TryHackMe Lab Report

Name: Pelumi Johnson
Date: 11/29/2025
Course / Platform: TryHackMe
Room: Walkthrough

Objective

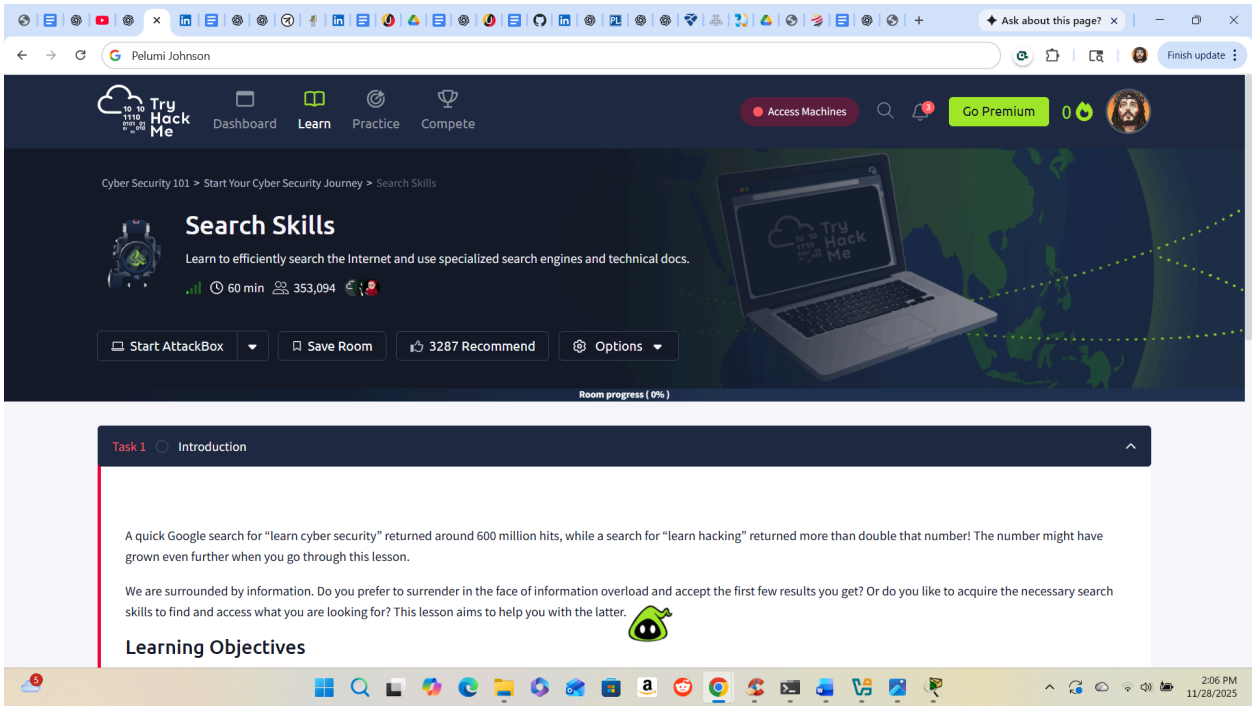
To build strong search techniques for cybersecurity by learning how to evaluate information sources, use advanced Google operators, explore specialized search engines (Shodan, Censys), and analyze files and breaches using VirusTotal and HaveIBeenPwned.

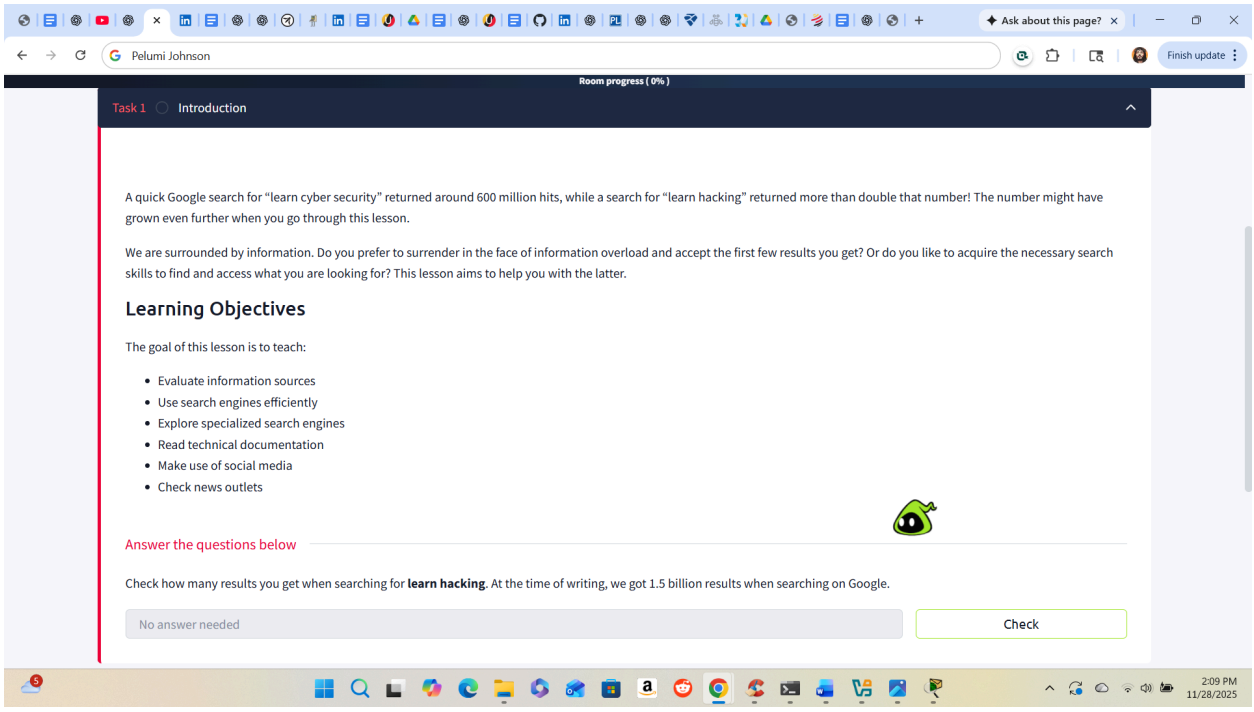
Tools & Platforms Used

- **Google Advanced Search Operators**
- **Shodan** (Internet-connected device search engine)
- **Censys** (Internet asset discovery engine)
- **VirusTotal** (File & URL malware analysis)
- **HaveIBeenPwned (HIBP)** (Breach lookup service)

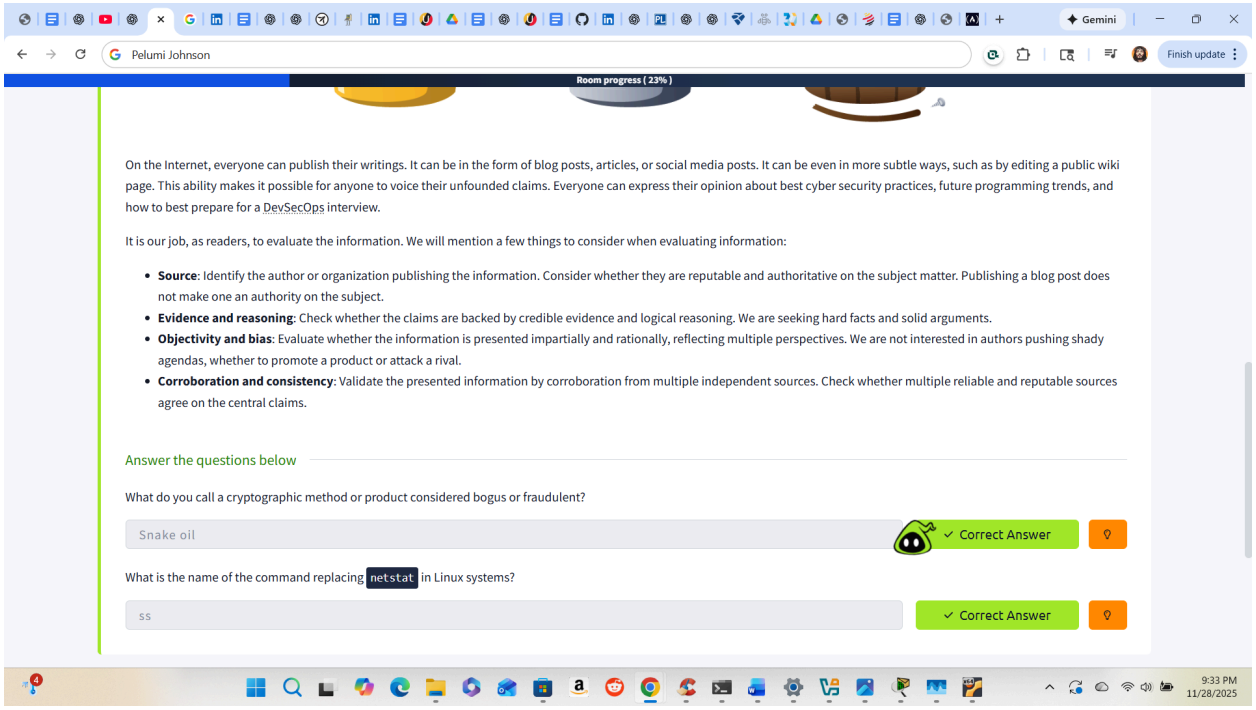
Step 1:

The introduction explains that the internet has more information than the average person can sort through, which makes it easy to get overwhelmed or misled. It highlights the need for intentional searching skills to find reliable and accurate results instead of relying on whatever appears first. This section also outlines that the lesson will cover evaluating sources, using search engines more effectively, exploring specialized tools, and gathering trustworthy information online.


