Open Source Intelligence Gathering 201

Archived security papers and articles in various languages.

EDB-ID: 45360	**Author**: appsecco (https://www.exploit-db.com/author/?a=9669)	**Published**: 2018-09-06
Type: Papers (https://www.exploit-db.com/papers/)	**Platform**: Multiple (https://www.exploit-db.com/platform/?p=Multiple)	**Language**: English (https://www.exploit-db.com/papers/?l=1)
Advisory/Source: Link (https://blog.appsecco.com/open-source-intelligence-gathering-201-covering-12-additional-techniques-b76417b5a544)	**Paper**: Download (https://www.exploit-db.com/download/45360.md) / View Raw (https://www.exploit-db.com/raw/45360/)	

« Previous Paper (https://www.exploit-db.com/papers/45359/)
Next Paper » (https://www.exploit-db.com/docs/english/45362-hypervisor-from-scratch---part-1-basic-concepts-&-configure-testing-environment.pdf)

Open Source Intelligence Gathering 201 (Covering 12 additional techniques)
===

This post is the second in a series of technical posts we are writing about **Open Source Intelligence** (**OSINT**) gathering_._

We highly recommend that you follow the series in a sequence.

1. [Open Source Intelligence Gathering 101](https://blog.appsecco.com/open-source-intelligence-gathering-101-d2861d4429e3)
2. **You are reading this**
3. More to come

There is various kinds of data that can be categorised as OSINT data but all of this data is not of significance from a penetration tester point of view. As a penetration tester, we are more or less interested in the information that will fall under following categories —

1. Information that'll increase the attack surface (domains, net blocks etc)
2. Credentials (email addresses, usernames, passwords, API keys etc)
3. Sensitive information (Customer details, financial reports etc)
4. Infrastructure details (Technology stack, hardware equipment used etc)

> Open Source Intelligence (OSINT) is data collected from publicly available sources.

### 12 additional techniques for doing OSINT

1.SSL/TLS certificates have a wealth of information that is of significance during security assessments.

> _An SSL/TLS certificate usually contains domain names, sub-domain names and email addresses. This makes them a treasure trove of information for attackers._

Certificate Transparency(CT) is a project under which a Certificate Authority(CA) has to publish every SSL/TLS certificate they issue to a public log. Almost every major CA out there logs every SSL/TLS certificate they issue in a CT log. These logs are available [publicly](https://www.certificate-transparency.org/known-logs) and anyone can look through these logs. We wrote a script to extract subdomains from SSL/TLS certificates found in CT logs for a given domain. You can find the script here —

[**appsecco/the-art-of-subdomain-enumeration**
_This repository contains all the supplement material for the book "The art of sub-domain enumeration" …
_git.io](https://git.io/fAGW0 "https://git.io/fAGW0")[](https://git.io/fAGW0)

![](https://cdn-images-1.medium.com/max/1600/1*cCNRAHNHmw7RDY37F1Ivqw.png)
Extracting subdomains from SSL/TLS certificates listed in CT logs

SSLScrape is a tool that will take a netblock(CIDR) as input, queries each IP address for SSL/TLS certificates and extracts hostnames from SSL certificates that are returned. The tool is available here —

[**cheetz/sslScrape**
_SSLScrape | A scanning tool for scaping hostnames from SSL certificates. - cheetz/sslScrape_github.com](https://github.com/cheetz/sslScrape "https://github.com/cheetz/sslScrape")[](https://github.com/cheetz/sslScrape)

```
sudo python sslScrape.py TARGET_CIDR
```

![](https://cdn-images-1.medium.com/max/1600/1*dOF94BIsWhPmnHpjIW0pGA.png)
sslScrape extracting hostnames from SSL/TLS certificates returned by IPv4 hosts

2.WHOIS service is generally used during a penetration test to query information related to registered users of an Internet resource, such as a domain name or an IP address (block). WHOIS enumeration is especially effective against target organisations that have large presence on the Internet.

> Some public WHOIS servers support advanced queries that we can use to gather wide range of information on a target organisation.

