Lab: Practicing Google Dorking Commands

Estimated time: 20 minutes

Learning objectives

- Apply Google Dorking commands in a pen-testing scenario
- Analyze how different Google Dorking commands contribute to gathering information

Introduction

This activity involves tasks that will help you solidify your understanding of Google Dorking commands and how they contribute to uncovering vulnerabilities and hidden data.

Note: It is recommended that you use **Google Chrome** for this activity. Type the commands directly into your browser's search bar to complete each task.

Task 1: Finding historical content

Objective: To identify outdated technologies or practices that may present vulnerabilities

- 1. Launch your web browser and navigate to a search engine (for example, Google).
- 2. Ensure the search bar is empty.
- 3. Type the following command in the search bar:

1. 1

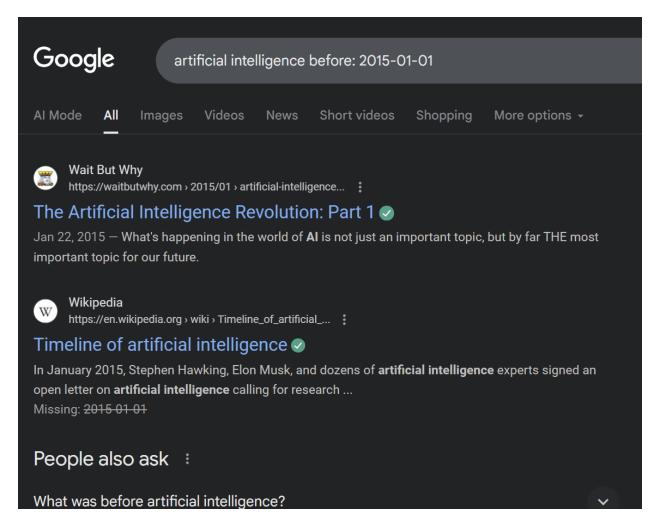
1. artificial intelligence before:2015-01-01

Copied!Wrap Toggled!

copied:Wiap loggied

- 4. Press Enter.
- 5. Examine the articles on artificial intelligence published prior to January 1, 2015.

What to look for: Older articles referencing outdated Al libraries or technologies still in use, which could be vulnerable.



Task 2: Locating file types

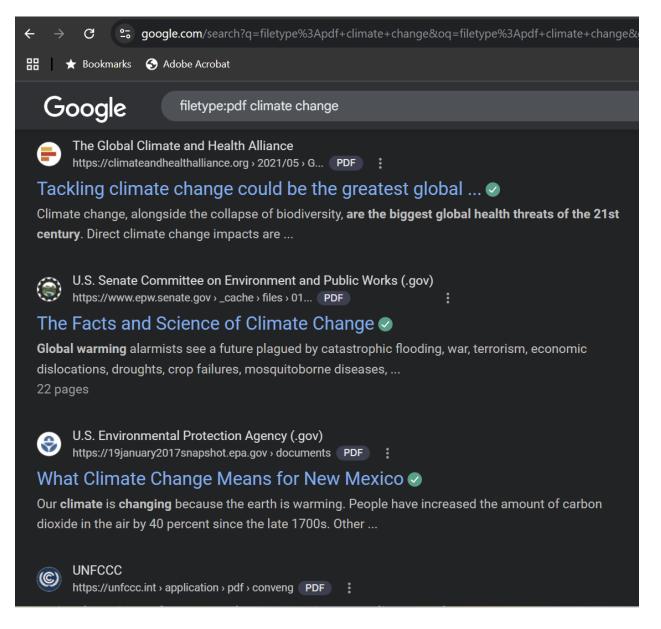
Objective: To find specific file types that might contain sensitive information

- 1. Ensure the search bar is empty.
- Type the following command in the search bar:

```
1. 1
1. filetype:pdf climate change
Copied!Wrap Toggled!
```

- 3. Press Enter.
- 4. Review the list of PDF documents related to climate change.

What to look for: PDF documents that might include confidential reports or internal data relevant to the target.