Pelumi Johnson
Finish update
Try Hack Me
Dashboard
Learn
Practice
Compete
Access Machines
Go Premium
Cyber Security 101 > Start Your Cyber Security Journey > Search Skills
Search Skills
Learn to efficiently search the Internet and use specialized search engines and technical docs.
60 min
353,094
Start AttackBox
Save Room
3287 Recommend
Options
Room progress (0%)
Task 1
Introduction
A quick Google search for “learn cyber security” returned around 600 million hits, while a search for “learn hacking” returned more than double that number! The number might have grown even further when you go through this lesson.
We are surrounded by information. Do you prefer to surrender in the face of information overload and accept the first few results you get? Or do you like to acquire the necessary search skills to find and access what you are looking for? This lesson aims to help you with the latter.
Learning Objectives
2:06 PM
11/28/2025


Pelumi Johnson
Finish update
Room progress (0%)
Task 1
Introduction
A quick Google search for “learn cyber security” returned around 600 million hits, while a search for “learn hacking” returned more than double that number! The number might have grown even further when you go through this lesson.
We are surrounded by information. Do you prefer to surrender in the face of information overload and accept the first few results you get? Or do you like to acquire the necessary search skills to find and access what you are looking for? This lesson aims to help you with the latter.
Learning Objectives
The goal of this lesson is to teach:
Evaluate information sources
Use search engines efficiently
Explore specialized search engines
Read technical documentation
Make use of social media
Check news outlets
Answer the questions below
Check how many results you get when searching for learn hacking. At the time of writing, we got 1.5 billion results when searching on Google.
No answer needed
Check
2:09 PM
11/28/2025

- I learned that cybersecurity requires strong search skills because the internet is full of overwhelming and unreliable information.
- The section emphasized that knowing *how* to search is just as important as the information itself.
- I gained an understanding of why evaluating credibility, checking sources, and using precise search techniques are essential for finding accurate, trustworthy results.



On the Internet, everyone can publish their writings. It can be in the form of blog posts, articles, or social media posts. It can be even in more subtle ways, such as by editing a public wiki page. This ability makes it possible for anyone to voice their unfounded claims. Everyone can express their opinion about best cybersecurity practices, future programming trends, and how to best prepare for a DevSecOps interview.

It is our job, as readers, to evaluate the information. We will mention a few things to consider when evaluating information:

- **Source:** Identify the author or organization publishing the information. Consider whether they are reputable and authoritative on the subject matter. Publishing a

blog post does not make one an authority on the subject.

- **Evidence and reasoning:** Check whether the claims are backed by credible evidence and logical reasoning. We are seeking hard facts and solid arguments.

- **Objectivity and bias:** Evaluate whether the information is presented impartially and rationally, reflecting multiple perspectives. We are not interested in authors pushing shady agendas, whether to promote a product or attack a rival.

- **Corroboration and consistency:** Validate the presented information by corroboration from multiple independent sources. Check whether multiple reliable and reputable sources agree on the central claims.

Additional Information:
During this portion of the lesson, I learned two key security-related terms:

- **Snake oil** is a term used to describe a cryptographic method or product that is considered fake, unreliable, or fraudulent.

- `ss` is the modern Linux command that replaces the older `netstat` command. It provides faster and more detailed information about network sockets and connections.

Step 2

Google Search Operators

Google supports several special search operators that help narrow down and refine results:

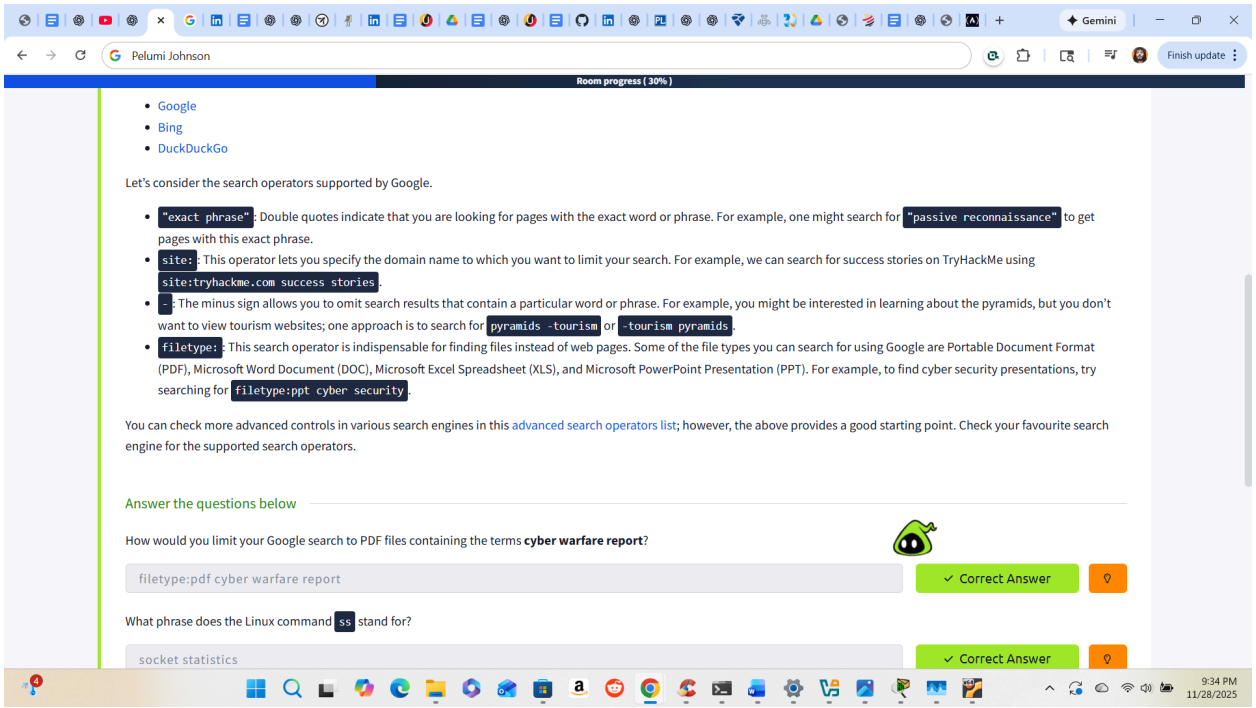
- “exact phrase”: Using double quotes searches for an exact word or phrase.
  Example: “`passive reconnaissance`”

- site: Limits results to a specific website or domain.
  Example: `site:tryhackme.com success stories`

- - (minus): Removes pages containing certain words.
  Example: `pyramids -tourism`

- filetype: Finds specific file formats such as PDF, DOC, XLS, or PPT.
  Example: `filetype:ppt cyber security`

These operators are powerful when you're trying to filter through large amounts of information and find precise content.



