**Walkthrough for my recon methodology**

Hello amazing hackers. Today I am going to walk you through my methodology from start to finish and explain what I am looking for along the way. Along with this walkthrough will be a infographic to show the steps clearly and concisely.

With this example lets say we have the target \*.google.com

The first thing I do with any target and list is make a file and folder. I will pull up my cli on kali and type in mkdir google then I will go into that directory and make a file called domains.txt

Commands look like nano domains.txt then enter in your domain without the wildcard. It will look like this

google.com

Now that I have this set up I can pass it to my tools and brute-forcers. The tools I use are subfinder, amass, httprobe, anew, gau, hakrawler, and ffuf. I run the amass command with the domain itself like this.”Amass enum -d google.com | tee subs.txt” I do that to run the urls into a file that I can check and pass through other tools. Subfinder is next it can run with your file you have made like this: “subfinder -dL domains.txt -o subs1.txt” with these two processes you can combine the two subdomain lists with anew like this: cat subs1.txt | anew subs.txt this tool called anew is from tomnomnom and allows you to add newlines that don’t exist from one file to the other.

Now we have a complete list of all of the subdomains in subs.txt and we can use them to get the responses and enumerate the ip addresses for open ports aswell as check the urls that have been crawled in the past. So lets display that, with our subs.txt file ready we will use these commands and get a lot of the data we need.

Cat subs.txt | httprobe | tee alive.txt

Namp -iL subs.txt -sC -sT -sV -oN nmapres.txt

Cat alive.txt | gau –blacklist png,jpg,jpeg,svg,img,woff1,woff2,woff3,eot,css | tee gaures.txt

Cat gaures.txt | httpx -mc 200,403 | tee gaulive.txt

Cat alive.txt | hakrawler | grep “.js$” | tee js.txt

Ffuf -u FUZZ.google.com -w /tools/wordlists/Seclists/Discovery/Web-Content/DNS/jhaddizsubs.txt | tee brute.txt

With all of these commands we will have all of the data we need to start with, We have javascript related to every subdomain, we have all of the ports that are open on the subdomains and we have any URLs that may have been crawled before in the history of the application.

What do I do with all of this data? I use the nmap scans to find subdomains with more than 3 open ports and pick those subdomains to enumerates further. I use their urls to find the javascripts and see if gau found anything with those subdomains in it.

After filtering the data to the subdomains that I found were interesting from the nmap scans I brute force the directories with ffuf like this: ffuf -u sub.google.com/FUZZ -w seclists/directories/web-content/common.txt or any other wordlist you want to use for directories. I will also go through the javascript files for directories or api endpoints. If youd like to know how I go through javascript please refer to the javascript walkthrough.

Now that I have all of the directories and files that I can find without an authenticated header, I use burpsuite and start visiting these sites manually. I lookup docs for the site and try to abuse the functions to find more attack surface.

Once I have recorded all of the pages I found in burpsuite, I start looking through the security headers for more connections and urls remember the headers have destinations too that communicate with your targets.

Now I have a full picture of the functions and routes of the subdomains and I need to bring it all the way back to my \*.google.com target right? So I will trace any calls with burpsuite from the main page that goto the domains that I have found as well as tracking my subdomains and their connection to my main target. For example if sub.google.com sends an OAUTH check to google.com or even sub2.google.com I will check if those are controllable if they are then I enumerate further.

With all of this data in hand I’m ready to manually hunt and check results.

This is my recon methodology I hope you enjoyed and can deploy it for yourself.