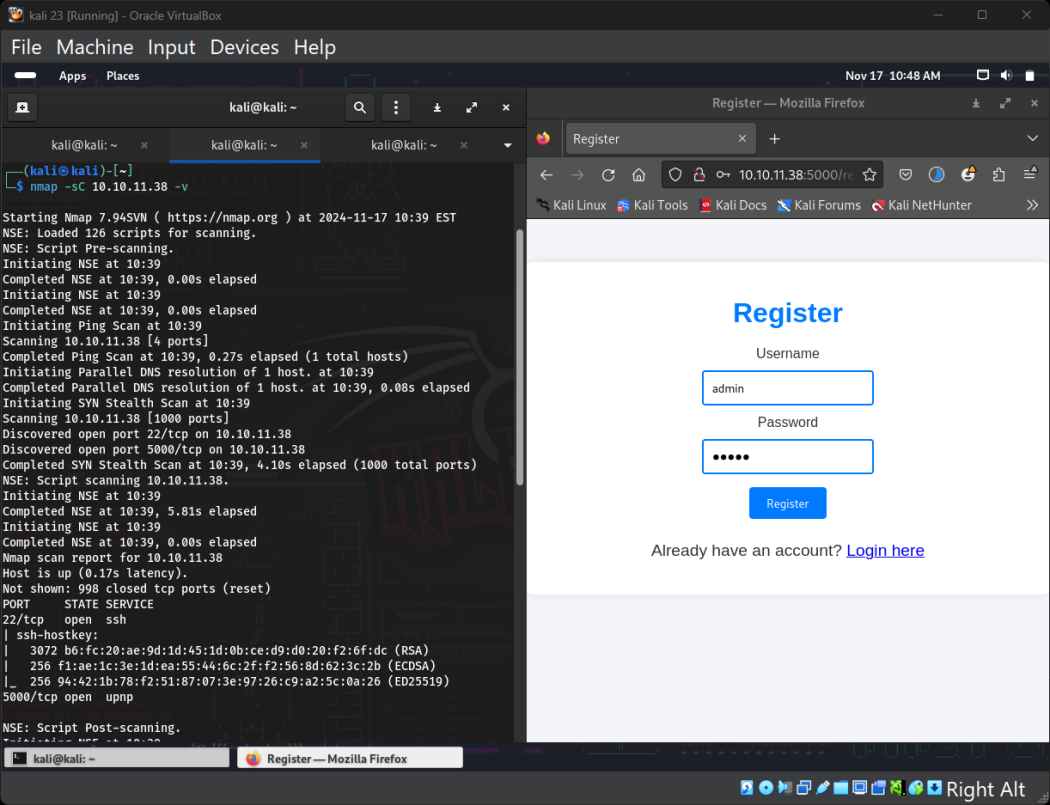
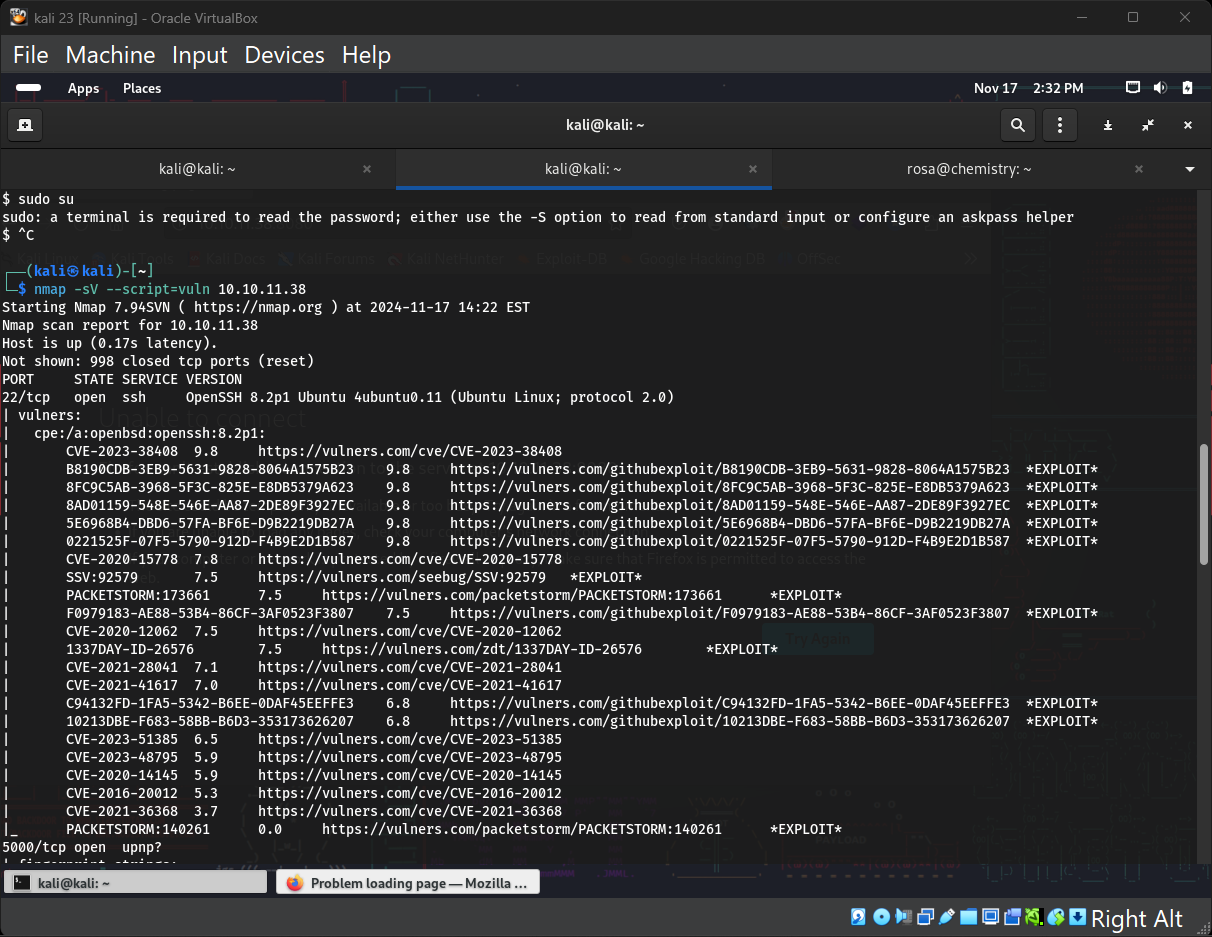
**Findings**

