

## Recon:

### 1. Discovering IP Addresses:

<http://bgp.he.net> -Autonomous System Number and IP range registered

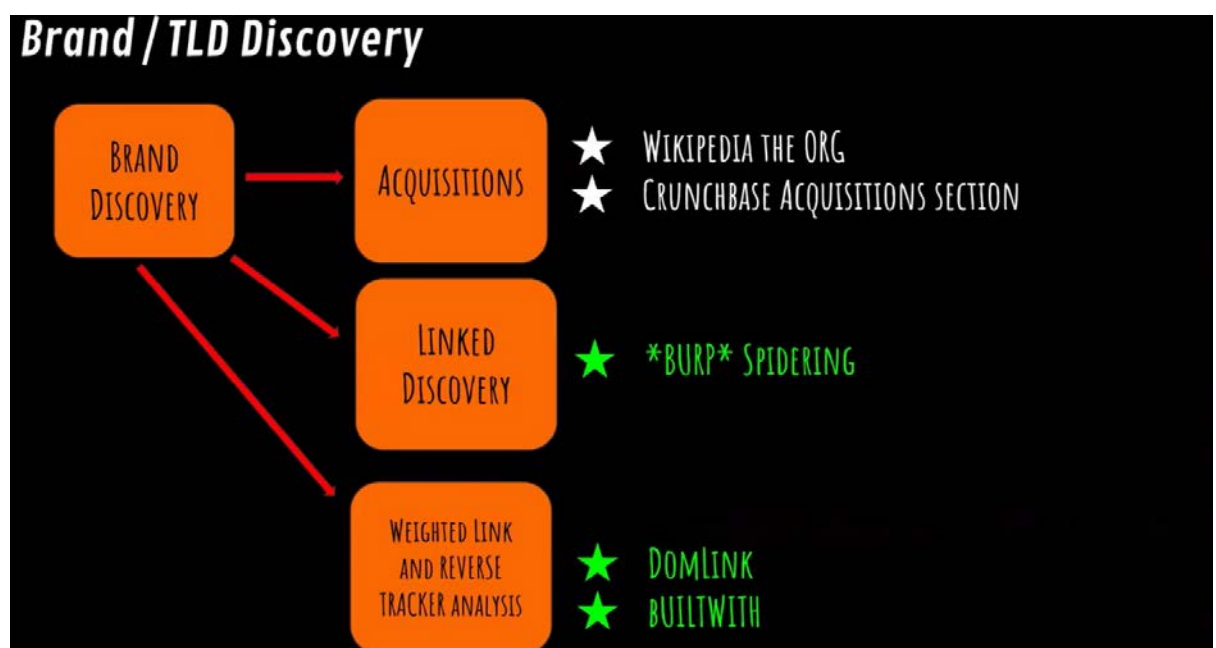
<https://whois.arin.net/UI/query.do>

<https://apps.db.ripe.net/db-web-ui/#/fulltextsearch/>

<https://reverse.report> -Reverse whois

<https://www.shodan.io>

### 2. Discovering New Targets (Brands & Top Level Domain's)



## **Linked Discovery (Burp Demo)**

- 1) TURN OFF PASSIVE SCANNING
- 2) SET FORMS AUTO TO SUBMIT (IF YOU'RE FEELING FRISKY)
- 3) SET SCOPE TO ADVANCED CONTROL AND USE STRING OF TARGET NAME (NOT A NORMAL FQDN)
- 4) WALK+BROWSE, THEN SPIDER ALL HOSTS RECURSIVELY!
- 5) PROFIT (MORE TARGETS)!