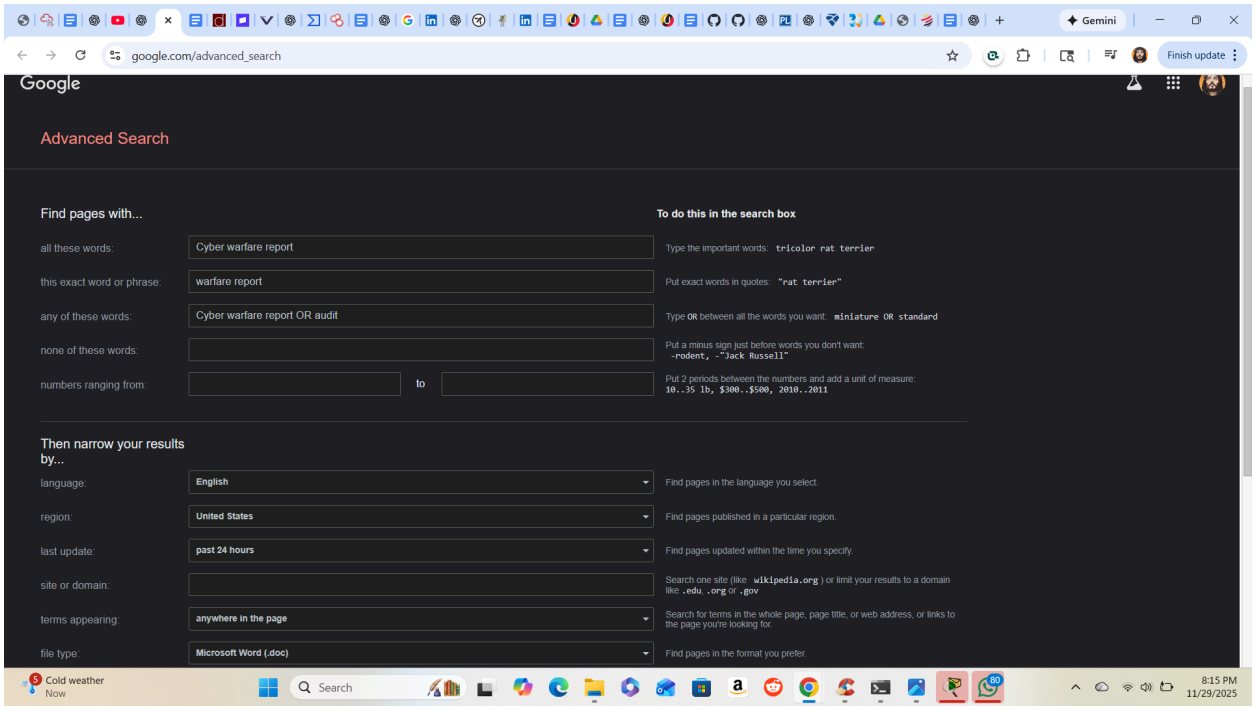
Questions + Explained Answers

1. filetype:pdf cyber warfare report

This search uses Google's filetype operator to show only **PDF** documents that contain the words "cyber warfare report."

2. ss (socket statistics)

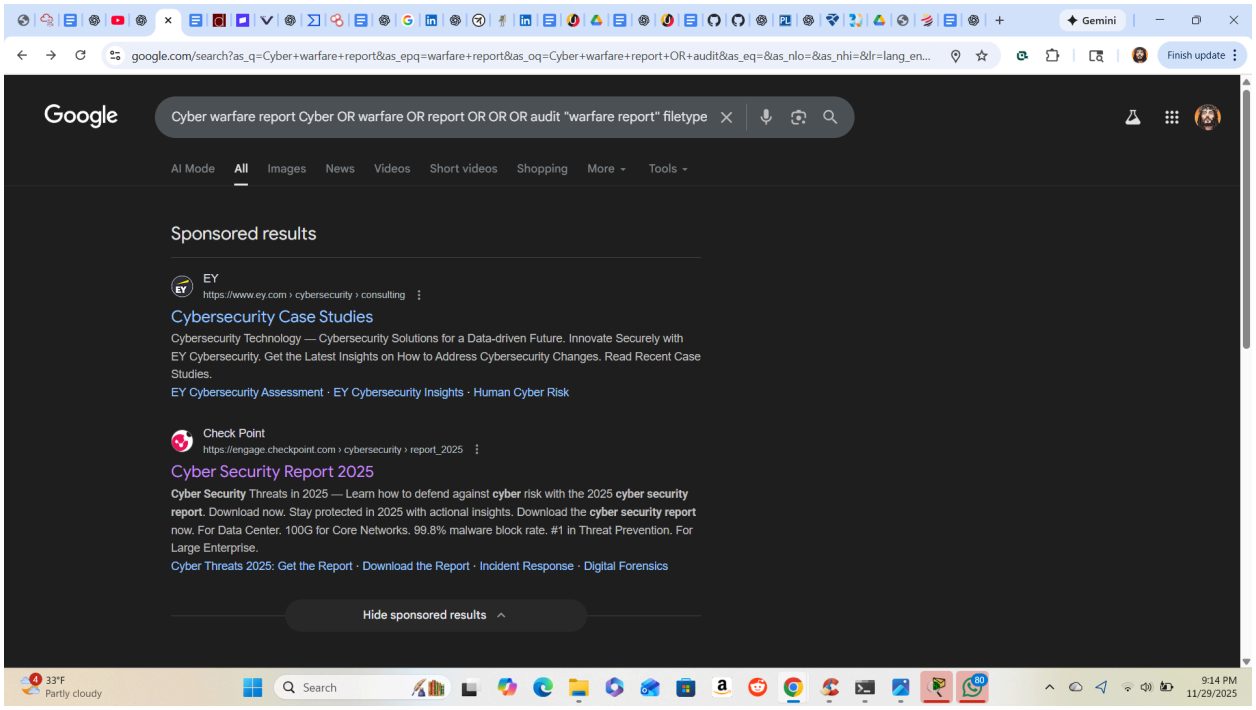
The Linux **ss** command stands for **socket statistics**, and it shows information about network connections and ports.



My Personal Practice Using Google Advanced Search

I explored the Google Advanced Search page to practice combining operators with visual controls. Here is what I tried:

- Searched for pages containing the words “cyber warfare report”
- Searched for the exact phrase “warfare report”
- Used OR logic with: *Cyber warfare report OR audit*
- Left the “none of these words” field empty
- Set custom filters like:
  - Language: English
  - Region: United States
  - Last update: past 24 hours
  - File type: Microsoft Word (.doc)

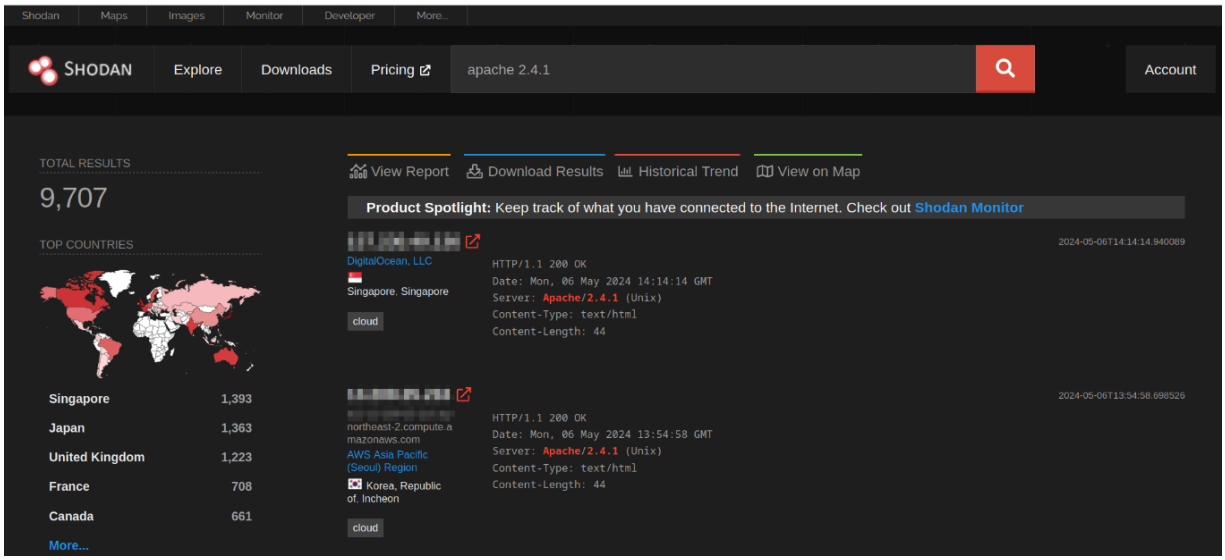


The search results displayed are based on combining several Google search operators such as filetype, OR, and exact phrase. Because the query included multiple variations of "cyber warfare report" and logical OR statements, Google returned a wide set of cybersecurity-related documents and case studies. The top results are sponsored ads, which means they appear first regardless of your search filters. Below them, Google would normally show actual documents or pages that match the search terms and operators used.