Task 3: Searching within text

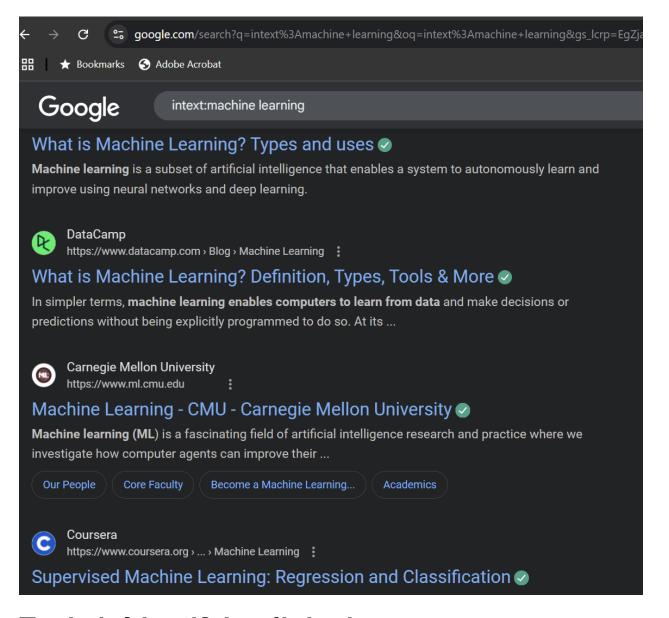
Objective: To identify web pages that discuss certain technologies or topics related to the target

- 1. Ensure the search bar is empty.
- 2. Type the following command in the search bar:

```
1. 1
1. intext:machine learning
Copied!Wrap Toggled!
```

- 3. Press Enter.
- 4. Review the web pages that mention **machine learning** in the body text.

What to look for: Web pages that discuss specific technologies or implementations that could indicate potential weaknesses.



Task 4: Identifying linked pages

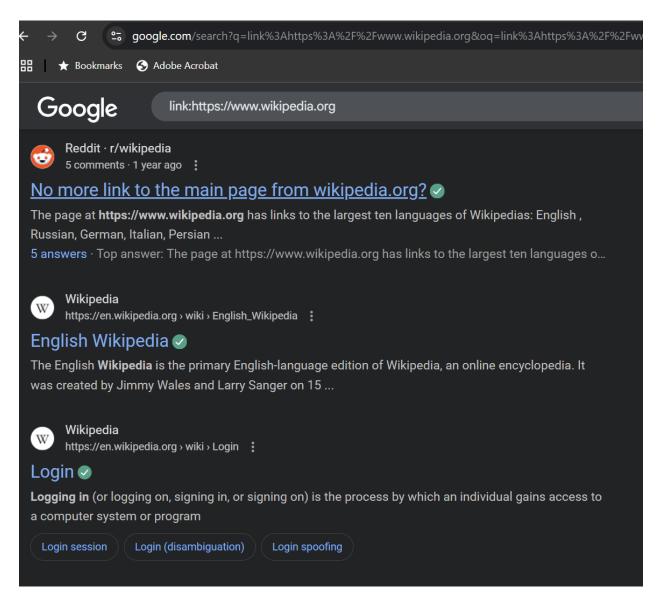
Objective: To find other sites that link to the target and could offer additional insights or entry points

- 1. Ensure the search bar is empty.
- 2. Type the following command in the search bar:

```
1. 1
1. link:https://www.wikipedia.org
Copied!Wrap Toggled!
```

- Press Enter.
- 4. Review the list of web pages that contain links to Wikipedia.

What to look for: External sites linking to the target; these might provide additional information or be potential points of compromise.



Task 5: Exploring anchor text

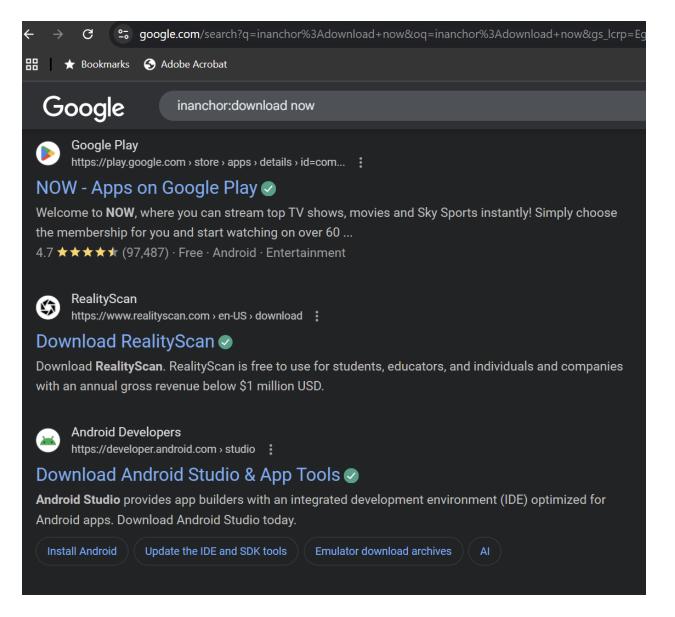
Objective: To find pages that use specific anchor texts, which might lead to sensitive or interesting information.

