Broad scope methodology - Manual

Introduction

Before we start running our tools, we need to know what they do in my opinion. Just running a script and expecting magic to happen is script kiddie behavior and we are far beyond that my friends. We are hackers.

I am not saying you should never use tools, mind that. Tools are very useful in automating our workflows but they miss so much. A tool only checks what you tell it to check while human eyes are unbeatable in detecting details that are odd or off. That's what it's all about my friends, we need to mind those details very much. We can't ignore them.

The other reason i always recommend doing manual recon is that you can not possible program every single scenario into your automation. Life is diverse and so is software. We can't rely on automation to find all the bugs. That being said, i am a HEAVY proponent of nuclei from project discovery but not the default templates. More on that later.

Automation is good but automation combined with manual testing ensures we get the best results possible. We also need to know the processes to improve our automation because for me it's very important to keep improving. I don't know about you friend but i always have new ideas on how i can improve my workflow. If i can implement those into my test automation, i have a big advantage since i won't have to manually retest a target every time.

Test objectives

We want to achieve the following test objects with our manual recon:

- Find an asset that is suiteable for our attack strategy
- Explore our target
- Execute our attack strategy

This may seem simple but there is a lot of hidden truth in these simple words. First and foremost, we want to find an asset to execute our attack strategy on. This means that we need an attack strategy first. That's the exact reason we started with single scope applications in our course. We want to build a solid strategy before we even begin thinking about recon, how else would we even recognize a suiteable target if we saw it?

We need to explore the assets we found thouroughly, we can either do this manually or automatically but both aim at different vulnerabilities.

- The manual approach is actually semi automated as we've seen in the main app chapter but in general we need to explore our target's assets very thoroughly to actually find vulnerabilities.
- The automated approach aims at things like CVE's or misconfigurations.

Whichever approach we pick, we need to be aware that simply running a tool is not going to be enough.

Disadvantages of tools

I'm going to list the disadvantages of tools but again, i don't recommend against them. I recommend you first learn how to do it yourself.

- Everyone can run the same tool. It's not hard, there's usually instructions with the tool and it's free online. If everyone runs the same tool, everyone gets the same dupes.

- If you don't know what your tool does you are missing out on a lot of stuff
- Running a tool without knowing what it does is stupid. I'm not going to sugar coat it, it's just plain stupid. You could be testing very intrusively without even realising it and your target might not like that.

How can we do manual recon?

Google dorking is very important in this process for me. Google indexes websites to a stunning degree and we query those websites if we use the correct syntax. In the following scenario i will be investigating Tesla as they have an excellent bug bounty program. I am making sure i am not showing you guys any vulnerabilities and if i would find them, believe me i would report them myself 😂. This being said, today we will explore how to find a target, not a bug. That will be taken up in other chapters.