Step 3

- **Shodan** is a specialized search engine that scans the internet for connected devices instead of websites.

  It finds things like servers, webcams, routers, databases, industrial systems, and IoT devices. Shodan lets you search for specific software versions, open ports, and exposed services, making it a powerful tool for cybersecurity analysis and discovering vulnerable systems online.



This page shows the results of searching for “apache 2.4.1” on Shodan, a search engine that scans devices connected to the internet.

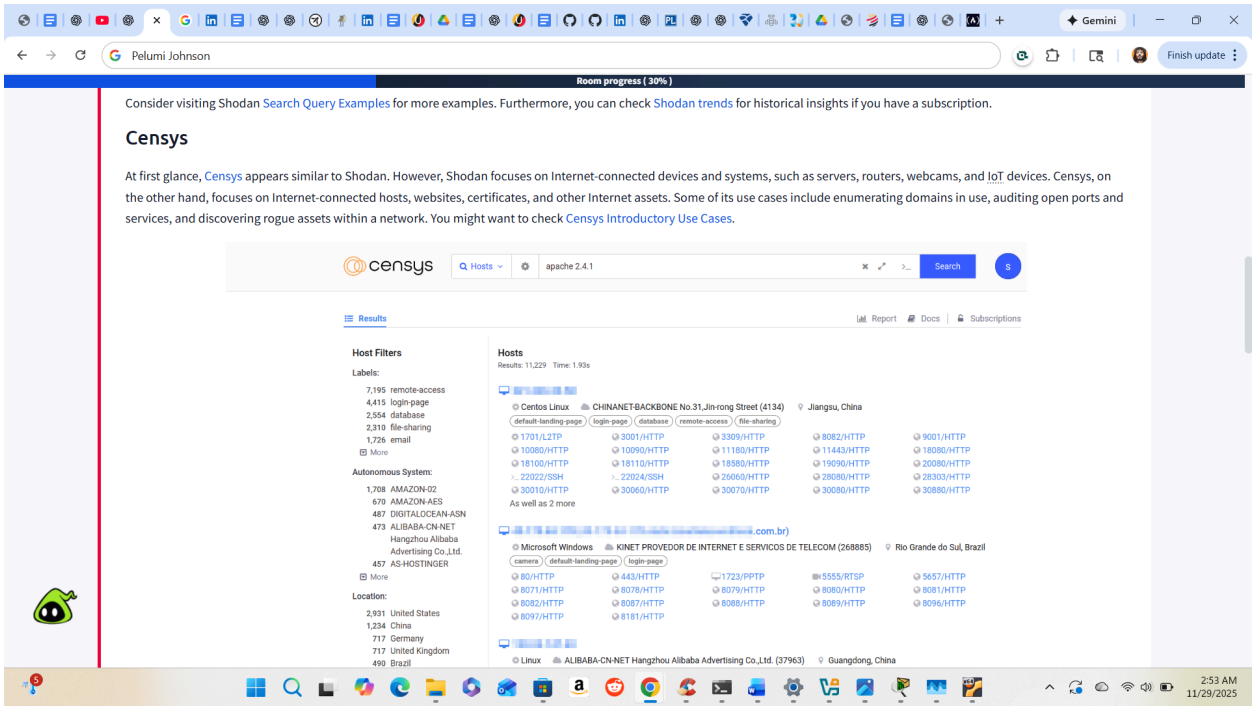
Shodan returns the total number of servers still running this version of Apache and shows which countries they are located in. It also displays details like the server’s IP address, country, and headers identifying the Apache version.

This helps analysts understand how many outdated or vulnerable web servers are still exposed online.

- **Censys** is a specialized search engine that scans and indexes Internet-connected assets, such as websites, servers, certificates, and open ports.

  It allows security analysts to look up hosts and see what services they expose, what software versions they are running, and whether any systems appear misconfigured or vulnerable.

  While Shodan focuses on devices like routers, webcams, and IoT equipment, Censys focuses more on websites, certificates, and infrastructure visibility, making it useful for auditing networks and discovering exposed assets.



This page shows the same Apache version search, but using Censys instead of Shodan.

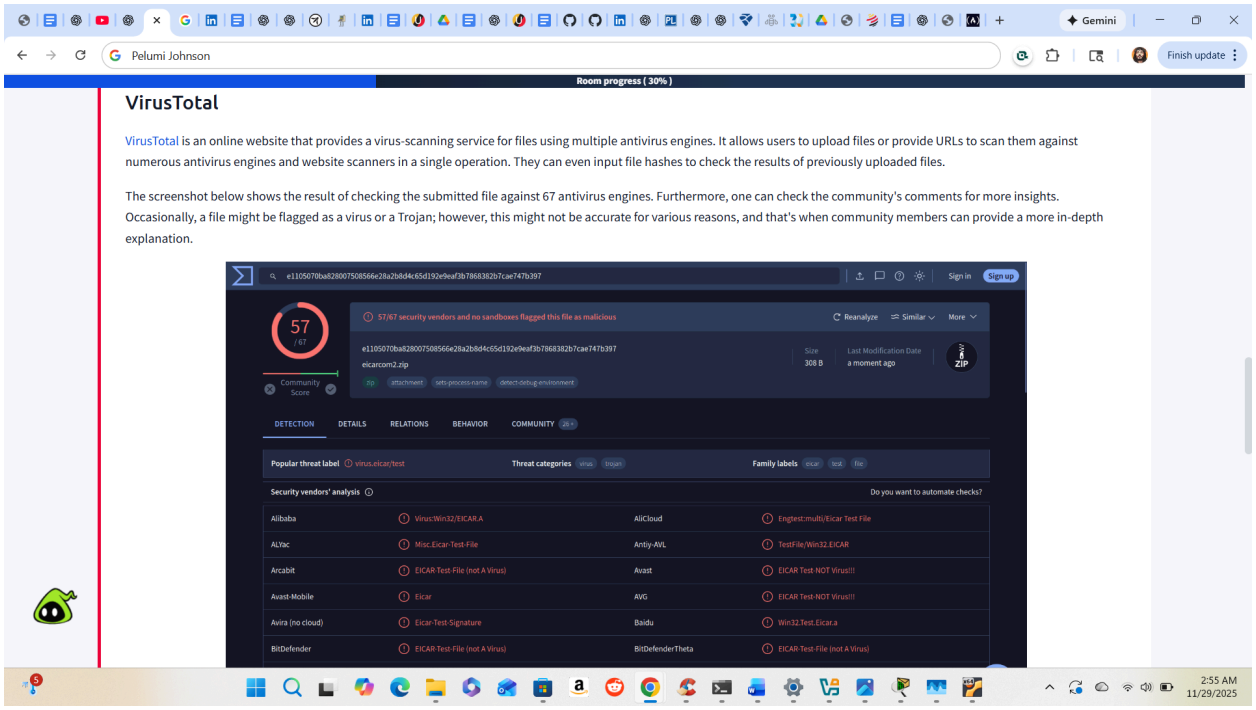
Unlike Shodan, which focuses on internet-connected devices (routers, webcams, servers, IoT), Censys focuses more on websites, hosts, certificates, and services.

The results show hosts using Apache 2.4.1, the types of services running (HTTP, SSH, login pages), and their geographic or network locations.

It's useful for identifying exposed infrastructure and mapping how many systems are still using older software versions.

- **VirusTotal** is an online malware-scanning service that analyzes files, URLs, and hashes using multiple antivirus engines at the same time.

  It allows users to upload a file or enter a link, and VirusTotal will check it against dozens of security vendors to see whether it is flagged as malicious. The platform also shows detection names, file behavior, and community comments, helping users quickly determine if a file is safe or suspicious.

