1. Read the program scope:

Yes, it appears to be obvious thing to do… but always double-check the program rules. This initial examination is helpful in learning a lot about the target:

→What is its scope?

→What should I avoid testing?

→What kinds of vulnerabilities and reports are accepted?

→What flaws have already been reported? (If, for instance, public disclosure on HackerOne)

→Rewards?

The function of URLs etc., can be readily understood by looking at a program's rules. Reading everything at once saves time because you won't have to focus on a domain or program that isn't relevant.

*I usually hunt on wildcard domains like *.domain.com rather than main web app domain like www.domain.com !!*

2. Enumeration

Assetfinder — It uses multiple sources like certificate transparency, Facebook, Virustotal, etc. It works out of the box, but if you want more results, you can configure the API keys for the services which need one.

Provided that you have installed and configured Go, the command is simple, you just have to pipe your target to the tool.

```
echo domain.com | assetfinder --subs-only
```

OWASP Amass — It supports passive and active enumeration, performs DNS resolution and can also brute-force the subdomains based on the wordlist of your choice. The user guide is detailed and gives example commands that you can run. The simplest and quickest subdomain enumeration command would be:

```
amass enum -d domain.com -passive
```

Sublist3r is a python tool designed to enumerate subdomains of websites using OSINT. It helps penetration testers and bug hunters collect and gather subdomains for the domain they are targeting. Sublist3r enumerates subdomains using many search engines such as Google, Yahoo, Bing, Baidu, and Ask. Sublist3r also enumerates subdomains using Netcraft, Virustotal, ThreatCrowd, DNSdumpster, and ReverseDNS.

```
python sublist3r.py -d domain.com
```

AQUATONE — It is a set of tools for performing reconnaissance on domain names. It can discover subdomains on a given domain by using open sources as well as the more common subdomain dictionary brute force approach. After subdomain discovery, AQUATONE can then scan the hosts for common web ports and HTTP headers, HTML bodies and screenshots can be gathered and consolidated into a report for easy analysis of the attack surface.

> *Subdomain Takeover: You should check "Can I take over XYZ". It's a GitHub repository created by Ed Overflow and you will know if it's possible to takeover a subdomain used by a service (GitHub pages, Heroku, CloudFront, etc.)*
>
> *It has most of the service covered if it can be taken over or not.*

3. Tech Stack Used

Knowing what technologies your target uses to function is crucial before you start searching for vulnerabilities:

Are WAFs like CloudFront or CloudFlare used by them?

Do they use a CMS like Wordpress, Drupal or Joomla ?

Use frameworks like CakePHP or AngularJS among others?

What version of Apache is this?

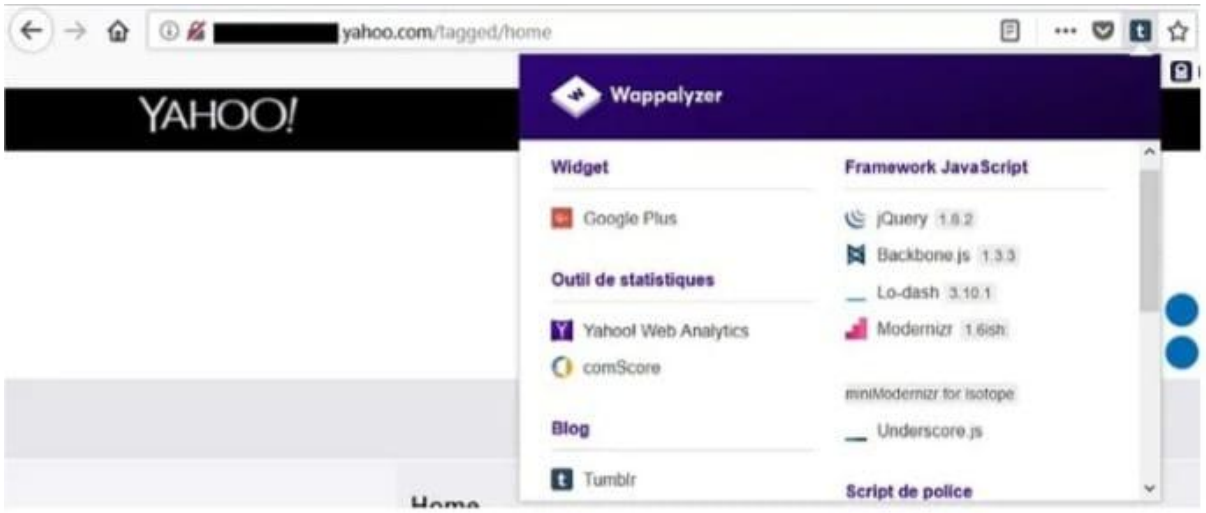
Do they using a template engine like Smarty or Jinja2?

Knowing which technology they employ will be helpful in assisting you in your search for weaknesses, and it will be preferable if you had the version.

For example : Your target have a profil page where you can control input (like name or nickname) which is reflected on a public page. If you know the framework version, you can possibly try XSS PayLoads or others payloads!

If your target use a template engine like Smarty, check the version and try template injection!

To get these information about a target, I installed a plugin called Wappalyzer.



This plugin works on Google Chrome and Firefox. And to use it, you just need to go on your target. You will see a small icon displayed next to the URL.

If you click on it, Wappaylzer should display interesting information about your target. Sometimes you will not see the version, but it can be useful anyway.

4. Dorks

1. *Shodan : With its ever-growing database and ease of use, Shodan has become one of the most popular tools used by security researchers for gathering IoT intelligence.*

Shodan provides a great starting point for researchers performing any information gathering task. By being able to filter data by its location, software version, when it

was last seen and much more, Shodan can help researchers target specific research points, making their work easier and more efficient.

Dork list: https://github.com/IFLinfosec/shodan-dorks

2. Google : Google helps you to find Vulnerable Websites that Indexed in Google Search Results. Here is the latest collection of Google Dorks. A collection of 13.760 Dorks ..!

Dork List: https://github.com/BullsEye0/google_dork_list

3. Github: Github Search is a quite powerful and useful feature that can be used to search for sensitive data on repositories. Collection of Github dorks can reveal sensitive personal and/or organizational information such as private keys, credentials, authentication tokens, etc. This list is supposed to be useful for assessing security and performing pen-testing of systems.

Dork List: https://github.com/techgaun/github-dorks

5. Perfect Wordlist and Directory Fuzzing

My experience has allowed me to understand that targets always have a page that should not be public, should not be here or should not be accessible without permission.

To help me discover these "secret" pages, I recommend this excellent GitHub repository where awesome guys share their lists and knowledge.

SecLists is the security tester's companion. It's a collection of multiple types of lists used during security assessments, collected in one place. List types include usernames, passwords, URLs, sensitive data patterns, fuzzing payloads, web shells, and many more. The goal is to enable a security tester to pull this repo onto a new testing box and have access to every type of list that may be needed.

To use the SecList, I recommand you to use one of these tools:

- *Dirsearch : https://github.com/maurosoria/dirsearch*
- *Dirb : Installed by default on Kali Linux*
- *FFUF : sudo apt install ffuf*

You can specify a list of words you want to use and an extension file or extension list. The tool will test one by one all entries and you will see the result directly in the terminal.

Author - Himangshu Sarkar
Github - https://github.com/Himangshu30