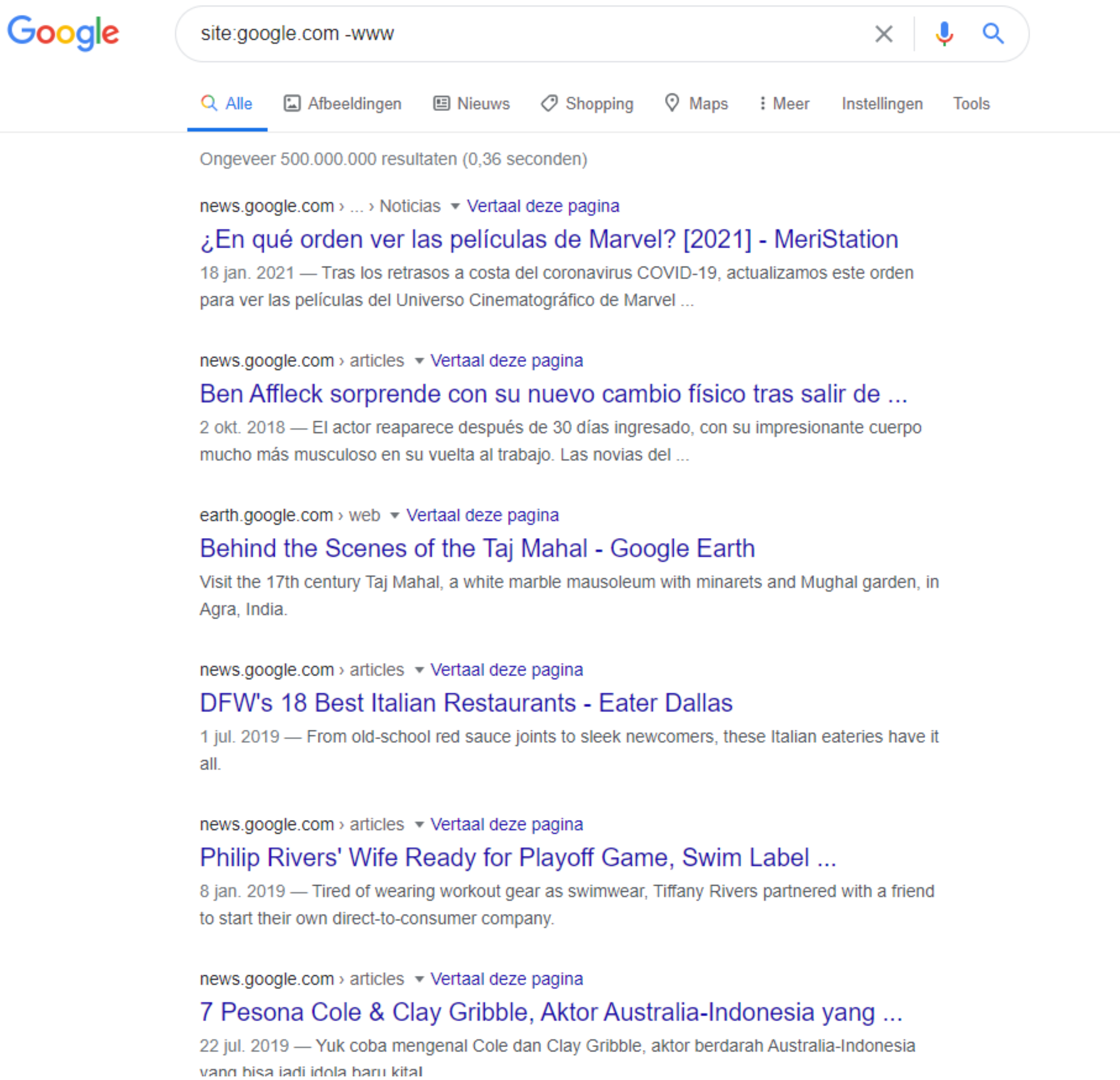
Google dorking

We will start with some good old google dorking. This is the first step in our process and we will try to emulate "subdomain" enumaration using this method.

We will start with a basic dork

```
site:google.com -www
```

This will show us all the subdomains from google except the main www subdomain.



site:google.com -www

Ongeveer 500.000.000 resultaten (0,36 seconden)

news.google.com › ... › Noticias ▾ Vertaal deze pagina
¿En qué orden ver las películas de Marvel? [2021] - MeriStation
18 jan. 2021 — Tras los retrasos a costa del coronavirus COVID-19, actualizamos este orden para ver las películas del Universo Cinematográfico de Marvel ...

news.google.com › articles ▾ Vertaal deze pagina
Ben Affleck sorprende con su nuevo cambio físico tras salir de ...
2 okt. 2018 — El actor reaparece después de 30 días ingresado, con su impresionante cuerpo mucho más musculoso en su vuelta al trabajo. Las novias del ...

earth.google.com › web ▾ Vertaal deze pagina
Behind the Scenes of the Taj Mahal - Google Earth
Visit the 17th century Taj Mahal, a white marble mausoleum with minarets and Mughal garden, in Agra, India.

news.google.com › articles ▾ Vertaal deze pagina
DFW's 18 Best Italian Restaurants - Eater Dallas
1 jul. 2019 — From old-school red sauce joints to sleek newcomers, these Italian eateries have it all.

news.google.com › articles ▾ Vertaal deze pagina
Philip Rivers' Wife Ready for Playoff Game, Swim Label ...
8 jan. 2019 — Tired of wearing workout gear as swimwear, Tiffany Rivers partnered with a friend to start their own direct-to-consumer company.

news.google.com › articles ▾ Vertaal deze pagina
7 Pesona Cole & Clay Gribble, Aktor Australia-Indonesia yang ...
22 jul. 2019 — Yuk coba mengenal Cole dan Clay Gribble, aktor berdarah Australia-Indonesia yang bisa jadi idola baru kita!