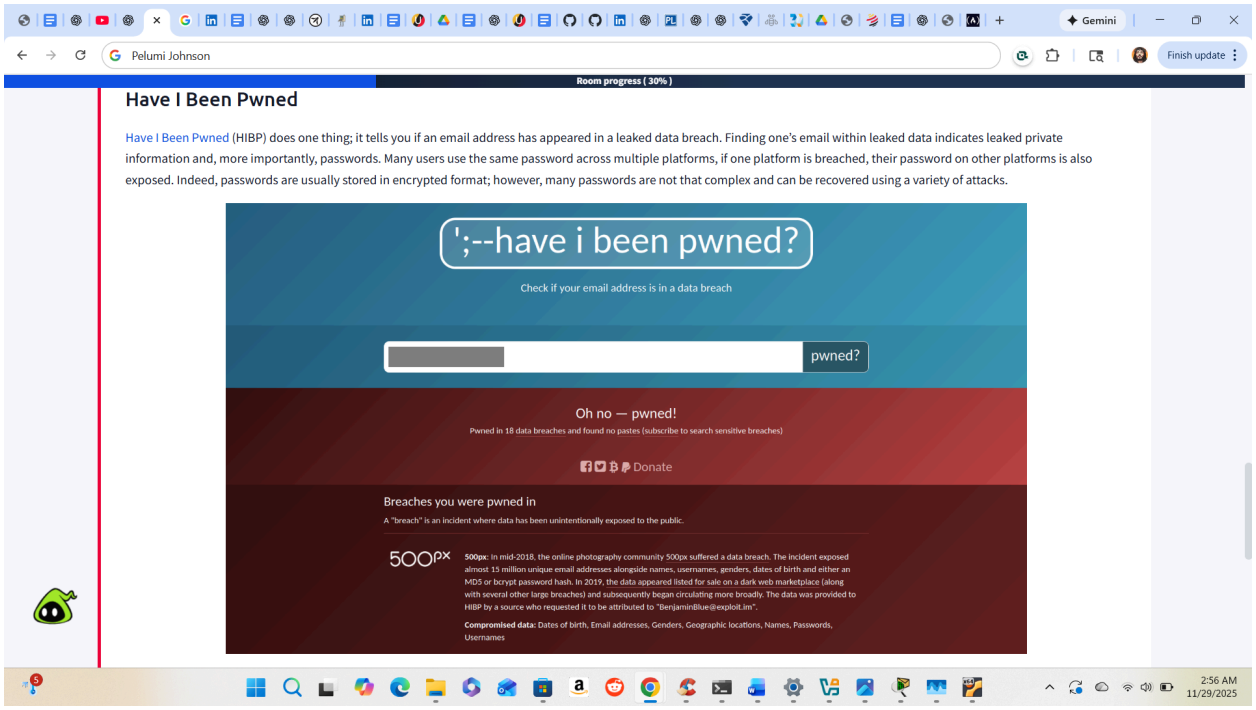


This page is from **VirusTotal**, which scans files using multiple antivirus engines.

The screenshot shows the result of uploading the EICAR test file, a harmless file used for testing antivirus detection. VirusTotal scanned it with 67 antivirus engines, and **57 flagged it as malicious**, which is expected. You can also see threat names, family labels, behavior analysis, and which security vendors detected it. VirusTotal helps security analysts quickly check whether a file is malware and how different AV engines classify it.

- **Have I Been Pwned (HIBP)**

  Have I Been Pwned is an online service that lets you check whether your email address or passwords have been exposed in a data breach. It collects and analyzes leaked data from hacked websites and notifies users if their information appears in any breach, helping them stay aware of security risks and protect their accounts.



This page shows the website **Have I Been Pwned**, which checks whether an email address appears in a leaked data breach.

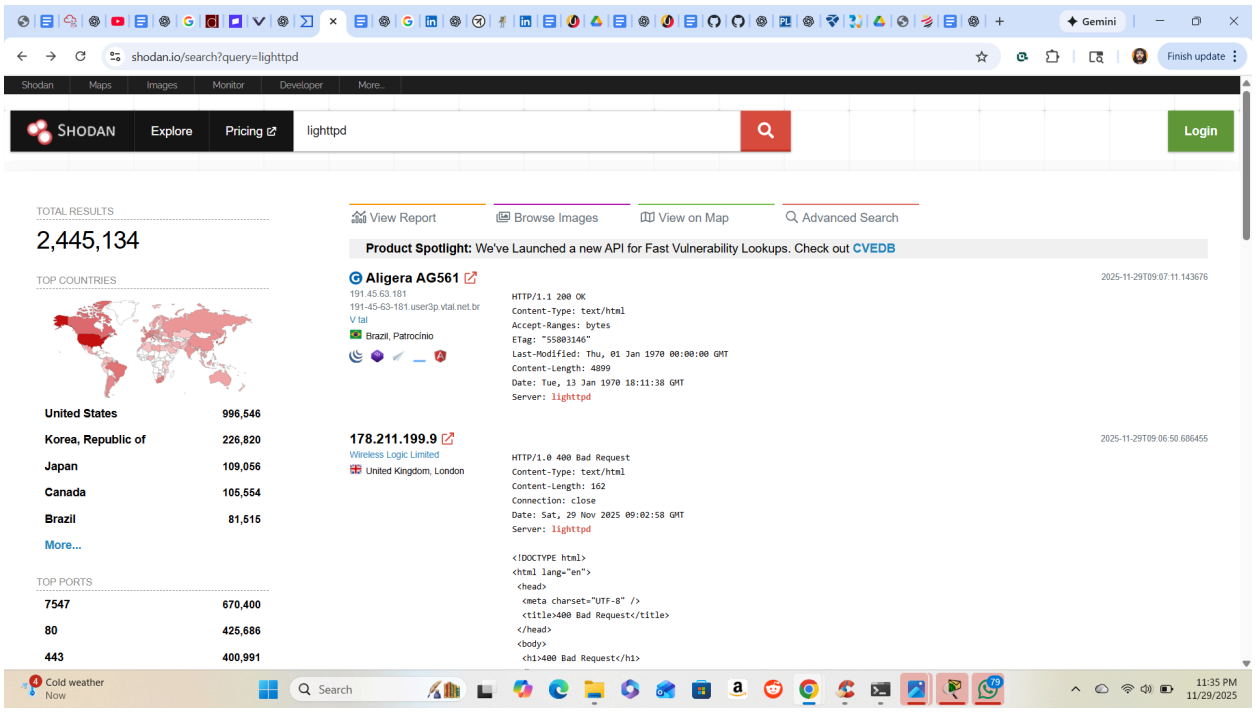
After typing an email address, the site lists the breaches where that email was exposed, along with details like the type of data leaked (passwords, addresses, usernames, etc.).

This helps users understand if their information has been compromised and whether they should update their passwords or enable stronger security.

Practical Exercise Summary

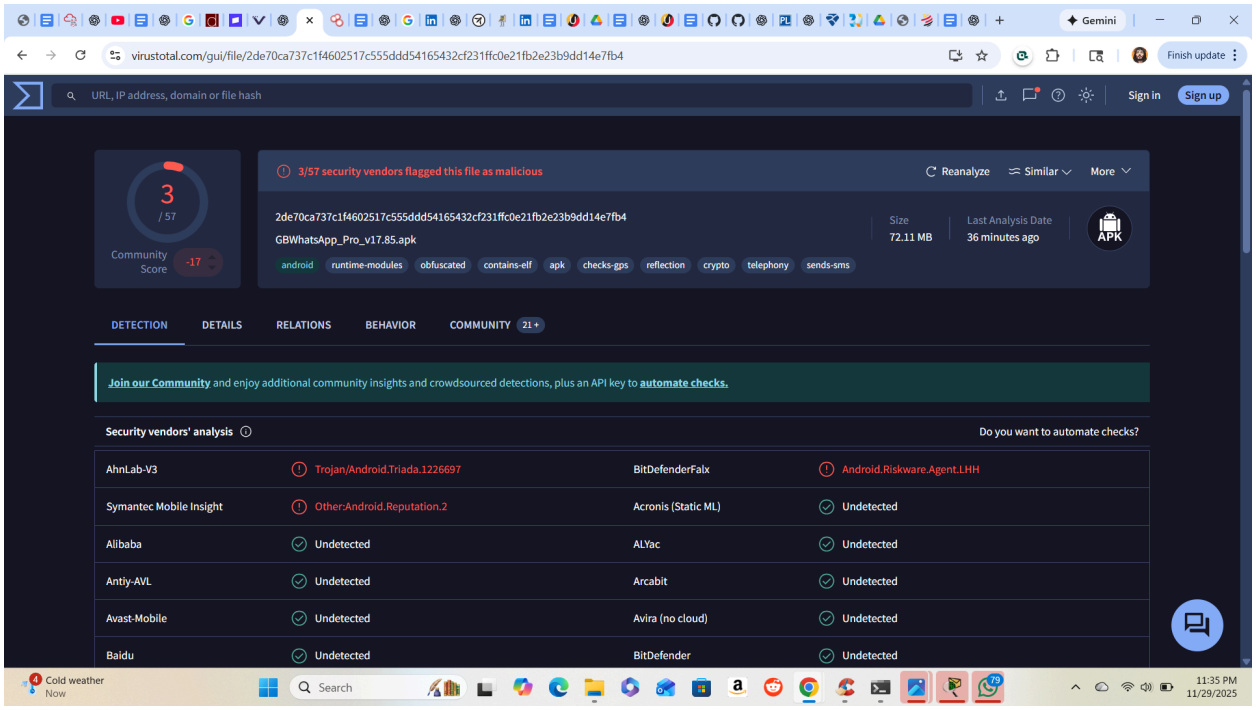
- For the hands-on part, I used **Shodan** to search for the web server **lighttpd**. The search returned over 2 million results and showed where these servers are located around the world. The United States had the highest number of lighttpd servers, followed by the Republic of Korea and Japan.