Let's look at some advanced WHOIS queries to gather information —

*   We can query [ARIN WHOIS server](https://www.arin.net/resources/services/whois_guide.html) to return all the entries that has email address of a given domain name, which in this case is _icann.org._ We are extracting only the email addresses from the results.

```
whois -h whois.arin.net "e @ icann.org" | grep -E -o "\b[a-zA-Z0-9.-]+@[a-zA-Z0–9.-]+\.[a-zA-Z0–9.-]+\b" | uniq
```

![](https://cdn-images-1.medium.com/max/1600/1*oYGnT18lLtml0WUfGpACpA.png)
Extracting email addresses from WHOIS by querying entries that contain email address of a specific domain

*   We can query [RADB WHOIS server](http://www.radb.net/support/query1.php) to return all the netblocks that belong to an Autonomous System Number(ASN)

```
whois -h whois.radb.net -- '-i origin AS111111' | grep -Eo "([0-9.]+){4}/[0-9]+" | uniq
```

![](https://cdn-images-1.medium.com/max/1600/1*iXy-zyHi-VGtx80ysAD93w.png)
Listing all the netblocks under an ASN using a query against RADB WHOIS server

*   We can query ARIN WHOIS server to return all POC, ASN, organizations, and end user customers for a given keyword.

```
whois -h whois.arin.net "z wikimedia"
```

![](https://cdn-images-1.medium.com/max/1600/1*TZfmfe-oIlv3vcG-62S1dw.png)
Finding out information regarding an organisation using WHOIS service

3. Finding [Autonomous System (AS) Numbers](https://www.iana.org/assignments/as-numbers) will help us identify netblocks belonging to an organisation which in-turn may lead to discovering services running on the hosts in the netblock.

*   Resolve the IP address of a given domain using `dig` or `host`

```
dig +short google.com
```

*   There are tools to find ASN for a given IP address

```
curl -s http://ip-api.com/json/IP_ADDRESS | jq -r .as
```

We can use WHOIS service or NSE scripts to identify all the netblocks that belong to the ASN number.

```
nmap --script targets-asn --script-args targets-asn.asn=15169
```

![](https://cdn-images-1.medium.com/max/1600/1*z6W4_GxLJsya69L9cusFXg.png)
Finding netblocks that belong to an ASN using targets-asn NSE script

4. Usage of Cloud storage has become common especially object/block storage services like Amazon S3, DigitalOcean Spaces and Azure Blob Storage. In last couple of years, there have been [high profile data breaches](https://github.com/nagwww/s3-leaks) that occurred due to mis-configured S3 buckets.

> In our experience, we have seen people storing all sorts of data on poorly secured third-party services, from their credentials in plain text files to pictures of their pets.

There are tools like [Slurp](https://github.com/yamakira/slurp) , [AWSBucketDump](https://github.com/jordanpotti/AWSBucketDump) and [Spaces Finder](https://github.com/appsecco/spaces-finder) to hunt for service specific publicly accessible object storage instances. Tools like Slurp and [Bucket Stream](https://github.com/eth0izzle/bucket-stream) combine Certificate Transparency log data with permutation based discovery to identify publicly accessible S3 buckets.

![](https://cdn-images-1.medium.com/max/1600/1*j6REBngZ8C__74cjXbXLgw.png)
Slurp discovering Amazon s3 buckets using keywords and permutation scanning

![](https://cdn-images-1.medium.com/max/1600/1*FJqT0uGHb8OfN0rzT1Bn_A.png)
Slurp discovering Amazon s3 buckets using CT log data and permutation scanning

5.Wayback Machine is massive digital archive of the World Wide Web and other information on the Internet. Wayback Machine also contains the historical snapshots of websites. [Wayback CDX Server API](https://archive.org/help/wayback_api.php) makes it easy to search through the archives. [waybackurls](https://github.com/tomnomnom/waybackurls/) is neat tool to search for data related to a site of interest.

