

Forum:[Hack Forums](#)

Thread: [Mastering the Art of Google Dorks \[Part 1\]](#)

Shaggy - August 3, 2020, 7:13 pm

Code I have been making Private Dorks for five years now for Oday & research purposes and, it is still the best way for me to find vulnerabilities manually. This is aimed towards beginner level so don't be disappointed if you are that advanced already.

Overview: What is Google Hacking? It is a method used by Penetration Testers or Hacker to successfully find a loophole based on the query itself in which leads to the vulnerable opening of the website itself.

Often known as the most basic of the basic in terms of Hacking and is the most dangerous when a critical private dork falls into the wrong hands that will affect multiple domains and platforms.

Are you struggling to find Fresh and HQ Dorks for you to test or exploit?
Submit Dorks at [GHDB](#) to rank yourself?

Query: Frequently posted on Hacking blogs or posts around the internet and since we are on the topic, I'll give you more information about it and clear up the unnecessary confusion that may arise. I would also just mention the one I commonly use and remove the extras. `inurl: ---` The query I frequently use, It specifically targets `/[vulnerability.php]` on a domain. It basically has the same functions as `[domain:]` but using that query only targets 1 website/domain.

`intext: ---` Whatever is written inside the website that matches your keyword. Used effectively with `[intitle:]` on advanced blind searches.

`intitle: ---` The Header or Title of a Website, Vulnerable Administrator Panels can specifically targeted with this one.

`ext: ---` Used to determine file types on the end of the URL `[filetype:]` is an alternative.

+ --- If you have 2 or more keywords that you want to target specifically. The `[-]` sign is only usable on accurate specific searches.

Believe me or not that is the only queries you will have to use in searching dorks unless you are directly searching something. No need to memorize a whole Encyclopedia of queries.

Too Long or Too Short? I don't know what to describe this one as it's usually making a Google Dork too long or too short making it not usable.

Usually, in simple terms, we can't predict how long or how less it is when making a Google Dork, But let me tell you this,

There's no perfect way on how to know whether your dork is too long or too short.

For example your long dork is:

Code `inurl:wp-content/plugins/updraftplus/js/updraft-admin-restore.js` The searches returned 0 results but let's fix that,

Deleting some keywords will make it look like:

Code `inurl:wp-content/plugins/updraftplus/Problem Fixed` ; Just by deleting some queries will make the search more accurate thus giving you tons of results.

But what about the short dork you have created?

This is actually harder since you are tasked to blind guess directories. It has the same concept as LFI[Local File Inclusion] where the directory is masked with

Code `../../../../admin.php` or `/*/*/*`

Example: Short dork =

Code `inurl:D:\inetpub` The results hardly show any sensitive directory and only leads to Question or Forums.

In this case I will add `[webs\]` since I always see this regarding inetpub keywords. + `[/../]` or `[\.\.]` which will show us the exact vulnerable directories of a website.

Code `inurl:D:\inetpub\webs\.\.\.`

*Do take note that the LFI Query is used on rare websites only. It's basically your last hope on blind guesses. And it won't always matter whether you put it

Extensions: TXT, CSV, XML, SQL, DB, CGI, EXE, JAR, JS,

I could go on for ages on what are the common file types/extensions. But what about the uncommon/rare file types that is either unknown or rarely used?

You can find millions of dorks that have the most file types and that proves to be mostly useless.

What if I told you that I have a private list for the unseen file types that even Most Hackers don't know? And what if that file type is where the most confidential documents of a website are located?

Like come on I always see this common file type dorks everywhere,

Let's reveal a bit of a safe file on a popular website.

This is the perfect example of uncommon file types

Code `Inurl:nasa.gov ext:???` I'll bet you on this one. No matter what Common file types you input after the ext:, The result will always be 1-2 on which contains uninteresting information or Publicly known Articles. In this case I will give you a Unknown Extension that will show NASA or any other Big Websites give out sensitive information!

Code `Inurl:nasa.gov ext:tns` and watch the searches

What? You can't understand any? It's encrypted of course

TNS files can be decrypted but that's my trade secret. Also just look at the URL for the general idea on what it is.

TNS files is mainly used for codes or results of a test. Such as the source code of the robots and many more.

We will be using this encrypted keywords later on and find out how can this encryption can make us actually find a vulnerability.

The Source Code: Would you expect the Source Codes of websites have any meaning on it? Well, it's nonsense because we are in the Year 2020, where Hacking Source Codes doesn't work. But in this possibility that I discovered myself, I was able to find useful keywords that can break-through the defense of a secured website

*I Modified this trick to avoid spying on Real Time Facebook Messages cause we are White hat

Found somewhere in Facebook Source

Code `div id="fbRequestsJewelLoading`

Dork =

Code `intext:div id="fbRequestsJewelLoading` You will see a ton of results and simply go to it and copy the results that looks like a Facebook Source Code

Paste it on your Notepad and save it as .html and open it. You will see their Activities.

But this looks useless, I won't teach you it's exact keywords since it beyond black hat. With proper query and keywords you can specifically target any Facebook Accounts. But if you really want to learn it. I suggest analyzing the source codes and important js scripts and using Live Http Headers as a tool to help you to further exploit it.

Why I wont report to Facebook? It's not on their scope since it uses external connections or almost the same concept as Man in the Middle Attack.

The Part 2: I will end this for now since my fingers are already numb typing, but in the next part[s] I will cover on what's down below

Combining

Creation

Linked Vulnerabilities

Encrypted Dorks

If you have any questions feel free to ask me!

[No Black Hat Questions]

The_Defiant_One - August 4, 2020, 5:51 pm

Nice Tutorial. Looking forward to part 2. Thanks for sharing.