  I also clicked into individual results to see the information exposed by Shodan, such as the server's IP address, location, HTTP response, headers, and the "Server: lighttpd" banner. This practice helped me understand how Shodan identifies internet-connected systems and how attackers or analysts can use it to find exposed services.



- For the second exercise, I analyzed a file using **VirusTotal** by submitting its hash. VirusTotal scanned the file through 57 antivirus engines, and a few security vendors marked it as malicious. One of the detections I observed came from **BitDefenderFalx**, which identified the file as **Android.Riskware.Agent.LHH.**

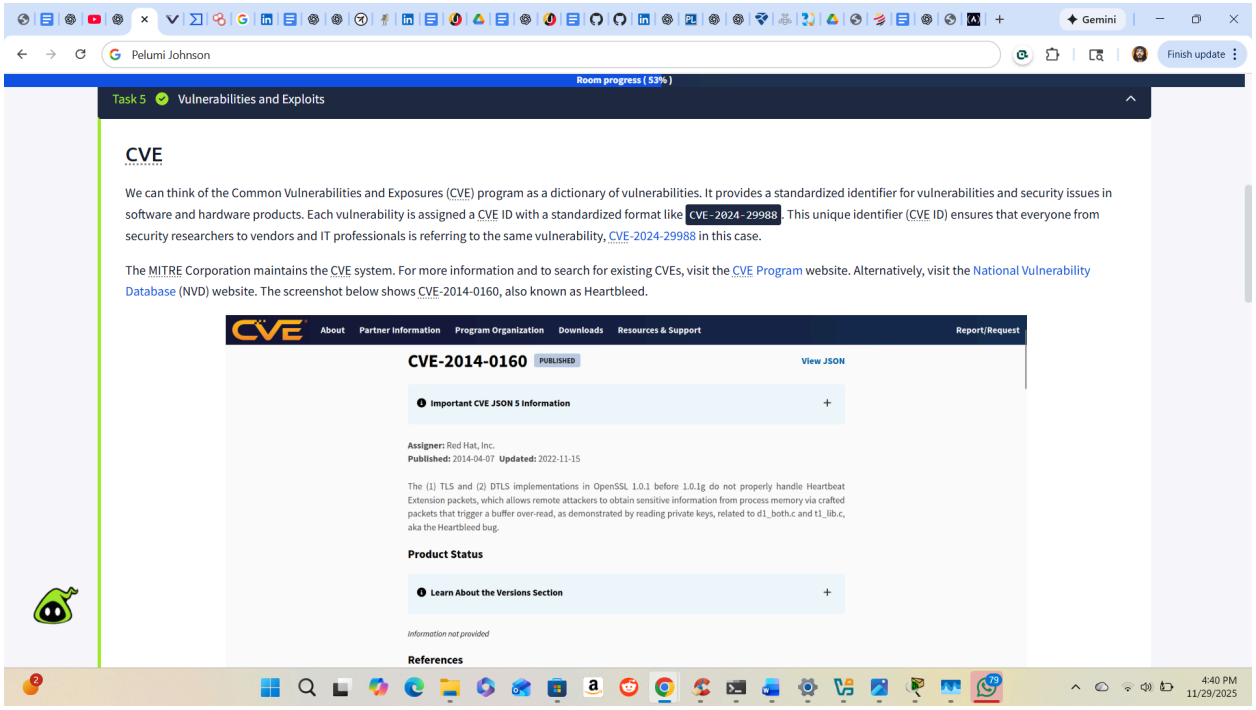
  I reviewed the detection list to compare which engines flagged it and which ones didn't. This demonstrated how VirusTotal uses multiple engines to cross-check a file's safety and helps analysts confirm whether a file might be harmful.



Step 4

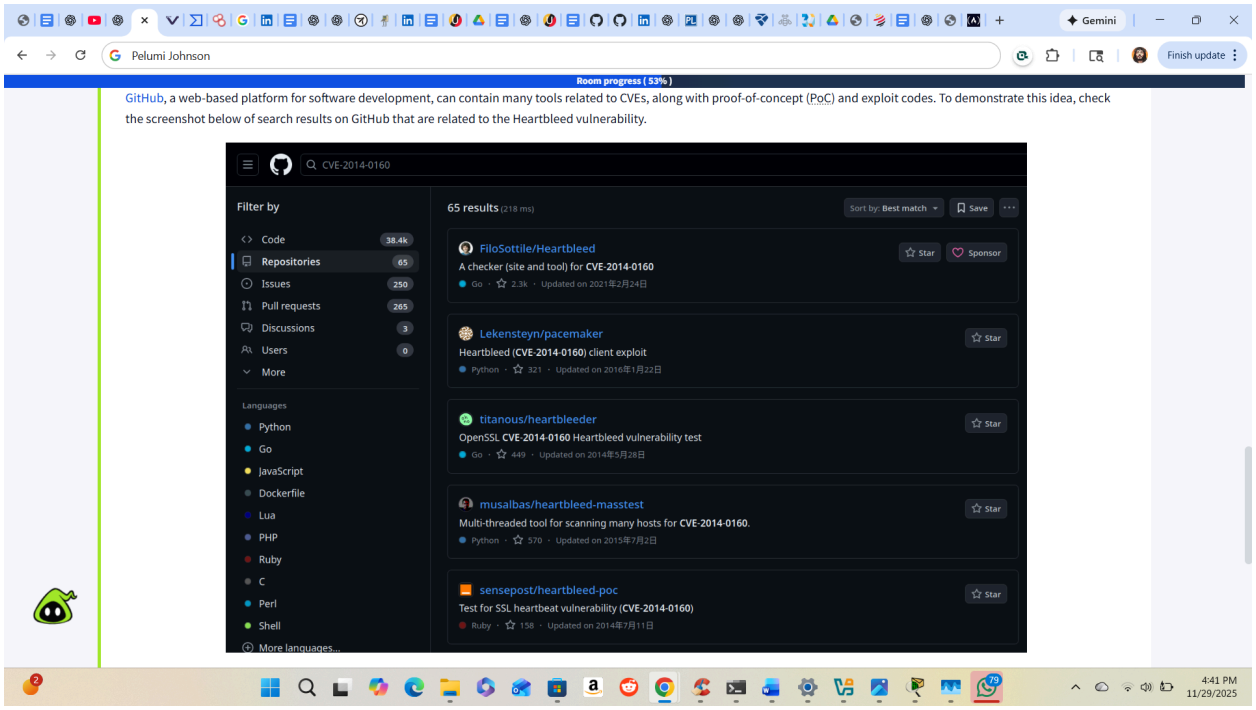
- **CVE (MITRE CVE System):**

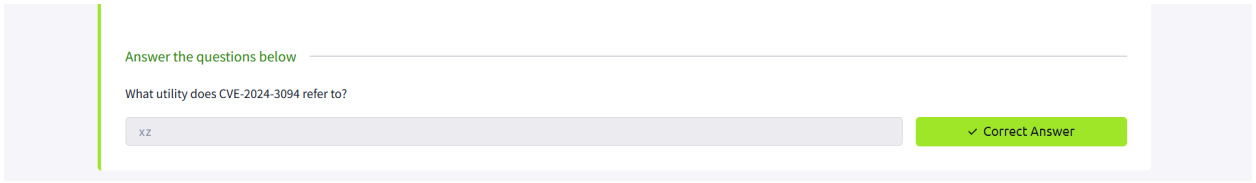
This page introduces CVE (Common Vulnerabilities and Exposures), which works like a dictionary of known security flaws. Each vulnerability receives a unique ID so that security teams, vendors, and researchers all refer to the same issue using the same label. The screenshot shows CVE-2014-0160, also known as *Heartbleed*, a widely publicized OpenSSL vulnerability that allowed attackers to read sensitive information from memory due to improper handling of heartbeat packets.



