External Attack Surface

from mapping to monitoring

Alessio Petracca

$ whoami

Alessio Petracca

computer engineer working in the information security field focused on application and infrastructure security both offensive and defensive.

@alessiopetracca

Why this talk?

sharing a practical approach to the discovery, the mapping and the monitoring of the external attack surface of a company

https://flic.kr/p/Mk3ogQ

What is the external attack surface?

the entire set of services
exposed to the Internet
that can potentially be exploited
by a malicious user

https://flic.kr/p/dh9rQt

Context

- Security Manager: that must challenge the internal security teams
- Blue Teamer: that should put in place effective monitoring and defense mechanism
- Red Teamer: that is conducting a wide scope assessment
- Bug bounty hunter: that is enjoying a program with a company-wildcard scope

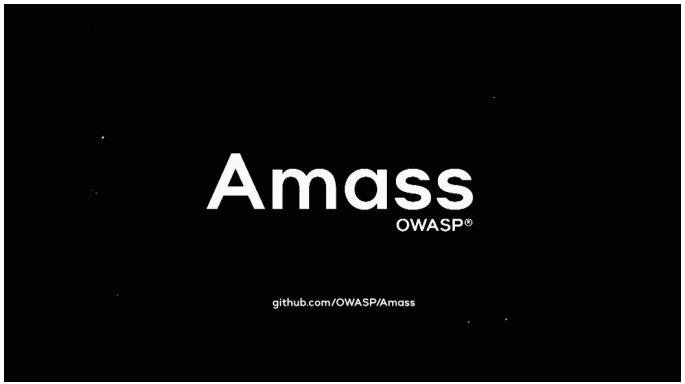
You need to know the company's external facing assets!

Recon

The research and identification of targets

OWASP Amass

“The OWASP Amass Project performs network mapping of attack surfaces and external asset discovery using open source information gathering and active reconnaissance techniques.”



- originally written by Jeff Foley (@caffix)
- adopted by OWASP Foundation in March 2018

https://github.com/OWASP/Amass

Root Domains

To help us in discovering as many **root domains** related to the organization as possible, we will leverage on:

It is usually a good practice the study of mergers and acquisitions of the company.

Root Domains - ASNs enumerations

An autonomous system (AS) is a large network or group of networks with a single routing policy. It has control over a specific set of IP addresses and it is typically operated by a single large organization.
Each AS is assigned a unique ASN by the Internet Assigned Numbers Authority (IANA).

Note: even after this accurate manual analysis, we are missing a significant part of the attack surface due to the presence of assets on third parties and cloud environments.

- https://bgp.he.net/
- https://bgpview.io/
- https://hackertarget.com/as-ip-lookup/
- https://github.com/OWASP/Amass
- https://github.com/j3ssie/metabigor
- https://github.com/yassineaboukir/Asnlookup

Root Domains - ASNs enumerations

```
~ > amass intel -org "tesla"
8911, TESLATEL-AS
50313, TESLATEL-AS
394161, TESLA - Tesla
```

```
~ > amass intel -active -asn 394161
zip.zayo.com
teslamotors.com
solarcity.com
tesla.com
mysolarcity.com
```


```
~ > amass intel -active -cidr 199.66.11.0/24
teslamotors.com
tesla.com
solarcity.com
```

Root Domains - Reverse Whois lookups

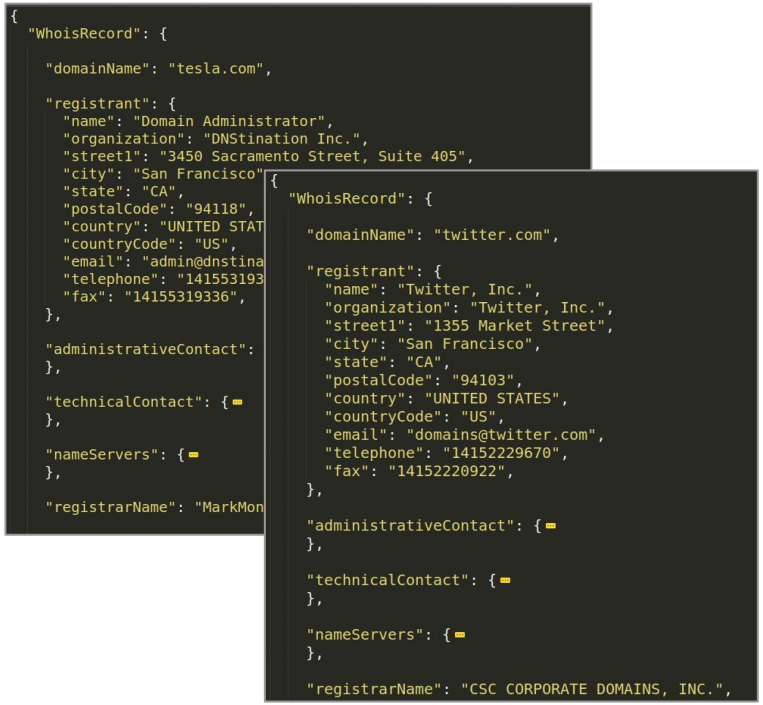
The Reverse Whois is a technique that allows to find root domains registered using the same details of the in scope domain in the Whois records (e.g. organization, email, ...).