### 3. Discovering New Targets : Subdomains

#### Subdomain SCRAPING –



#### Tools : Amass

```
root@Test2:~/tools/amass# ./amass.sh netflix.com
www.netflix.com
media.netflix.com
www.geo.netflix.com
www.eu-west-1.prodna.netflix.com
```

```
root@Test2:~/tools/amass# cat amass.sh

#!/bin/bash
mkdir $1
touch $1/$1.txt
amass -active -d $1 |tee /root/tools/amass/$1/$1.txt
```

## Subfinder – Bruteforcing Available but massdns is faster

```
root@Test2:~/tools/subfinder# ./subfinder.sh twitch.tv

SubFinder v0.1.0      Made with ♥ by @Ice3man
```

```
root@Test2:~/tools/subfinder# cat subfinder.sh

#!/bin/bash
mkdir $1
touch $1/$1.txt
subfinder -d $1 |tee /root/tools/subfinder/$1/$1.txt
```

## Subdomain BruteForcing – Use All.txt file from Jhaddix available on GITHUB for DNS Resolver List

### Subdomain Brute Forcing

1,136,964 LINE SUBDOMAIN DICTIONARY (ALL.TXT)


SUBFINDER?

Tool	Time to run	Threads	Found
subbrute <small>time ./subbrute.py -c 100 all.txt \$TARGET.com   tee subbrute.output</small>	errored	100	0
→ gobuster <small>time gobuster -n dns -u \$TARGET.com -t 100 -w all.txt</small>	21m15.857s	100	87
→ massdns <small>time ./subbrute.py /root/work/bin/all.txt \$TARGET.com   ./bin/massdns -r resolvers.txt -t A -a -o -w massdns_output.txt -</small>	1m24.167	n/a	213
dns-parallel-prober <small>time python dns-queue.py \$TARGET.com 100 \$TARGET_outputfile -t /root/work/bin/all.txt</small>	42m2.868s	100	43
blacksheepwall <small>time ./blacksheepwall_linux_amd64 -clean -dictionary /root/work/bin/all.txt -domain \$TARGET.com</small>	256m9.385s	100	61

## Auxiliary

- ★ DNSSEC / NSEC / NSEC3 WALKING
  - LDNSUTILS, NSEC3WALKER, NSEC3MAP
- ★ GITHUB RECON
  - SEARCH FOR GOODIES
- ★ DORKING: ADS KEY, PRIV POL, TOS, AWS, S3

### ESOTERIC SUB-DOMAIN ENUMERATION TECHNIQUES



**BHARATH KUMAR**  
BUGCROWD LEVELUP | JULY 15TH 2017

[HTTPS://GITHUB.COM/APPSECCO/BUGCROWD-LEVELUP-SUBDOMAIN-ENUMERATION/blob/master/esoteric\\_subdomain\\_enumeration\\_techniques.pdf](https://github.com/appsecco/bugcrowd-levelup-subdomain-enumeration/blob/master/esoteric_subdomain_enumeration_techniques.pdf)

[HTTPS://WWW.YOUTUBE.COM/WATCH?v=1K6D\\_537EoI](https://www.youtube.com/watch?v=1K6D_537EoI)

**Github Recon**

- Environments (dev, stage, prod)
- Secret Keys (API\_key, AWS\_Secret, etc.)
- Internal credentials
- API endpoints
- Domain patterns

## Enumerating Targets:

Masscan :

## Port Scanning

65536 UNVERIFIED HOSTS (A LARGE TARGETS ASN)

Tool	Time to run	Found
Masscan <small>masscan -p1-65535 -iL \$TARGET_LIST --max-rate 100000 -oG \$TARGET_OUTPUT</small>	11m4.164s	196
nmap	∞	ZZZ

```
#!/bin/bash
strip=$(echo $1|sed 's/https\?:\/\/\\\\/\\\\/')
echo ""
echo "#####"
host $strip
echo "#####"
echo ""
masscan -p1-65535 $(dig +short $strip|grep -oE "\b([0-9]{1,3}\.){3}[0-9]{1,3}\b"|head -1)
--max-rate 1000 |& tee $strip_scan
```


Eyewitness:

## Visual Identification

```
root@kali:~/Desktop/tools/Eyewitness# python Eyewitness.py --prepend-https -f ../domain/tesla.com.lst --all-protocols --headless
```

## Brutespray:

### Credential bruteforce



<https://github.com/x90skysn3k/brutespray>

```
python brutespray.py --file nmap.gnmap -U
/usr/share/wordlist/user.txt -P /usr/share/wordlist/pass.txt
--threads 5 --hosts 5
```