- **Exploit Database (Exploit-DB)**

  This page demonstrates how the **Exploit Database** is used to find actual exploit code for known vulnerabilities. It lists published exploits from different authors and marks whether they have been verified. In the screenshot, you see several exploit entries connected to the *Heartbleed* vulnerability. The Exploit-DB is a common resource for penetration testers and researchers when testing systems they have permission to assess.

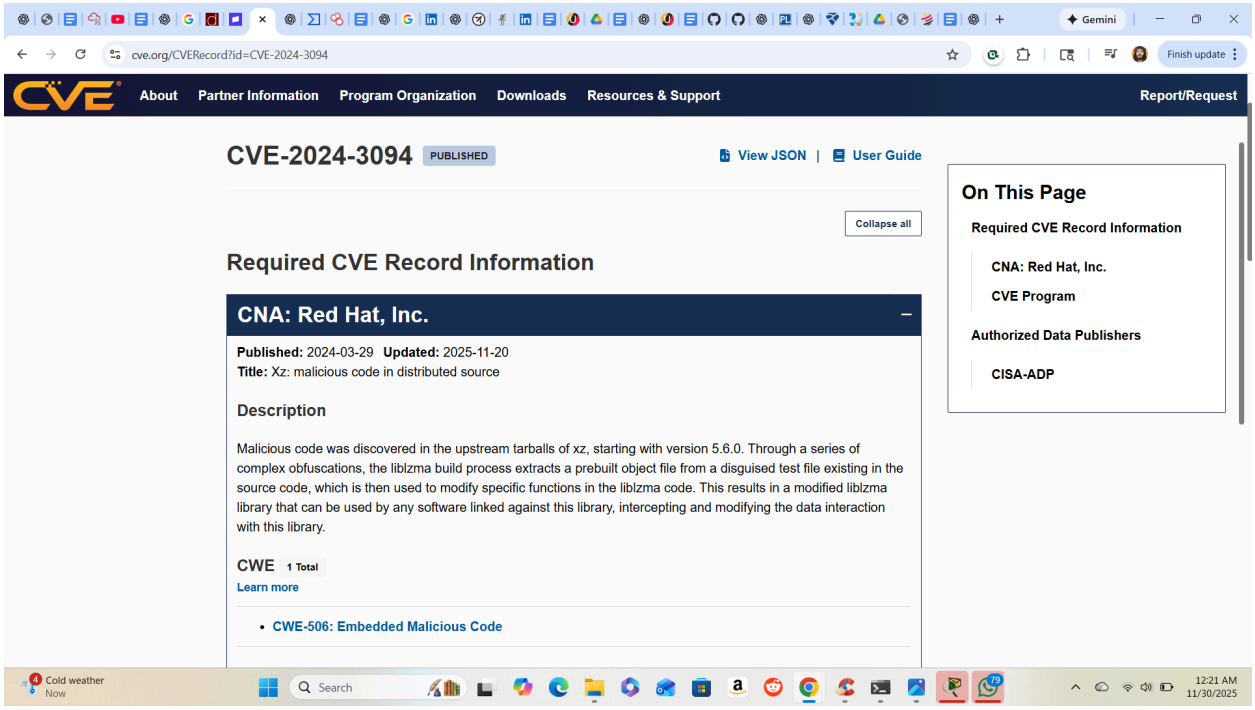




Practical Exercise: CVE-2024-3094 (xz Utility)

For this exercise, I looked up **CVE-2024-3094**, a vulnerability involving the **xz** utility. This CVE describes a case where malicious code was inserted into the upstream source of xz, starting from version 5.6.0. The hidden code altered the **liblzma** library so that attackers could intercept, modify, or influence data processed by any software that used this library.

I confirmed the details using the official CVE page, which showed the description, publisher (Red Hat), impact, and the related CWE category (*Embedded Malicious Code*). This helped reinforce how CVEs document security issues and how to verify them using trusted vulnerability databases.



Below is each technical term explained

1. CVE-2024-3094:

A **CVE** is just an ID number for a security problem.
Think of it like a **case file number** detectives use to track a crime.

- **CVE-2024-3094**: the 3094th vulnerability recorded in 2024.
  It lets everyone talk about the same problem using the same ID.

2. xz Utility:

xz is a **file compression tool** on Linux.

Think of it as a **zip tool** for packing files into a smaller size, like folding clothes neatly into a suitcase.

3. Upstream Source:

Upstream means **the original source code** before it gets sent to users.

Analogy: upstream is like the **original spring of water**.
If the water is poisoned at the spring, everyone downstream drinks poison.

4. Version 5.6.0:

This is the version where the attacker slipped in the malicious code.

5. liblzma library:

A **library** in programming is like a toolbox.

liblzma is a compression toolbox that many programs rely on to shrink/expand data.

If that toolbox is corrupted, every program using it becomes unsafe.

6. Intercept, modify, or influence data:

It means the attacker could:

- **listen** to the data
- **change** the data
- **inject** fake or harmful data

7. Embedded Malicious Code (CWE Category)

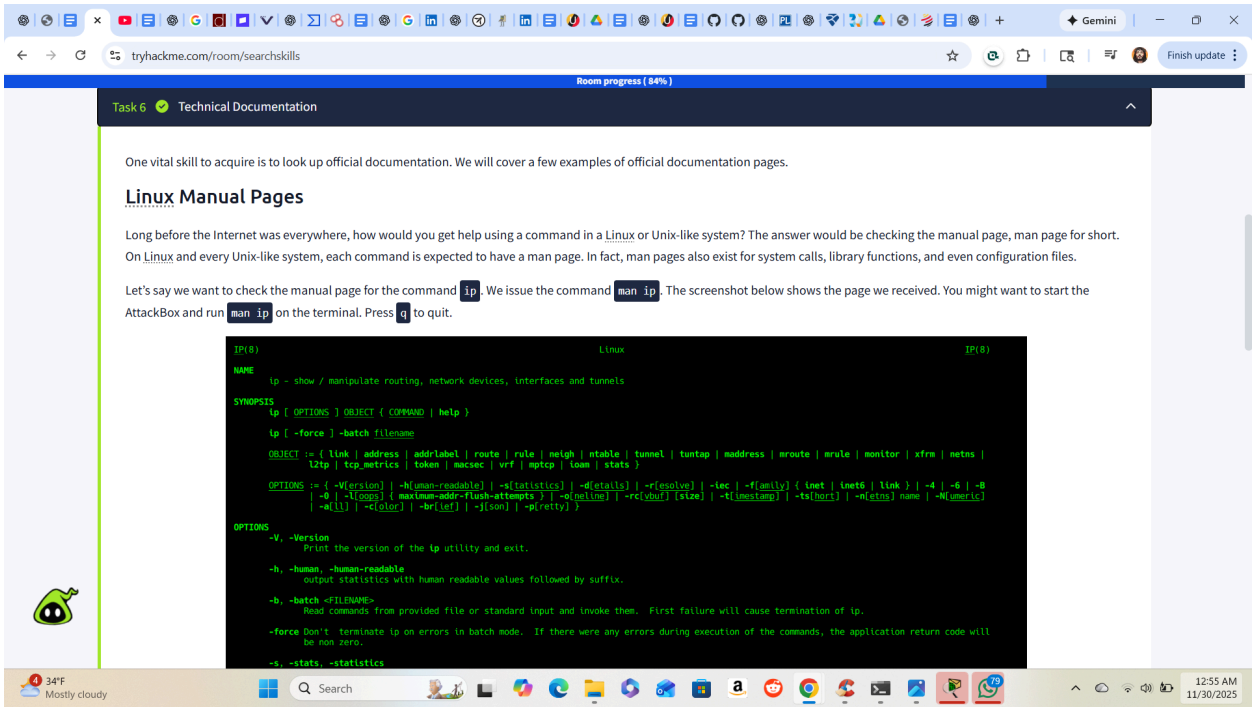
This means the attacker hid harmful code *inside* legitimate code.

8. Vulnerability Database: Meaning These are websites that document weaknesses in software. They are like **public crime reports** that help the whole world stay alert and protected. You used one to verify the details.