Note: if the target domain has “REDACTED FOR PRIVACY” or “hidden” fields this method does not return any result.

- https://viewdns.info/reversewhois/
- https://www.reversewhois.io/
- https://www.whoxy.com/
- https://reversewhois.domaintools.com/
- https://tools.whoisxmlapi.com/reverse-whois-search
- https://github.com/OWASP/Amass
- https://github.com/vysecurity/DomLink

https://flic.kr/p/dv1gNz

Root Domains - Reverse Whois lookups

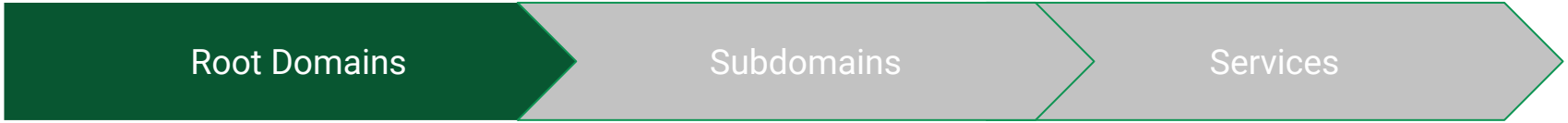



```
~ > amass intel -whois -d tesla.com
cars-tesla.com
hailing-tesla.com
it-tesla.com
sa-tesla.com
show-tesla.com
art-tesla.com
baterias-tesla.com
hail-tesla.com
na-tesla.com
ride-hailing-tesla.com
-tesla.com
```

```
~ > amass intel -whois -d twitter.com
periscopecast.com
as13414.net
hyperlinks.us
periscoep.com
periscoepe.com
```

amass recursively intel techniques

$ amass intel -active -asn 394161 -whois -d tesla.com



- ASNs
- Reverse Whois

Manual approach is suggested to validate all the root domains returned.

Subdomains

In order to find as many **subdomains** as possible, we will leverage on the following techniques:

subdomain scraping

subdomain brute-forcing

Subdomains scraping

Subdomains can be exposed:

- in Certificate Transparency Logs
- Web Archives
- Social media
- Text storage hosting service
- HTTP Headers
- ...

Note: there are many reliable tools that scrape these data sources and aggregate results quite well! Get out the best of the data sources by using their API keys.

- https://github.com/OWASP/Amass
- https://github.com/projectdiscovery/subfinder

Subdomains brute-forcing

Even more subdomains can be discovered by just “guessing” their names:

- based on wordlist
- based on alterations/permutations
- based on patterns (mask attack)

Note: to speed-up the process it is suggested to use multiple DNS resolvers at the same time.