### Credential bruteforce

```
Loading File: \
Welcome to interactive mode!

WARNING: Leaving an option blank will leave it empty and refer to default

Available services to brute-force:
Service: ftp on port 21 with 9 hosts
Service: smtp on port 25 with 8 hosts
Service: smtp on port 587 with 1 hosts
Service: ssh on port 22 with 8 hosts
Service: telnet on port 23 with 1 hosts
Service: mysql on port 3306 with 1 hosts

Enter services you want to brute - default all (ssh,ftp,etc): ftp,ssh,telnet
Enter the number of parallel threads (default is 2): 5
Enter the number of parallel hosts to scan per service (default is 1): 10
Would you like to specify a wordlist? (y/n): y
Enter a userlist you would like to use:
Enter a passlist you would like to use: /usr/share/wordlists/
/usr/share/wordlists/dirb /usr/share/wordlists/ferret-wifi /usr/share/wordlists/sqlmap.txt
/usr/share/wordlists/dirbuster /usr/share/wordlists/metasploit /usr/share/wordlists/wfuzz
/usr/share/wordlists/dnsmap.txt /usr/share/wordlists/nmap.lst
/usr/share/wordlists/fasttrack.txt /usr/share/wordlists/rockyou.txt.gz
Enter a passlist you would like to use: /usr/share/wordlists/metasploit/password.lst
Would you like to specify a single username or password (y/n): y
Enter a username: admin

Starting to brute, please make sure to use the right amount of threads(-t) and parallel hosts(-T)... \
Brute-Forcing...
Medusa v2.2 [http://www.fooofus.net] (C) JoMo-Kun / Foofus Networks <jmk@fooofus.net>
```

## Wayback:

### Wayback Enumeration

**Jason Haddix** @jhaddix · May 9

#BountyProTip: found a 401/403, basic auth, or domain that seems interesting but is somehow locked down? Look at its [archive.org/web/](https://archive.org/web/) entries. Sometimes you win instantly with API keys or URL structure that you can forcefully browse to unprotected content still there.

**Brett Buerhaus** @bbuerhaus · May 9

Replying to @jhaddix

Can confirm, [archive.org](https://archive.org/) has led me to many bounties. Highest impact is usually old files on websites (not properly cleaned up) or acquisitions with older code. They also have a json endpoint that allows for easy automation, e.g.: [web.archive.org/cdx/search?url...](https://web.archive.org/cdx/search?url...)

**Mohammed Diaa** @mhmddiaa · May 10

Replying to @jhaddix

If there are so many entries that you can't go through all of them manually, you can use [waybackunifier](https://github.com/mhmddiaa/waybackunifier) to get the unique parts out of each snapshot and save them together in a unified file.

**Dawood Ikhtaq** @daudmalik06 · May 10

Replying to @jhaddix

i think [github.com/daudmalik06/ReconCat](https://github.com/daudmalik06/ReconCat)... this tool can help for this purpose.

**daudmalik06/ReconCat**

ReconCat - A small Php application to fetch archive url snapshots from archive.org, using it you can fetch complete list of snapshot urls of any year or complete l...

[tomnomnom / waybackurls](https://github.com/tomnomnom/waybackurls)

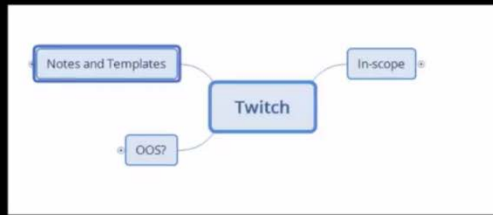
Code Issues Pull requests Projects Wiki Insights