**1. Nmap Scan**

An Nmap scan identified the following open port and service:

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

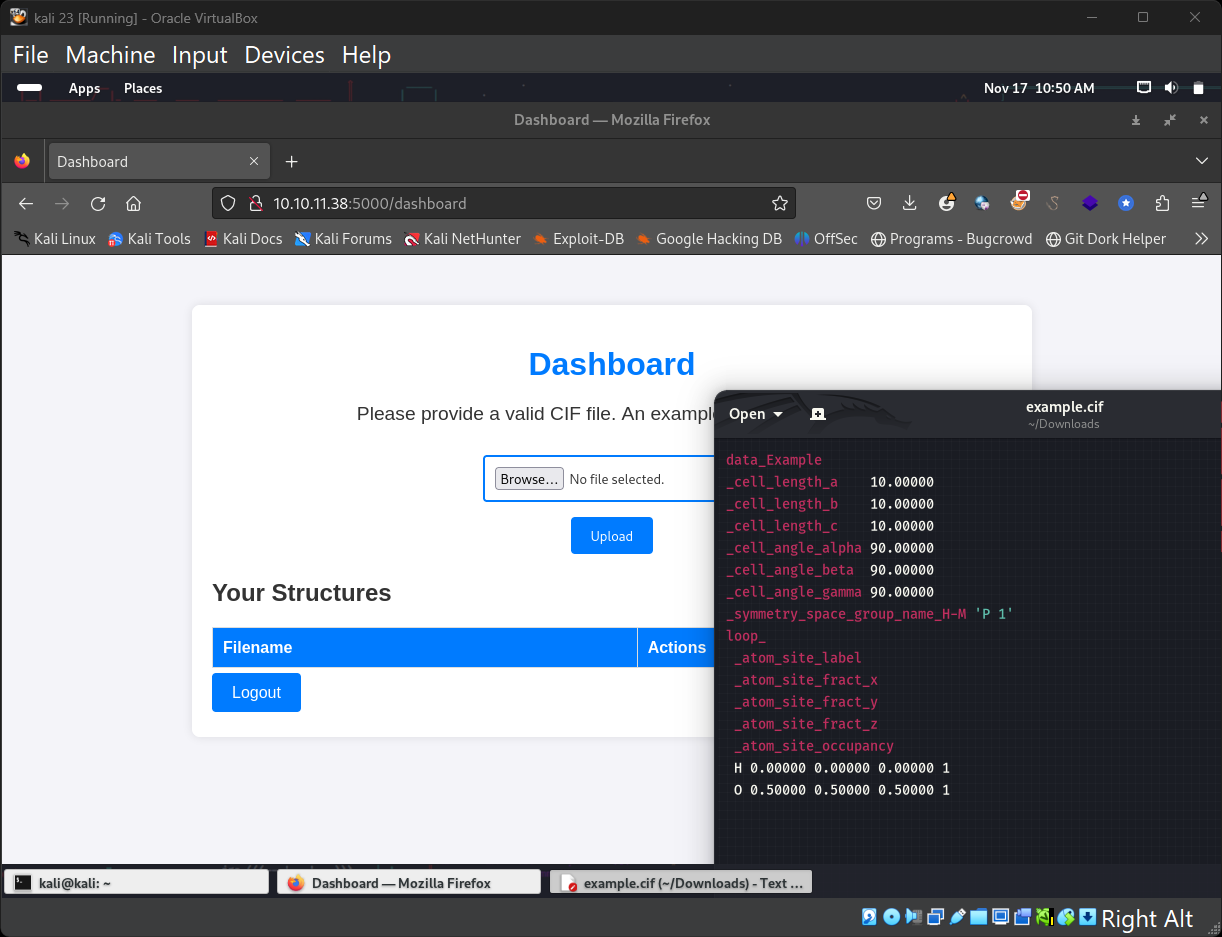