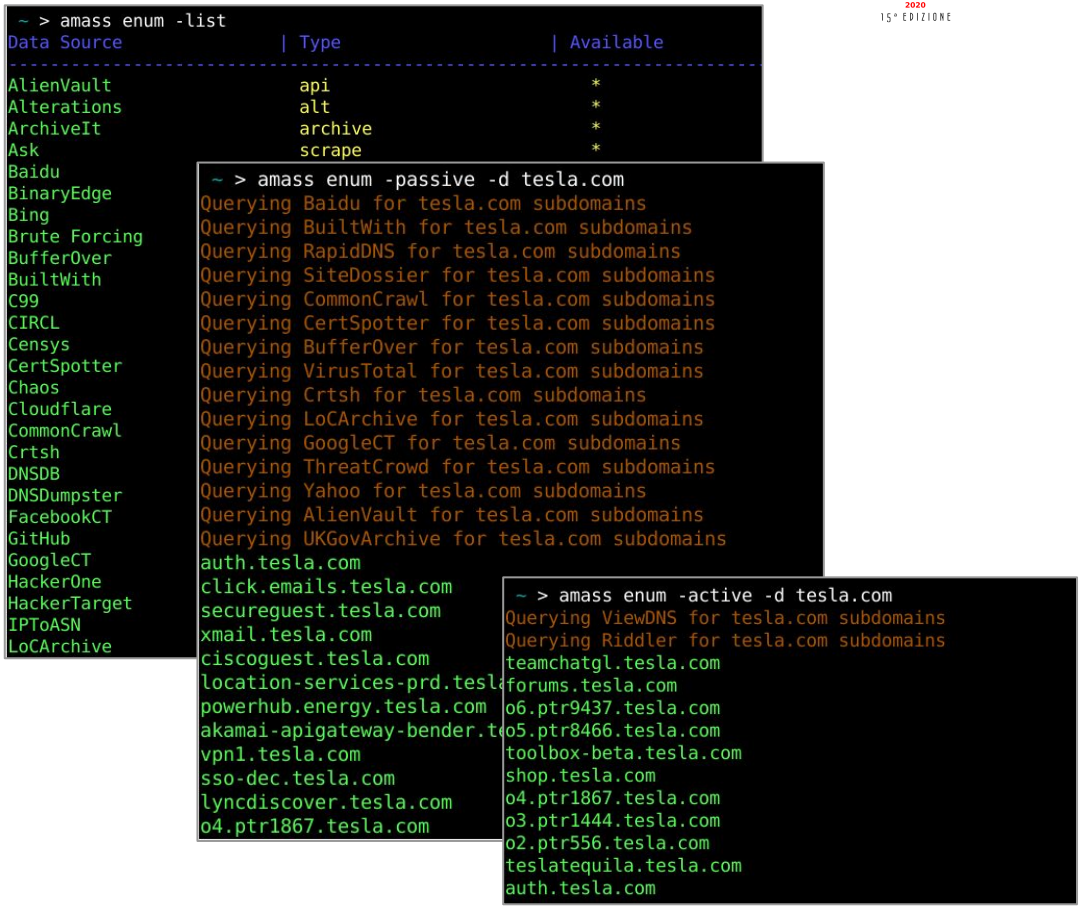
- https://github.com/OWASP/Amass
- https://github.com/projectdiscovery/shuffledns

Subdomains scraping

$ amass enum -list

$ amass enum -passive -d domain.com

$ amass enum -active -d domain.com



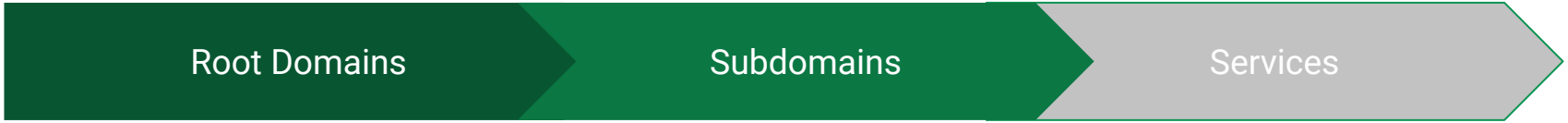
Subdomains brute-forcing

$ amass enum -active -brute -src -rf resolv.txt -d tesla.com
# brute-forcing on wordlists + alterations

$ amass enum -active -brute -src -asn 394161 -d tesla.com
# added ASN context and get better results

$ amass enum -active -brute -src -asn 394161 -rf 20resolvers.txt
-max-dns-queries 10000 -d tesla.com
improve speed and reduce risk of being rate limited

```
[ArchiveIt]       forums.tesla.com
[Brute Forcing]   3.tesla.com
[ThreatCrowd]     akamai-apigateway-fta.tesla.com
[Baidu]           edr.tesla.com
[CertSpotter]     teslatequila.tesla.com
[VirusTotal]      ownership.tesla.com
[ThreatCrowd]     akamai-apigateway-payment.tesla.com
[VirusTotal]      static-assets-pay.tesla.com
[AlienVault]      image.emails.tesla.com
[ArchiveIt]       auth.tesla.com
[RapidDNS]        autobidder.powerhub.energy.tesla.com
[ThreatCrowd]     schedule.tesla.com
[ArchiveIt]       shop.tesla.com
[AlienVault]      click.email.tesla.com
[ThreatCrowd]     akamai-apigateway-logisticsratesapi.tesla.com
[Brute Forcing]   autodiscover.tesla.com
[AlienVault]      gridlogic.energy.tesla.com
[AlienVault]      mfa.tesla.com
[AlienVault]      static-assets.tesla.com
[Brute Forcing]   monitoring.tesla.com
[AlienVault]      sso-dev.tesla.com
[Brute Forcing]   link.tesla.com
[AlienVault]      sip.tesla.com
[Riddler]         image.email.tesla.com
[Brute Forcing]   bi.tesla.com
[DNSDumpster]     mta2.email.tesla.com
[Brute Forcing]   feedback.tesla.com
```



- ASNs
- Reverse Whois

- scraping
- brute-forcing

Manual approach is suggested to validate all the root domains returned.

This part can be automated.