Fetch all the URLs that the Wayback Machine knows about for a domain

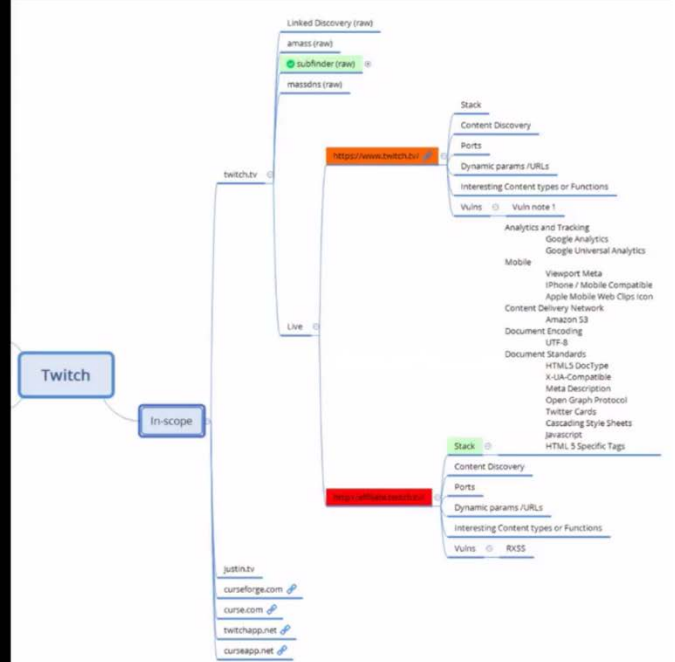


### Mapping:

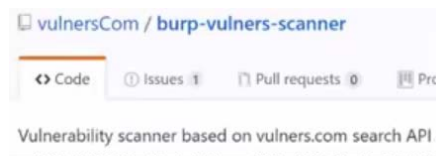
## Xmind Organization



- ★ GREEN w/ CHECKMARK IS DONE
- ★ ORANGE IS IN PROGRESS
- ★ RED IS VULNERABLE

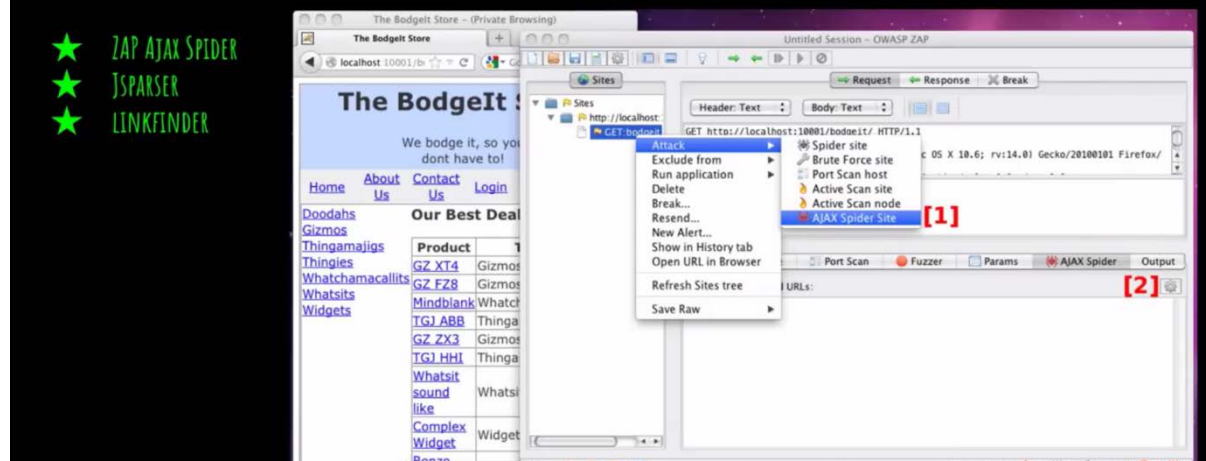


## Platform Identification and CVE searching



## Parsing Javascript:

## Coverage for Heavy js sites



# Linkfinder

- Full URLs (<https://example.com/>)
- Absolute URLs or dotted URLs (/ or ../)
- Relative URLs with atleast one slash (text/test.php)
- Relative URLs without a slash (test.php)

The output is given in HTML. Karel\_origin has written a chrome extension for LinkFinder which can be found [here](#).