> Digging through Wayback Machine archive is quite useful in identifying subdomains for a given domain, sensitive directories, sensitive files and parameters in an application.

```
go get github.com/tomnomnom/waybackurls
waybackurls icann.org
```

![](https://cdn-images-1.medium.com/max/1600/1*ByofojY2taaQgpvoHiOdZw.png)
"waybackurls" extracting URLs that belong to a domain that are listed in Way back machine archive

6. [Common Crawl](http://commoncrawl.org/) is a project that builds and maintains a repository of web crawl data that can be accessed and analysed by anyone. Common Crawl contains historical snapshots of websites along with metadata about the website and services providing it. We can use [Common Crawl API](https://index.commoncrawl.org/) to search their indexed crawl data for sites of interest. [cc.py](https://github.com/si9int/cc.py) is a neat little tool to search for crawl data for sites of interest.

```
python cc.py -o cc_archive_results_icann.org icann.org
```

![](https://cdn-images-1.medium.com/max/1600/1*4X5LLtM8Y9w7VZnv6oUHGw.png)
"cc.py" extracting URLs that belong to a domain that are listed in Common Crawl archive

7. [Censys](https://censys.io/) is a platform that aggregates massive Internet wide scan data and provides an interface to search through the datasets. Censys categorises the datasets into three types — IPv4 hosts, websites, and SSL/TLS certificates. Censys has treasure trove of information on par with [Shodan](http://shodan.io), if we know what to look for and how to look for it.

[Censys has an API](https://censys.io/api) that we can use to run queries against the datasets. We wrote a Python script that connects to the Censys API, queries for SSL/TLS certificates for a given domain and extracts sudomains and email addresses that belong to the domain. The script is available here —

[**yamakira/censys-enumeration**
_A script to extract subdomains/emails for a given domain using SSL/TLS certificate dataset on Censys …
_github.com](https://github.com/yamakira/censys-enumeration "https://github.com/yamakira/censys-enumeration")[](https://github.com/yamakira/censys-enumeration)

![](https://cdn-images-1.medium.com/max/1600/1*lDZBfwWjghMgjrVv9t1rhQ.png)
"censys-enumeration" extracting subdomains and email addresses using Censys API

![](https://cdn-images-1.medium.com/max/1600/1*EIZuxVhmLNL8MXewJ0zRbA.png)
Subdomains and email addresses extracted by "censys-enumeration" using Censys API

8. Censys project collects SSL/TLS certificates from multiple sources. One of the techniques used is to probe all the machines on public IPv4 address space on port 443 and aggregate the SSL/TLS certificates they return. Censys provides a way to correlate SSL/TLS certificate gathered with IPv4 hosts that provided the certificate.

> Using correlation between SSL/TLS certificates and the IPv4 host that provided the certificate, it is possible to expose origin servers of a domains that are protected by services like Cloudflare.

CloudFlair is a tool that does a great job at exposing origin servers of a domain using Censys. The tool is available here —

[**christophetd/CloudFlair**
_Find origin servers of websites behind by CloudFlare using Internet-wide scan data from Censys. …
_github.com](https://github.com/christophetd/CloudFlair "https://github.com/christophetd/CloudFlair")[](https://github.com/christophetd/CloudFlair)

![](https://cdn-images-1.medium.com/max/1600/1*2lNGZe1j-yPY5kby5yT0Gw.png)
"Cloud Flair" identifying the origin server IP addresses for medium.com

9. Source code repos are a treasure trove of information during security assessments. Source code can reveal a lot of information ranging from credentials, potential vulnerabilities to infrastructure details etc. GitHub is an extremely popular version control and collaboration platform that you should look at. Gitlab and Bitbucket are also popular services where you might find source code of a target organisation.

Tools like [GitHubCloner](https://github.com/mazen160/GithubCloner) comes in very handy to automate the process of cloning all the repos under a Github account.

```
$ python githubcloner.py --org organization -o /tmp/output
```

[**mazen160/GithubCloner**
_A script that clones Github repositories of users and organizations. - mazen160/GithubCloner_github.com](https://github.com/mazen160/GithubCloner "https://github.com/mazen160/GithubCloner")[](https://github.com/mazen160/GithubCloner)

There are various tools that automate the process of finding secrets in source code repos such a [Gitrob](https://github.com/dxa4481/truffleHog), [truffleHog](https://github.com/dxa4481/truffleHog), [git-all-secrets](https://github.com/anshumanbh/git-all-secrets) etc.