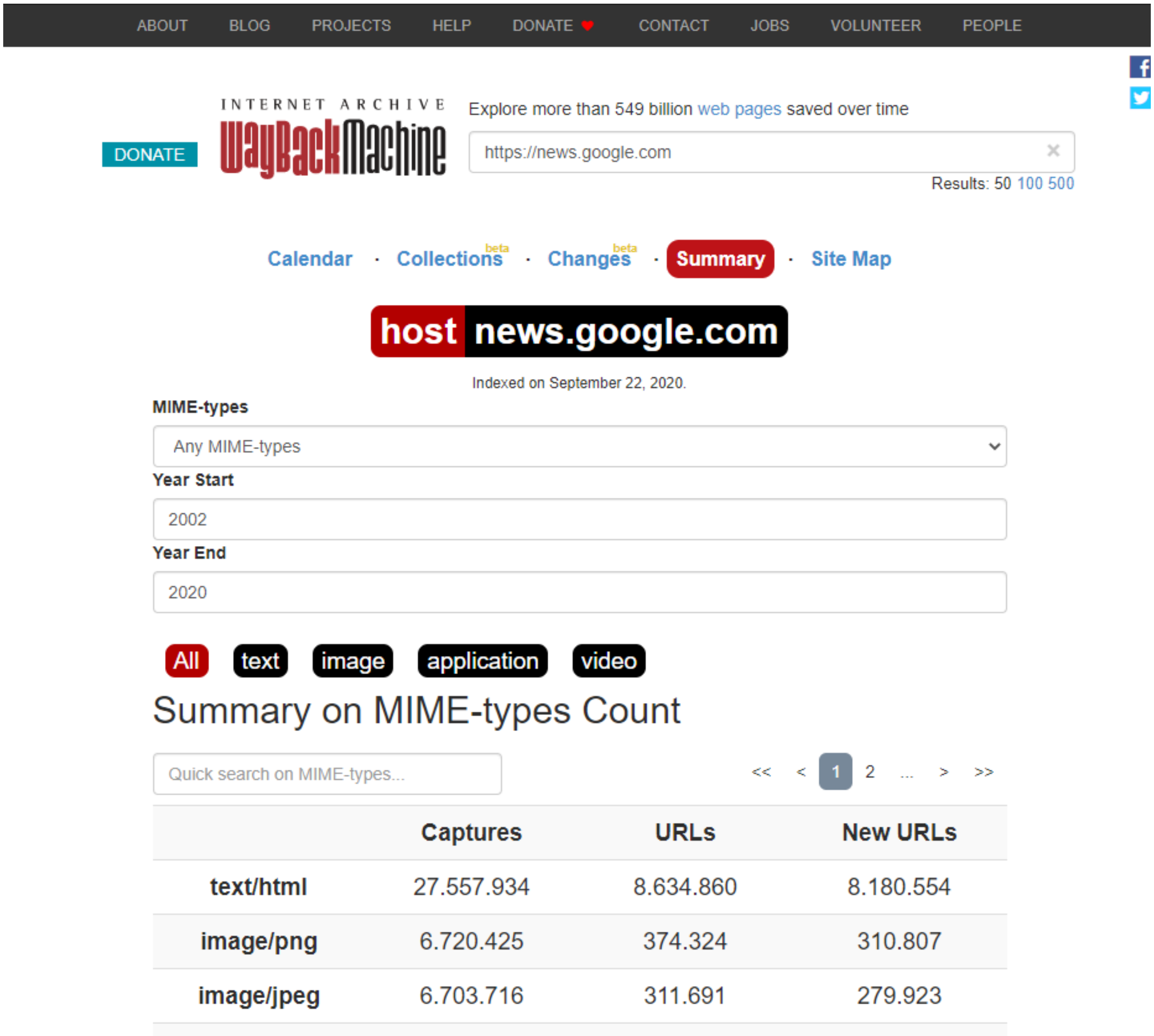
I will then take on the first subdomain that i see. This is where the testing begins. I explore the subdomain and try to find out what functionality is available via the website. I even go as far as to make a mindmap of all of the functionality so i'm sure i will remember later on.

After exploring the subdomain i will go to waybackmachine to find more hidden functionality and even read the javascript file if i like the functionality.

Waybackmachine

We will use this website to emulate waybackurls. Waybackurls will grab all the URI's of a certain subdomain from waybackmachine but the waybackmachine has some pretty awesome functionality for us.

http://web.archive.org/



Explore more than 549 billion web pages saved over time

https://news.google.com

Results: 50 100 500

host news.google.com

Indexed on September 22, 2020.