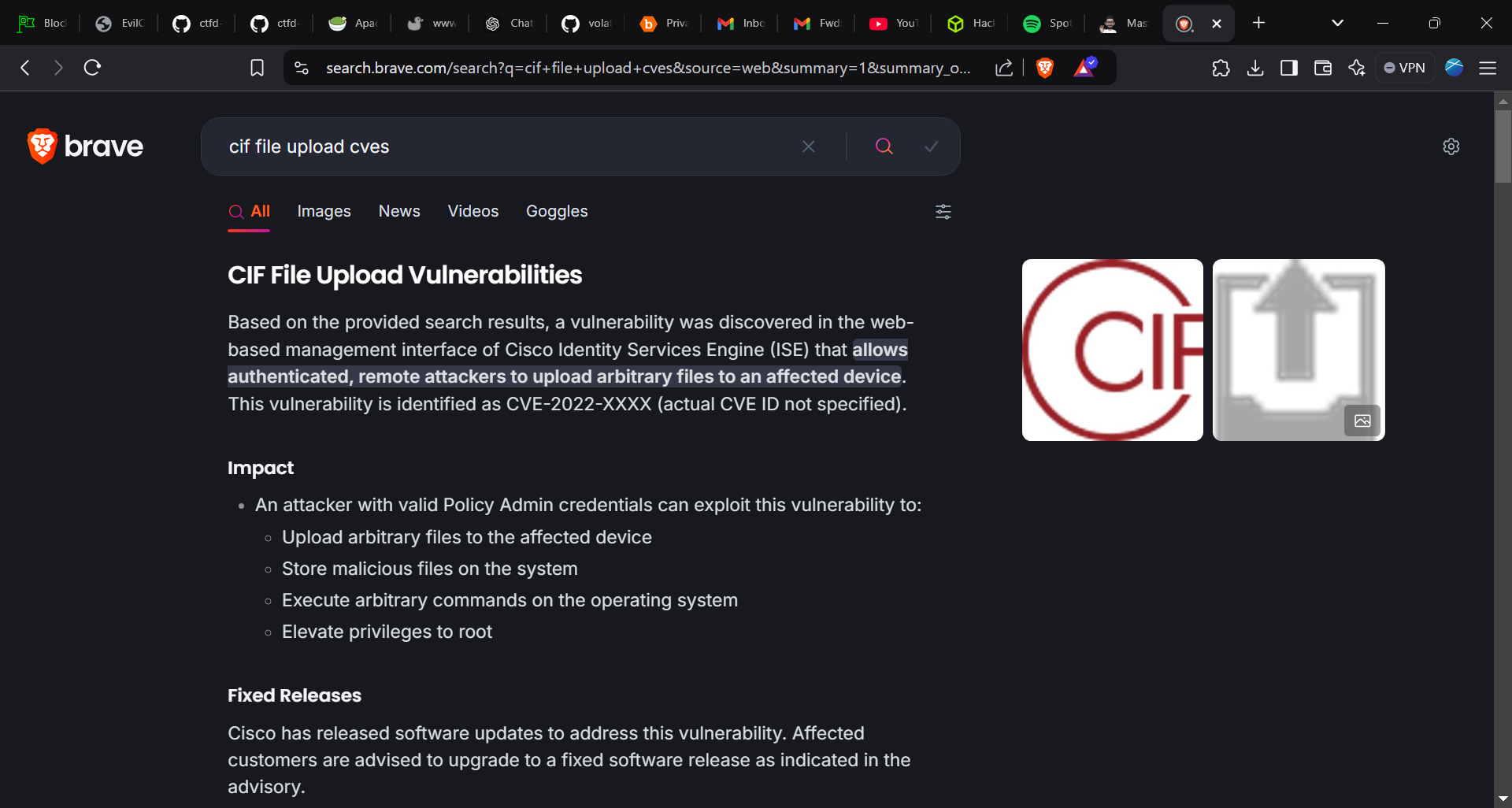


**2. Initial Exploitation**

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**