## Screenshots

```
/transfer_eligible_programs
```

```
url: "/reports/" + t + "/transfer_eligible_programs",
```

```
/notifications
```

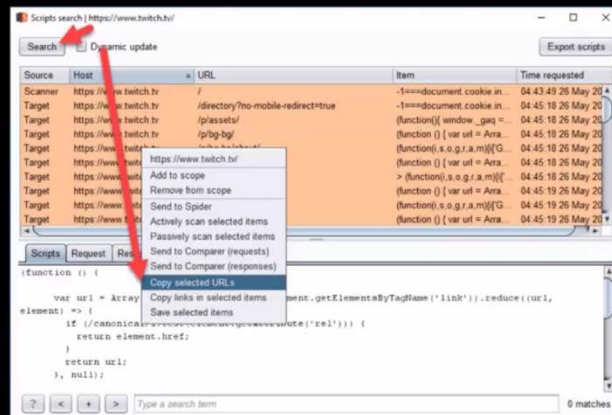
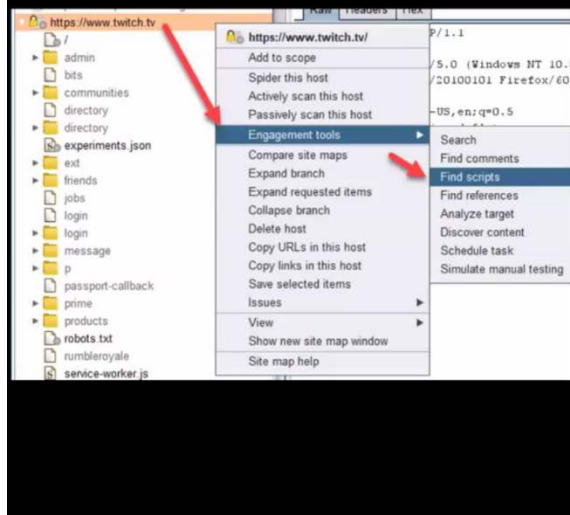
```
return "/notifications"
```

```
/creditcards
```

```
url: "/" + this.team.get("handle") + "/creditcards",
```

```
/creditcards/deactivate
```

# Feeding these tools



## Content Discovery:

# Content Discovery / Directory Bruteforcing

TBHMV1

- SECLISTS / RAFT / DIGGER WORDLISTS
- PATATOR
- WPSCAN
- CMSMAP

- ★ GOBUSTER
- ★ BURP CONTENT DISCOVERY
- ★ ROBOTS DISALLOWED
- ★ -\\_(ツ)\_/-

```
root@kali:~/Desktop/tools/gobuster# wc -l ../seclists/Discovery/Web_Content/raft-large-words.txt
119600 ../seclists/Discovery/Web_Content/raft-large-words.txt
root@kali:~/Desktop/tools/gobuster# time ./gobuster -w ../seclists/Discovery/Web_Content/raft-large-words.txt -s 200,301,307 -t 100 -u https://www.tesla.com/

Gobuster v1.3.0 OJ Reeves (@TheColonial)
=====
[*] Url/Domain : https://www.tesla.com/
[*] Threads : 100
[*] Wordlist : ../seclists/Discovery/Web_Content/raft-large-words.txt
[*] Status codes : 200,301,307
=====
/Includes (Status: 301)
/Modules (Status: 301)
/contact (Status: 200)
/user (Status: 307)
/Log (Status: 200)
/about (Status: 200)
/gallery (Status: 302)
/themes (Status: 301)
/shop (Status: 301)
/admin (Status: 307)
/customer (Status: 301)
/ (Status: 200)
/admin (Status: 307)
/misc (Status: 301)
/libraries (Status: 301)
/scripts (Status: 301)
/info (Status: 201)
/profiles (Status: 301)
/r/tes (Status: 301)
/home (Status: 301)
/updates (Status: 200)
/legal (Status: 301)
/compare (Status: 200)
/careers (Status: 200)
/models (Status: 200)
/homepage (Status: 301)
/used (Status: 200)
/energy (Status: 200)
```

gobuster

Code Issues Pull requests

Directory/file & DNS busting tool written in Go

go pentesting tool dns web

danielmiessler / RobotsDisallowed

Code Issues Pull requests Projects Wiki Insights

A harvest of the Disallowed directories from the robots.txt files of the world's top websites.

```

rootkali:~/Desktop/tools/gobuster# wc -l ../secLists/Discovery/Web_Content/raft-large-words.txt
119600 ../secLists/Discovery/Web_Content/raft-large-words.txt
rootkali:~/Desktop/tools/gobuster# time ./gobuster -w ../secLists/Discovery/Web_Content/raft-large-words.txt -s 200,301,307 -t 100 -u https://www.tesla.com/

```

Use Content\_discovery\_all.txt from Jhaddix on github.