Step 5

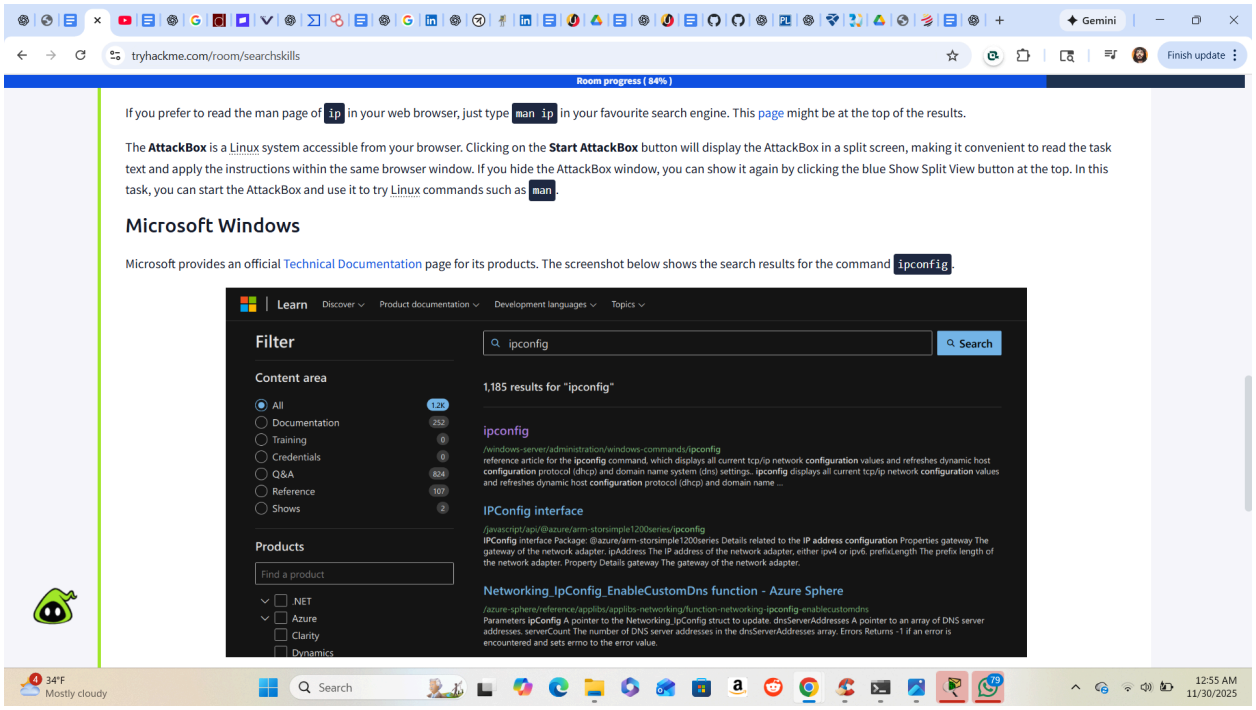
Linux Manual Pages (man ip)

This screenshot shows the Linux *manual page* for the ip command. The man page displays a full description of what the command does, along with its available options. In this case, the ip utility is used for viewing and modifying network settings such as routing, interfaces, and tunnels. The page lists the syntax and flags, giving a complete reference for how the command works. Linux manual pages act as built-in documentation, so a user can quickly learn how a command behaves by running man <command>.



Microsoft Windows Documentation (ipconfig)

This screenshot shows Microsoft's official product documentation website after searching for "ipconfig." It returns articles explaining how the `ipconfig` command displays a system's network configuration. The documentation page provides explanations, examples, and parameters for Windows users. This demonstrates how official vendor documentation remains the most accurate source for understanding system tools.



Product Documentation Questions

Here I answered two documentation-based questions. First, the Linux command `cat` stands for *concatenate*, meaning it reads and displays file content. Second, the Windows `netstat` parameter `-b` is used to show which executable is responsible for each active or listening network connection. These questions reinforce how both Linux and Windows offer documented tools for investigating system activity.

Examples of executables

Windows

- `chrome.exe`
- `cmd.exe`
- `notepad.exe`

Linux

- `/bin/bash`

- `/usr/bin/python3`
- `/usr/sbin/sshd`

Why this matters for `netstat -b`

`netstat -b` shows *which executable (which program)* opened or is listening on a network port.

Example:

- Port 80: opened by `apache2` (web server executable)
- Port 22: opened by `sshd` (SSH server executable)
- Port 443: opened by `chrome.exe` (web browser executable)

So you can see **which program is communicating on the network**.

This is huge in cybersecurity because it tells you:

- what program is using the internet
- what might be suspicious
- what might be malware

When Windows uses the command:

```
netstat -b
```

It shows **which .exe program** is responsible for a network connection.

Example output:

```
TCP   0.0.0.0:80   LISTENING
[apache.exe]
```

This means:

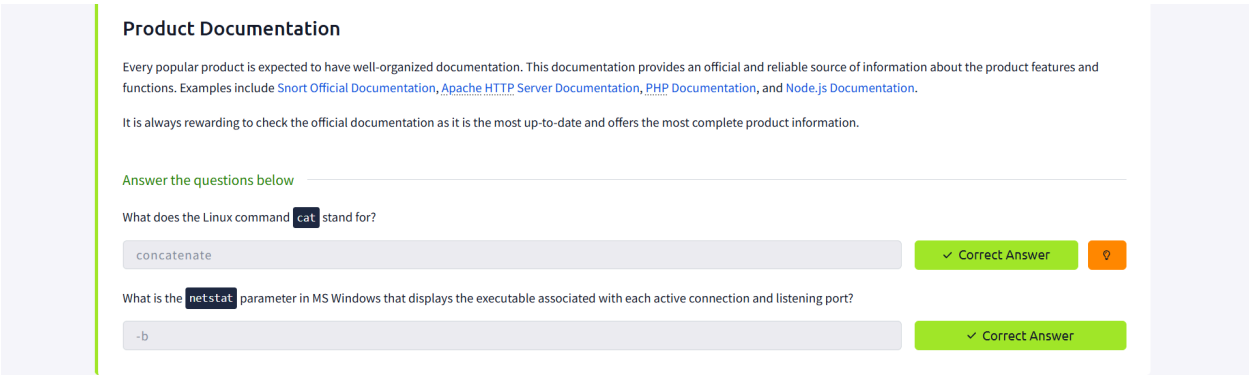
- Port 80 is open
- The executable responsible is `apache.exe`

Another example:

```
TCP    192.168.1.5:443    ESTABLISHED
[chrome.exe]
```

This means:

- Chrome is using port 443 to access a website
- The program is `chrome.exe`



Linux Command: `cat`

To confirm what the `cat` command stands for, I looked up the official Linux manual page. I did this by searching for **"Linux cat man page"**, which brought me to the documentation on die.net. The page clearly stated that **`cat` means "concatenate."**

The documentation explained that the command is used to concatenate files and print their contents to standard output. This verified the correct answer for the exercise question.

Windows Command: `netstat -b`

To understand the `-b` parameter for the Windows **netstat** command, I started by using **Google Search**.
I searched the phrase **"netstat -b Windows meaning**, which led me to the official Microsoft documentation page.

Step 6

Social Media Research

For this task, I explored how social media can play a major role during **reconnaissance**, the first phase of gathering information in cybersecurity. Using **Google**, I looked at how different platforms expose different types of data.

I identified **LinkedIn** as the best place to learn about a person's **technical background**, job history, skills, and certifications. From a security perspective, LinkedIn becomes a valuable reconnaissance source because attackers can map out an employee's role, the technologies they use, and sometimes even what systems a company relies on.

Next, I looked at how someone might answer a **secret question** like Which school did you go to as a child? For that scenario, I chose **Facebook**, because people often share personal details there, old photos, hometowns, childhood schools, and family connections.

This kind of information is exactly what attackers look for during **OSINT (Open-Source Intelligence)** gathering when trying to bypass account security or reset passwords.

LESSONS LEARNED

1. **Be mindful of what you post online:** as a cybersecurity student, I learned that personal details on social media can reveal more than expected, so limiting what I share protects my privacy and security.

2. **Avoid oversharing sensitive personal information:** attackers can use old schools, hometowns, or family details to answer secret questions or reset accounts, so I must think carefully before posting.

3. **Use a separate email when exploring new platforms:** creating a temporary or secondary email helps me test, learn, and research sites safely without exposing my real identity or inbox.

4. **Engage with platforms instead of avoiding them:** cybersecurity requires exposure, not isolation; learning how platforms work strengthens my OSINT skills

and helps me understand real-world attack surfaces.

5. **Recognize social media as a reconnaissance tool:** this exercise taught me how easily attackers gather information during OSINT, reminding me to protect myself and also to analyze platforms from a defender's perspective.