Google dorking for beginners: how to find more vulnerabilities using Google search

BY BLACKBIRD-EU · OCTOBER 27, 2024 · LAST UPDATED ON JULY 28, 2025

Bug bounty hunters who spend time in content discovery and reconnaissance, in general, are always rewarded well for their efforts as they often come across untested and hidden assets or endpoints. Google dorking is another way to leverage search engines to discover hidden assets and endpoints to increase your chances of finding vulnerabilities.

This article is a guide specifically for beginners with no prior knowledge about using search engines to find exposed files.

We are certain that by the end of this article, you will be able to find more content and assets for you to test by leveraging search engines like Google. Let's dive in!

What is Google dorking?

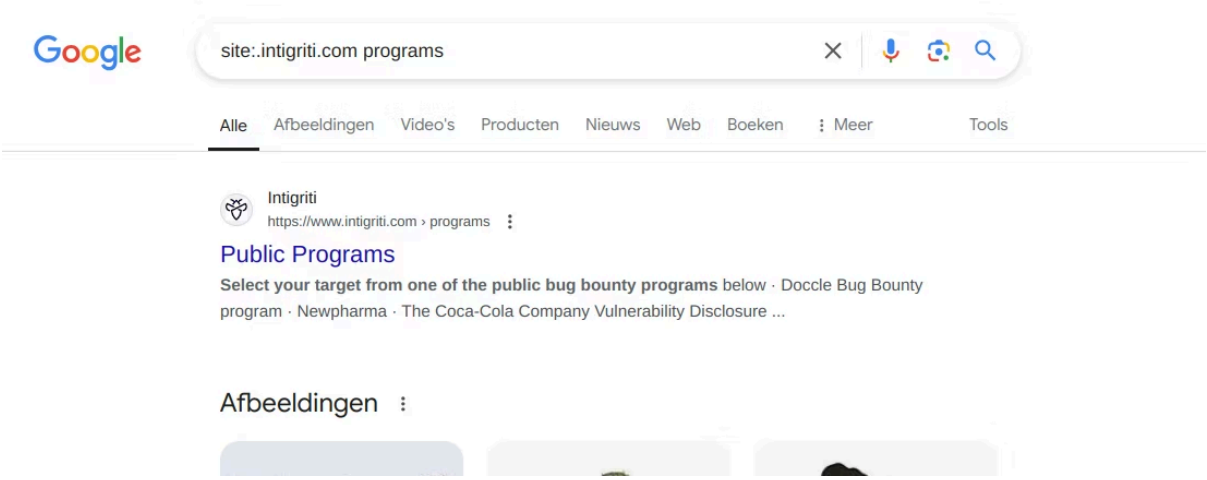
Google dorking (often also referred to as Google hacking) is leveraging search engines and their capabilities to discover more assets or links that were previously indexed by Google. Some indexed files are interesting as they can contain sensitive data or are vulnerable to certain vulnerability types.

Search engines provide syntaxis (or search operators) that we can use to further narrow down our search. We will be specifically focusing on these search operators as we can use them to search through index files to find interesting links, suspicious files and other exposed data.

Basic search operators

We all know that when we search for a certain keyword, all the results will be returned that match this specific keyword. But we can further narrow down our search to only list indexed results that are found on a certain domain:

```
site:.intigriti.com programs
```



Example of a basic search

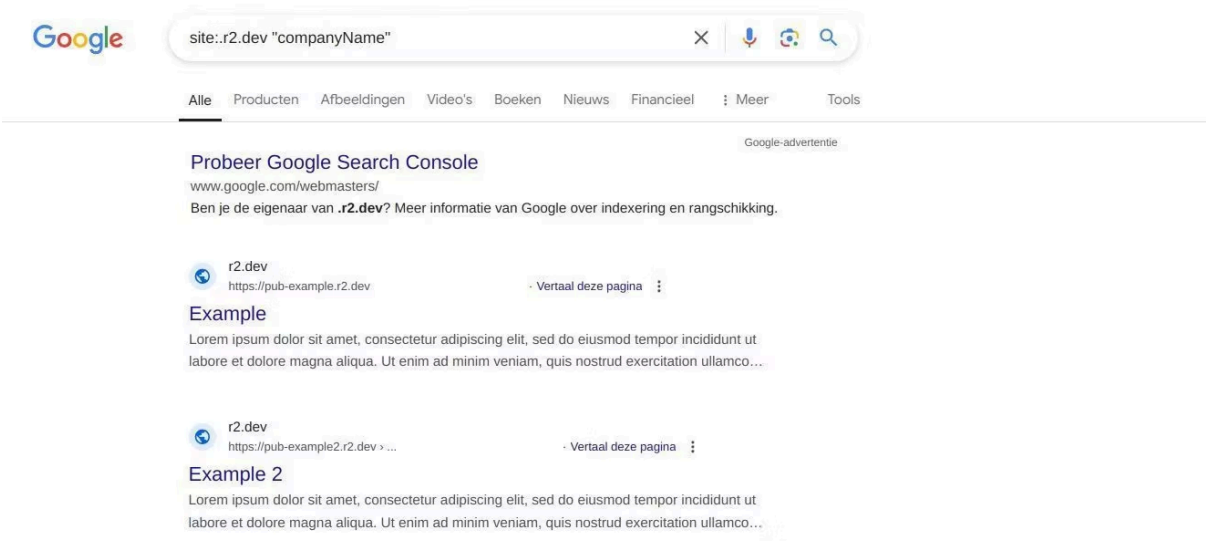
Let's break the above search query down:

- The **site:** search operator is used to filter search results by domain. As you can see, we included a single dot character in front of the domain name (in this case intigriti.com), this acts as a wildcard character and instructs Google to include results for all subdomains of that domain.
- The keyword "programs" in quotation marks is set to only match results for that specific keyword

We can now use this to find:

- Login or registration forms
- Admin panels or other authentication portals

We've previously also used this [to enumerate Cloudflare R2 cloud buckets](#) for example:



Enumerating CF R2 cloud buckets

Now that we covered the basics, let's get into the more advanced cases that will help us find more vulnerabilities!

Advanced search operators

In several cases, we'd want to find indexed results that contain interesting or suspicious keywords in the page title, URL or the body itself. Ideally, we also want to filter by file type or extension.

So let's cover some interesting search operators that help us narrow down our searches.

Filtering in-page keywords:

As we've mentioned before, filtering results that contain interesting keywords can help us identify components tremendously. A common example would be to search a login or register panel:

```
site:.example.com intext:Login
```

Or if we are interested in several keywords, we can make use of the "OR" operator and chain our query:

```
site:.example.com (intext:Login OR intext:Register or intext:Create Account)
```

This query would return all pages that:

- Match the domain name
- Match the keywords "Login", "Register" or "Create Account"

We can do the same by filtering page titles.

Filtering by page title

We can replicate the query but this time only filter by page title:

```
site:.example.com AND (intitle:Login OR intitle:Register or intitle:Create An Account)
```

And of course, this search query would match all indexed links that have the keywords in the page title.

Let's move on to the next search operator that would allow us to filter by URL.

Filtering by URL

We can further narrow down our search by looking for certain keywords in the URL itself. If we try to replicate the query above and again focus on finding portals, we could craft the following query:

```
site:.example.com (inurl:/signin OR inurl:/login OR inurl:/register)
```

This search query would essentially match all indexed login portals under the target domain.

We're starting to grab more of the basics here, it will get more interesting from now on as we will start to come across more suspicious indexed files and directories.

Filtering by file extension

Google search is quite powerful and also allows us to filter by file extensions. We can leverage this to filter for interesting file types such as PDF files (think of invoices, order receipts, etc.):

```
site:.example.com ext:pdf
```

Filtering by file type

Just like we can search for file extensions, we can also filter by file type (MIME type). With this filter, you'd match every PHP file (even if it doesn't have a valid file extension):

```
site:.example.com filetype:php
```

Filtering by date (finding outdated or legacy assets)

Outdated files are more prone to vulnerabilities, we can take advantage of this last search operator "before" to filter by date before the file was indexed.

The before operator takes the following format: `YYYY-MM-DD`. Let's take a look at an example:

```
site:.example.com before:YYYY-MM-DD
```

More use cases

So we've covered all the major search operators that you need to be able to find anything interesting on Google.

Below are some more common use cases where we try to incorporate multiple operators to really narrow down our search and only pull the interesting results that we want.

Find subdomains

```
site:.example.com -site:www.example.com
```

Return indexed results linked to `*.example.com` but exclude `www.example.com`.

Find query parameters

```
site:.example.com inurl:? || inurl:&
```

Return results linked to `*.example.com` but only if it contains the `?` or `&` characters in the URL.

Find PDF invoices or receipts

```
site:.example.com "invoice" "receipt" ext:pdf
```

Return results linked to `*.example.com` but only if it includes the keywords `invoice` or `receipt` and it must have the `.pdf` file extension.

TIP! Most software companies make use of cloud buckets (such as AWS S3)! Make sure to check these as well!

Find outdated content

```
site:.example.com before:2015-01-01
```

Return results linked to `*.example.com` but only if the publish date was before `2015-01-01`.

Conclusion

Google dorking can provide you as a bug bounty hunter with an edge and help you discover more content and expand your attack surface. However, you must know what you're looking for and narrow down your searches to avoid browsing through static content.

You're now capable of using Google (or any other search engine) to your advantage to find interesting indexed results! Why not start hunting today on a new program on Intigriti with your newly acquired skill set? Browse through our list of programs and start hacking today:

https://intigriti.com/programs