Parameter Discovery:

Parameth

Backlash Powered Scanner

## Parameter Bruteforcing?

★ YEP! - UNTESTED BUT LOVE THE IDEA

★ CAN BE COMBINED WITH BACKSLASH SCANNERS TOP 2500 ALEXA PARAMS

maK- / parameth

Code Issues Pull requests Projects

This tool can be used to brute discover GET and POST parameters

```

parameth/maK# ./parameth.py -u https://makthepla.net/parameth/simpletest.php
[Progress Bar]
parameth v1.0 - find parameters and cralic rocks
Author: Ciaran McElally - https://makthepla.net

Establishing base figures...
GET: content-length: 22 status: 200
POST: content-length: 22 status: 200
Scanning it like you own it...
GET (size): m | 22 ->36 ( https://makthepla.net/parameth/simpletest.php?m=discobiscuits )
POST (size): r | 22 ->42 ( https://makthepla.net/parameth/simpletest.php )
GET (status): redirect | 200->301 ( https://makthepla.net/parameth/simpletest.php?redirect=discobiscuits )
parameth/maK#

```

PortSwigger / backlash-powered-scanner

Code Issues Pull requests Projects Wiki Insights

Branch: master backlash-powered-scanner / resources / params

albinowax Detect soft string injection, handle HTTP errors better, detect back...

1 contributor

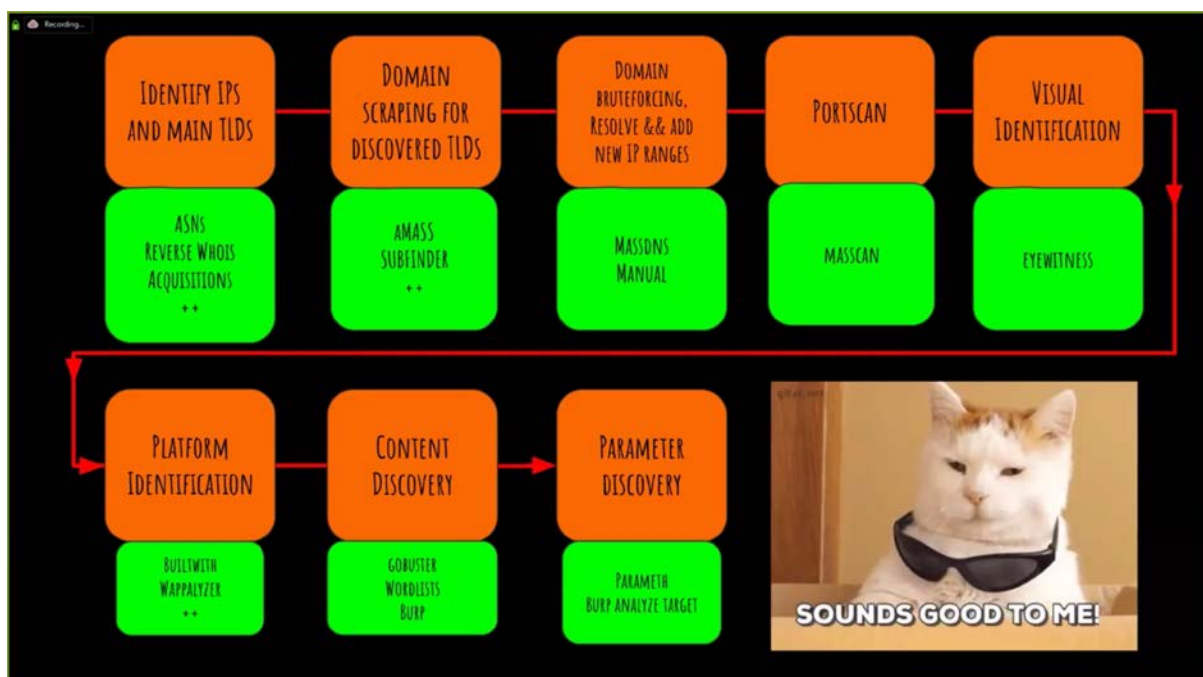
2588 lines (2588 sloc) | 18.8 KB

```

1 id
2 action
3 page
4 name
5 password
6 url
7 email
8 type
9 username
10 file
11 title
12 code
13 q
14 submit
15 user
16 token
17 delete
18 message
19 t
20 c
21 data
22 mode
23 order
24 lang
25 p
26 key
27 status