- 1. Ensure the search bar is empty.
- 2. Type the following command in the search bar:

```
1. 1
1. inanchor:download now
Copied!Wrap Toggled!
```

- 3. Press Enter.
- 4. Review the web pages with links containing **Download now** as the anchor text.

What to look for: Links with specific anchor texts that might lead to downloadable content or resources.



Task 6: Using wildcards

Objective: To capture pages with various terms that might reveal varied information.

- 1. Ensure the search bar is empty.
- 2. Type the following command in the search bar:

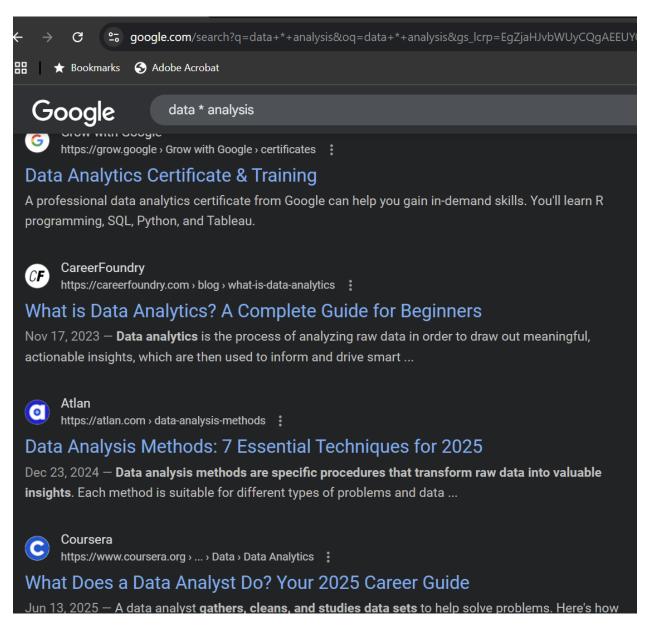
```
1. 1

1. data * analysis

Copied!Wrap Toggled!
```

- 3. Press Enter.
- Review the searches that capture different terms between data and analysis.

What to look for: Analyze the various terms and contexts between "data" and "analysis" to identify relevant content.



Task 7: Excluding terms

Objective: To find relevant information while excluding typical, irrelevant results.

- 1. Ensure the search bar is empty.
- 2. Type the following command in the search bar:

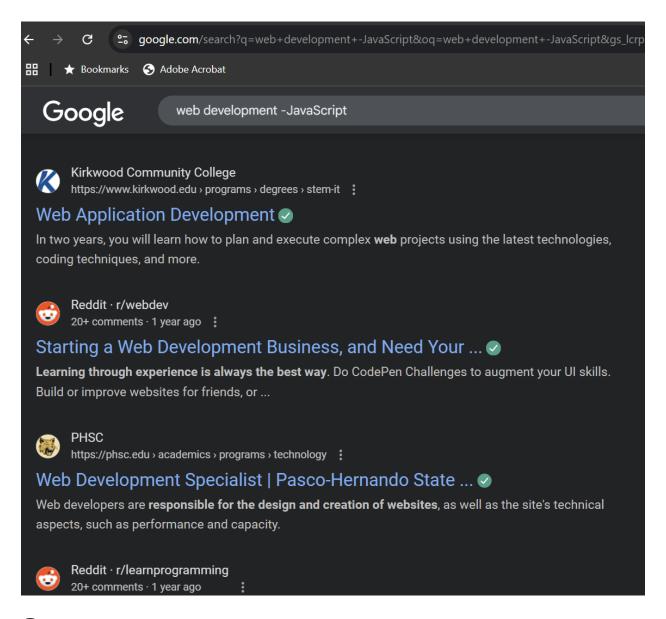
```
1. 1

1. web development -JavaScript

Copied!Wrap Toggled!
```

- 3. Press Enter.
- 4. Review the articles on web development that do not include the term JavaScript.

What to look for: Identify web development content that excludes "JavaScript," which might reveal alternative technologies or practices.



Summary

In this lab, you learned:

- How to apply Google Dorking commands in a pen-testing scenario
- The role of various Google Dorking commands in gathering critical information