Services

To identify all the services exposed by the organization we're going to perform:

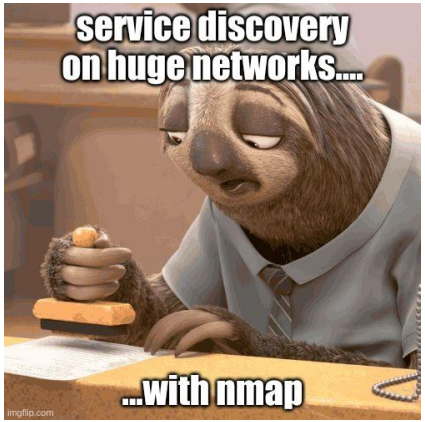
service discovery

service analysis

Service discovery

First thing that comes to our mind is to execute nmap on all domains/subdomains revealed so far. But this may raise two problems:

- nmap is very slow
- some subdomains could be hosted on a third party infrastructure that may be not in the scope



The solution is to use **masscan**, for its speed in finding open ports and to take at our advantage its limitation to scan only ranges of IPs.

masscan -iL ipranges.txt --rate 10000 --top-ports 1000 -oG massoutput.txt (https://github.com/robertdavidgraham/masscan)

```
Discovered open port 443/tcp on 199.66.[redacted]
Discovered open port 443/tcp on 199.66.[redacted]
Discovered open port 443/tcp on 199.66.[redacted]
Discovered open port 443/tcp on 199.66.[redacted]
Discovered open port 443/tcp on 199.66.[redacted]
Discovered open port 443/tcp on 199.66.[redacted]
Discovered open port 443/tcp on 199.66.[redacted]
Discovered open port 443/tcp on 199.66.[redacted]
Discovered open port 80/tcp on 199.66.[redacted]
Discovered open port 80/tcp on 199.66.[redacted]
rate:  1.00-kpps, 64.30% done,   0:00:09 remaining, found=24
```

Service analysis

After the masscan execution, we have a precise list live IPs and open ports:

```
grep Host: massoutput.txt | cut -d " " -f3 | sort -V | uniq > iplist.txt
grep Ports: massoutput.txt | cut -d " " -f5 | cut -d "/" -f1 | sort -n | uniq | paste -sd, > portlist.txt
```

Now, we can leverage the **nmap** accuracy to find information on exposed services running on these ports by grabbing versions and banners:

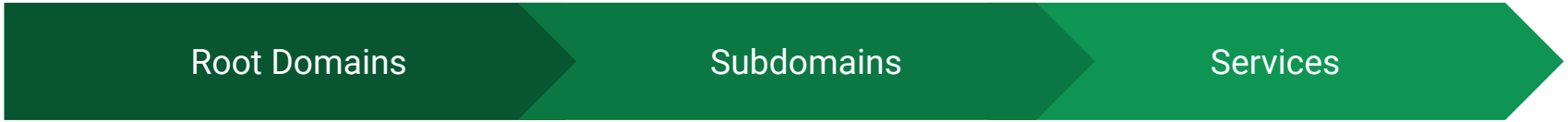
```
sudo nmap -sS -sV -p $(portlist.txt) -v -open -Pn -n --randomize-hosts -iL iplist.txt -oA nmap_output
```

Note: additional ports are usually detected by nmap (among those included in the portlist.txt)

Web Applications/Services analysis

The web services are fruitful of information that we can extract in order to expand our scope and also validate what we previous detected:

- favicon hashes correlation in combination with shodan.io (https://github.com/devanshbatham/FavFreak)
- tracking codes correlation (Google Analytics, Adobe Analytics, Facebook Pixel Tag, AddThis Tag, NewRelic Tag, ...) with https://builtwith.com/
- google dorking using copyright text or privacy text with inurl:company



- ASNs
- Reverse Whois

- scraping
- brute-forcing

- service discovery
- service analysis

Manual approach is suggested to validate all the root domains returned.

This part can be automated.

This part can be automated.

Tracking and Monitoring

The workflow explained above must be completed by an appropriate monitoring in order to promptly react to every changes that will occur.

tracking subdomains

$ amass track -d tesla.com

```
Found: bi.tesla.com 104.118.214.137
Moved: os.tesla.com
         from   95.101.200.75
         to     92.123.185.250
Moved: click.email.tesla.com
         from   23.203.249.40,2.21.33.131
         to     72.247.179.40,72.247.179.43
Removed: email.tesla.com
```

monitoring services

Integration with the ELK stack with the watcher feature

Takeaways

- Today we talked about the reconnaissance phase that is very useful from the external perspective but also from the internal perspective of an organization.
- There are many tools that can help in this activity, but our ability is to get the best from each of them find the right trade-off between speed and accuracy and with a minimum effort we'll see great results.
- Don't be afraid to develop your own tools!

Here you can find the main references:

- https://github.com/jhaddix/tbhm
- https://github.com/OWASP/Amass

Thank you

alessio.petracca@gmail.com
@alessiopetracca