```

Recon – Enumeration MAP:





## XSS:

# Blind XSS Frameworks Continued!

LewisArdern / bXSS

Watch 4 Star 51 Fork 8

Code Issues 4 Pull requests 0 Projects 0 Wiki Insights

bXSS is a simple Blind XSS application adapted from <https://cure53.de/m>

ssl / ezXSS

Watch 19 Star 244 Fork 54

Code Issues 1 Pull requests 1 Projects 0 Wiki Insights

ezXSS is an easy way to test (blind) XSS

payload xss blind php screenshot test xss-vulnerability xss-exploitation xss-detection xss-attacks xss-injection xss-scanner

blind-xss easy-to-use easy

## SSRF:

jhaddix / cloud\_metadata.txt

Edit Delete Star 55


Code Revisions 11 Stars 55 Forks 25 Embed <script src="https://gist..."

Cloud Metadata Dictionary useful for SSRF Testing

```
cloud_metadata.txt
# AWS
# from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-metadata.html#instancedata-data-categories
# http://169.254.169.254/latest/user-data
# http://169.254.169.254/latest/user-data/iam/security-credentials/[ROLE NAME]
# http://169.254.169.254/latest/meta-data/iam/security-credentials/[ROLE NAME]
# http://169.254.169.254/latest/meta-data/ami-id
# http://169.254.169.254/latest/meta-data/reservation-id
# http://169.254.169.254/latest/meta-data/hostname
# http://169.254.169.254/latest/meta-data/public-keys/[ID]/openssh-key
# http://169.254.169.254/latest/meta-data/public-keys/[ID]/openssh-key
# AWS - Dirs
# http://169.254.169.254/
# http://169.254.169.254/latest/meta-data/
# http://169.254.169.254/latest/meta-data/public-keys/
# Google Cloud
# https://cloud.google.com/compute/docs/metadata
# - Requires the header "Metadata-Flavor: Google" or "X-Google-Metadata-Request: True"
# http://169.254.169.254/computeMetadata/v1/
# http://metadata.google.internal/computeMetadata/v1/
# http://metadata.google.internal/computeMetadata/v1/instance/hostname
# http://metadata.google.internal/computeMetadata/v1/instance/id
# http://metadata.google.internal/computeMetadata/v1/project/project-id
# Google allows recursive pulls
# http://metadata.google.internal/computeMetadata/v1/instance/disks/?recursive=true
# Google
# Beta does NOT require a header atm (thanks Mathias Karlsson @evilid3ntrun)
# http://metadata.google.internal/computeMetadata/v1beta1/
```

## What to do with SSRF?

<https://gist.github.com/jhaddix/78cece26c91c6263653f318a453e2738>



There is no cloud  
it's just someone else's computer

## IDOR-MFLAC:

# IDOR - MFLAC

- ★ IDS
- ★ HASHES
- ★ EMAILS

Request

Raw Params Headers Hex

POST /v2...son/123456787&type=journeys&before=1&score=1433142020&brow...  
id=28338336&ang=en&uid=pghlevyBHVZ+ap9/JpwUpItnk8Q=&app\_version=6.5.0.1.1...  
Accept: \*/\*  
Content-Length: 214  
Accept-Encoding: gzip  
X-Zomato-API-Key: 2d3  
Content-Type: application/json  
User-Agent: Zomato/5.0  
Host: api.zomato.com  
Connection: Keep-Alive  
Cache-Control: no-cache

