What is Google Dorking?

With the advancement of technology, everyone may use Google, which is one of the most widely used search engines on the planet. Google is a search engine that allows you to find information, data, and other internet resources. But Google's capabilities aren't restricted to this. You will learn how to use Google search techniques for hacking tactics in this tutorial on what Google Dorking is.

What Does It Mean When Google Dorks?

Google Dorking is a hacking technique that utilizes Google's advanced search facilities to seek useful data or material that is difficult to find.

"Google hacking" is another term for Google Dorking. On the surface, Google Dorking entails modifying search results with certain modifiers.

Instead of scanning the entire Web, users can collect photos or obtain information about a single site by clicking on tags like "image" or "site." Other commands, such as "filetype" and "datarange," can be used to generate more particular search results.

Although some forms of Google Dorking are harmless and just use Google's resources, others are alarming to authorities and security experts because they could suggest hacking or cyberattack reconnaissance. Hackers can use these sorts of Google Dorking and other cybercriminals to access illegal data or exploit security flaws in websites, which is why the phrase is gaining a negative connotation in the security community.

Is it Possible for Hackers to Utilize Google to Break into Websites?

Google is frequently misunderstood as merely a search engine for text, photos, videos, and news. However, information security plays a critical role. Google can also be utilized as a hacking tool.

Google cannot be used to hack websites directly. Its incredible Web crawling capabilities, on the other hand, might be extremely useful in indexing nearly anything on any website, even sensitive information. This could include usernames, passwords, and other basic weaknesses that you are unaware of.

Basically, with the help of the native Google Search engine, you may uncover vulnerabilities in any web applications and servers using Google Dorking.

Significance of the Name "Google Dorking"

A Google Dork is an employee who unwittingly publishes confidential company information on the internet. A dork is a slang term for someone who is slow-witted or inept. Attackers util complex search strings known as Google dork searches to find sensitive information.

Getting Rid of Google Dorks

There are numerous methods for avoiding falling under the control of a Google Dork. The following are some of the proposed measures −

- Sensitive data is encoded or encrypted, such as usernames, passwords, payment information, messages, addresses, and phone numbers.
- Run queries against your own site to see if any sensitive information may be found. If you come across any sensitive information, you can use Google Search Console to delete it from search results.
- A **robots.txt** file in your root-level site catalog can be used to protect sensitive content. While using robots.txt helps prevent Google from crawling our site, it can also reveal valuable data to an attacker.

Google Dorking Examples

Let's look at a few Google Dork instances and see how they might be utilized to find private information on the internet.

For Login Credentials, Look Through LOG Files

This is a method for locating ".LOG" files that have been mistakenly uploaded to the internet. This is essentially a LOG file containing information on the system's credentials or the numerous user/administrator accounts in the system.

Two Google operators are used in the Dork command.

You can also use two Google operators in combination, all in text and filetype.

```
allintext:username filetype:log
```

The command above will show you all the results, including usernames in ***.log** files.

Consider the Options ENV files are used

Various prominent web development frameworks use env to declare global variables and configurations for the local and development environments.

```
DB_USERNAME filetype:env
```

```
DB_PASSWORD filetype:enc=v
```

You may get a list of sites that make their **env** file public on the internet by running the command. Most developers save their **".env"** files in the main website's public directory, which might be dangerous to their site if it falls into the hands of cybercriminals.

Unencrypted users, passwords, and IP addresses are directly revealed in the search results if you click any of the exposed ".env" files.

Probe Live Cameras

This may sound a little disturbing, but have you ever wondered whether anyone might see your private live camera on the internet?

Using Google hacking techniques, you can get live camera web pages that aren't IP-restricted. Suppose you're brave enough to experiment with Google Dork. In that case, you can view and operate the entire admin panel from afar and even re-configure the cameras to your liking.

You can get a list of publicly visible live cameras by entering "top.htm" in the URL with the current time and date.

```
inurl:top.htm inurl:currenttime
```

Another dork for cameras is a list of typical router-hosted live-view pages.

```
inurl:"lvappl.htm"
```

Open FTP Servers to Investigate

Internal information may be accidentally published as a result of a failure to set access permissions in the FTP. Even more perilous is the risk of the FTP server being used as "storage" for computer viruses and illegally copied files if it is set to "Write."

You may quickly explore the publicly exposed FTP Servers with the following dork command, which can sometimes examine many topics.

```
intitle:"index of" inurl:ftp
```

You can just execute the following dork command to find a list of websites that use the HTTP protocol.

```
intitle:"index of" inurl:http after:2018
```

By simply changing the text in the search title, you can be more specific and look for online forums that use HTTP.

```
intitle:"forum" inurl:http after -2018
```

Check Out the Most Recent Cache

This will display the most current cache of a given webpage. This can be useful for determining the last time a page was crawled.

```
cache:websitename.com
```