10. [_Forward DNS_ dataset](https://opendata.rapid7.com/sonar.fdns_v2/) is published as part of [Rapid7's Open Data project](https://opendata.rapid7.com/). This data a collection of responses to DNS requests for all forward DNS names known by Rapid7's Project Sonar. The data format is a gzip-compressed JSON file. We can parse the dataset to find sub-domains for a given domain. The dataset is massive though(20+GB compressed, 300+GB uncompressed). In the recent times, the dataset has been broken into multiple files based on the type of DNS records the data contains.

![](https://cdn-images-1.medium.com/max/1600/1*bV2B51GoRdiMSv0Yb28HEA.png)
Extracting domains/subdomains from FDNS dataset

11. Content Security Policy(CSP) defines the `Content-Security-Policy` HTTP header, which allows us to create a whitelist of sources of trusted content, and instructs the browser to only execute or render resources from those sources

Content-Security-Policy header will list a bunch of sources(domains) that might be of interest to us as an attackers. We wrote a simple script to parse and resolve the domain names listed in a CSP header. The script is available here —

[**yamakira/domains-from-csp**
_A script to extract domain names from Content Security Policy(CSP) headers - yamakira/domains-from-csp_github.com](https://github.com/yamakira/domains-from-csp "https://github.com/yamakira/domains-from-csp")[](https://github.com/yamakira/domains-from-csp)

![](https://cdn-images-1.medium.com/max/1600/1*olkJf6qPuSn9VDLEPTaivQ.png)

12. A Sender Policy Framework(SPF) record and is used to indicate to receiving mail exchanges which hosts are authorised to send mail for a given domain

Simply put, an SPF record lists all the hosts that are authorised send emails on behalf of a domain. Sometimes SPF records leak internal net-blocks and domain names.

Services like [Security Trails](http://securitytrails.com) provides historical snapshots of DNS records. We can take a look at historical SPF records to discover internal net-blocks and domain names for a given domain that are listed in the SPF record.

![](https://cdn-images-1.medium.com/max/1600/1*p8L72lfNQaAtbKFKBqLmIw.png)
Historical SPF records for icann.org displayed by [Security Trails](http://securitytrails.com)

We wrote a quick script that extracts netblocks and domains from SPF record for a given domain. The script can also return ASN details for each asset when it is run with _\-a_ option. The script is available here —

[**yamakira/assets-from-spf**
_A Python script to parse net blocks & domain names from SPF record - yamakira/assets-from-spf_github.com](https://github.com/yamakira/assets-from-spf "https://github.com/yamakira/assets-from-spf")[](https://github.com/yamakira/assets-from-spf)

```
python assets_from_spf.py icann.org -a | jq .
```

![](https://cdn-images-1.medium.com/max/1600/1*fkWAcHMa6gJHjiANNjOQxA.gif)

### Conclusion

In this article, we have looked at various OSINT techniques that we use day to day in our security assessments. Although this article is extensive, it is no way meant to be exhaustive. OSINT landscape is ever changing and there is no one size fits all. We made an effort to cover techniques that will improve coverage during the reconnaissance phase of a penetration test.

This brings us to the end of this post. If there are techniques that you frequently use that have yielded you interesting results and if you would like to share those, please do leave a comment.

Until next time, happy hacking!!

### References

*   https://blog.appsecco.com/open-source-intelligence-gathering-101-d2861d4429e3
*   https://blog.appsecco.com/certificate-transparency-part-3-the-dark-side-9d401809b025

*   https://blog.appsecco.com/a-penetration-testers-guide-to-sub-domain-enumeration-7d842d5570f6
*   https://www.certificate-transparency.org
*   https://www.arin.net/resources/services/whois_guide.html
*   https://index.commoncrawl.org/
*   https://www.upguard.com/breaches/cloud-leak-accenture
*   https://www.0xpatrik.com/censys-guide/
*   https://www.0xpatrik.com/osint-domains/
*   https://opendata.rapid7.com/sonar.fdns_v2/

« Previous Paper   Next Paper »