Subfinder)	https://github.com/projectdiscovery/subfinder
Subdomain Finder Tools (Sublister)	https://github.com/aboul3la/Sublist3r
Reconnaissance, Tactic TA0043 - Enterprise MITRE ATT&CK®	https://attack.mitre.org/tactics/TA0043/
Application Testing Methodology & Scope Based Recon (Harsh Bothra)	https://speakerdeck.com/harshbothra/application-testing-methodology-and-scope-based-recon
Turning your time into bugs — zseano's thoughts	https://zseano.medium.com/turning-your-time-into-bugs-zseanos-thoughts-22a2ca2c8ef5
Weaponizing Recon - Smashing Applications for Security Vulnerabilities & Profit (Harsh Bothra)	https://speakerdeck.com/harshbothra/weaponizing-recon-smashing-applications-for-security-vulnerabilities-and-profit
Mechanizing the Methodology: by Daniel Miessler	https://danielmiessler.com/blog/mechanizing-the-methodology/