MIME-types: Any MIME-types

Year Start: 2002

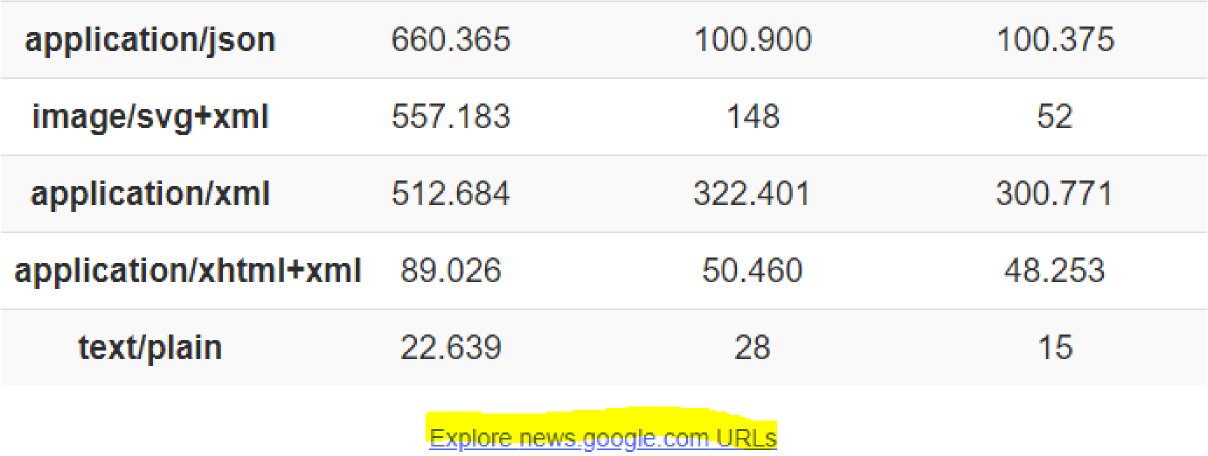
Year End: 2020

Summary on MIME-types Count

	Captures	URLs	New URLs
text/html	27.557.934	8.634.860	8.180.554
image/png	6.720.425	374.324	310.807
image/jpeg	6.703.716	311.691	279.923

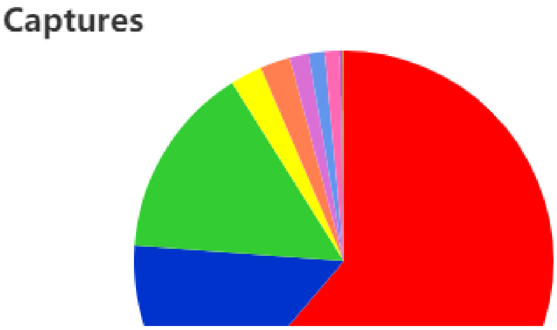
I really like the summary tab. In here you will find a distribution by mimetype, if we only see images it might not be interesting to investigate if those images are public for example.

We can also explore all the URLs of our asset.

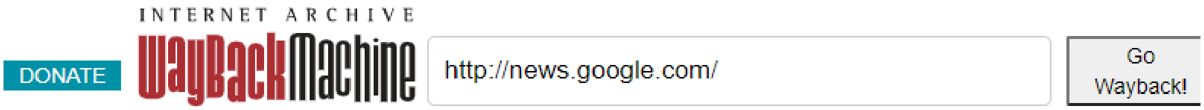


application/json	660.365	100.900	100.375
image/svg+xml	557.183	148	52
application/xml	512.684	322.401	300.771
application/xhtml+xml	89.026	50.460	48.253
text/plain	22.639	28	15

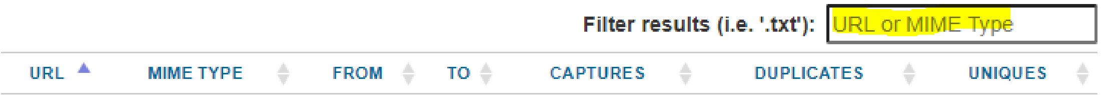
Explore news.google.com URLs



Here we can filter by URL or by MIME type



http://news.google.com/



Filter results (i.e. '.txt'): URL or MIME Type

URL MIME TYPE FROM TO CAPTURES DUPLICATES UNIQUES

I also really like the site map as it shows all the URLs in a handy pie chart which allows us to explore the site in a birds-eye view.

Moving on

If we are done with our target we can explore our next target. To get there, i simply remove my current asset from my search results by adepting the google DORK.

```
site:google.com -www -news
```

Other options

We can also use yahoo, duck duck go, bing, ... to explore our target more. This is basically what automated tools do. They look at as many sources as possible and try to gather all the subdomans.

Show me your secrets

When i've seen all the subdomains i want to see, i move on to trying to find some more secrets. Github is perfect for that, sometimes there's things like API keys, secrets or even login data hidden in a repo.

https://securityonline.info/github-dorks/