## Insecure Direct Object Reference 🔥🔥

{regex + perm} id	{regex + perm} user	
{regex + perm} account	{regex + perm} number	
{regex + perm} order	{regex + perm} no	
{regex + perm} doc	{regex + perm} key	
{regex + perm} email	{regex + perm} group	
{regex + perm} profile	{regex + perm} edit	REST numeric paths

<http://acme.com/script?user=21856>

## Subdomain Takeover:

Edoverflow – Can I take over XYZ

## Robbing Misconfigured Shit: Sa7mon – S3Scanner [AWS]

# Robbing Misconfigured Shit\*\* (AWS)

sa7mon / S3Scanner

Watch 17 Star 433 Fork 64

Code Issues (10) Pull requests (0) Projects (0) Wiki Insights

Scan for open S3 buckets and dump

amazon s3 aws

Examples

This tool accepts the following type of bucket formats to check:

- bucket name - google-dev
- domain name - uber.com, sub.domain.com
- full s3 url - yahoo-staging-s3-us-west-2.amazonaws.com (To easily combine with other tools like bucket-stream)
- bucketregion - flaws.cloudus-west-2

```
> cat names.txt
flaws.cloud
google-dev
testing.microsoft.com
yelp-production-s3-us-west-1.amazonaws.com
github-dev-us-east-1
```

```
> python ./s3scanner.py sites.txt
2018-03-18 22:15:28 [found] : ada-staging | Unknown Size - timeout | ACLs: {'allUsers': ['READ'], 'authUsers': ['READ', 'WRITE', 'READ_ACP']}
2018-03-18 22:15:32 [found] : admin | 107.0 KiB | ACLs: AccessDenied
2018-03-18 22:15:33 [not found] : google.cn
2018-03-18 22:15:45 [found] : alb-prod | Unknown Size - timeout | ACLs: {'allUsers': ['READ', 'WRITE', 'READ_ACP'], 'authUsers': []}
2018-03-18 22:15:47 [found] : app-dev | AccessDenied | ACLs: AccessDenied
2018-03-18 22:15:53 [found] : alexander-feil | 93.5 MiB | ACLs: {'allUsers': ['READ', 'READ_ACP'], 'authUsers': []}
2018-03-18 22:15:53 [invalid] : gm
2018-03-18 22:16:00 [found] : aneta | 10.7 MiB | ACLs: {'allUsers': ['READ', 'READ_ACP'], 'authUsers': ['READ', 'READ_ACP']}
2018-03-18 22:16:07 [found] : appshack | 297.3 KiB | ACLs: AccessDenied
```



★ OFTEN ON NEWER WEBSITES WE ARE HAMPERED BY WAF OR CDN VENDORS SECURITY PRODUCTS

- CLOUDFLARE AND AKAMAI
- DEDICATED WAFs

★ SOLUTIONS:

- ENCODING (MEH)
- FINDING ORIGIN
- FINDING DEV

Jason Haddix  
@Jhaddix

Security testing against Akamai? look for `origin-sub.domain.com` or `origin.sub.domain.com`, bypass the filtering by going to the source.

12:06 PM · 13 Sep 2017

43 Retweets 95 Likes

2 43 95

Add another Tweet

Tom @theothertom · 13 Sep 2017

Replying to @Jhaddix

Also try sending "Pragma: akamai-x-get-true-cache-key", the cache key often has the origin in it

1 2 3

Jason Haddix @Jhaddix · 13 Sep 2017  
also:

Keeping the Origin IP Address Secret is Difficult

## What's in a name?

- ★ DEV.DOMAIN.COM
- ★ STAGE.DOMAIN.COM
- ★ WW1/WW2/WW3...DOMAIN.COM
- ★ WWW.DOMAIN.UK/IP/...
- ★ ...

Jason Haddix  
@Jhaddix

WAF had me down on `www.$target.com` ='(  
too bad they missed `ww2.$target.com` !

sqli in progress...

#OMGSOMANYTABLESToEXFIL

8:14 PM · 16 Feb 2018

6 Retweets 104 Likes

★ <https://twitter.com/Jhaddix/status/9647456610279680>