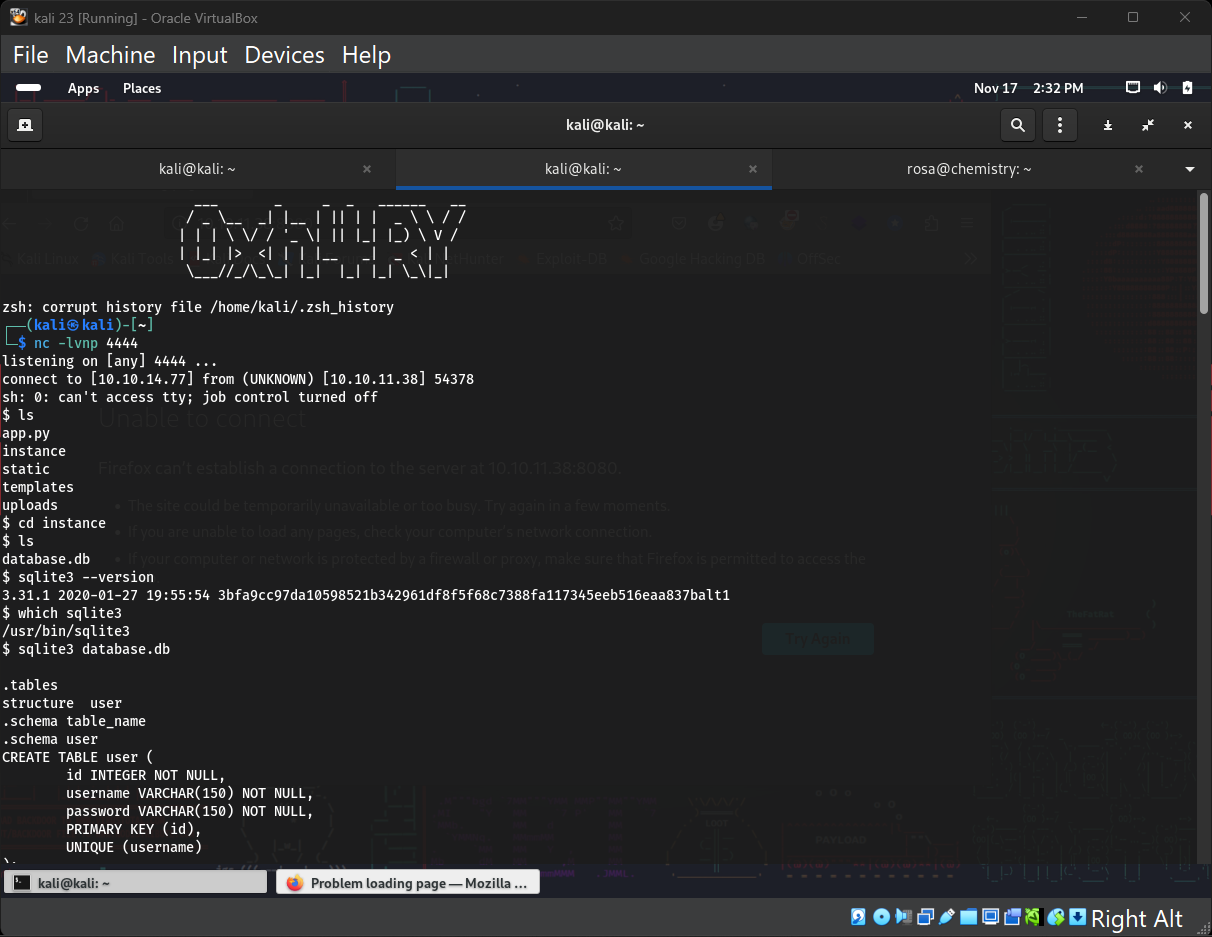
After obtaining a shell, a search of the file system uncovered a database file named database.b containing credentials for the SSH user rosa.

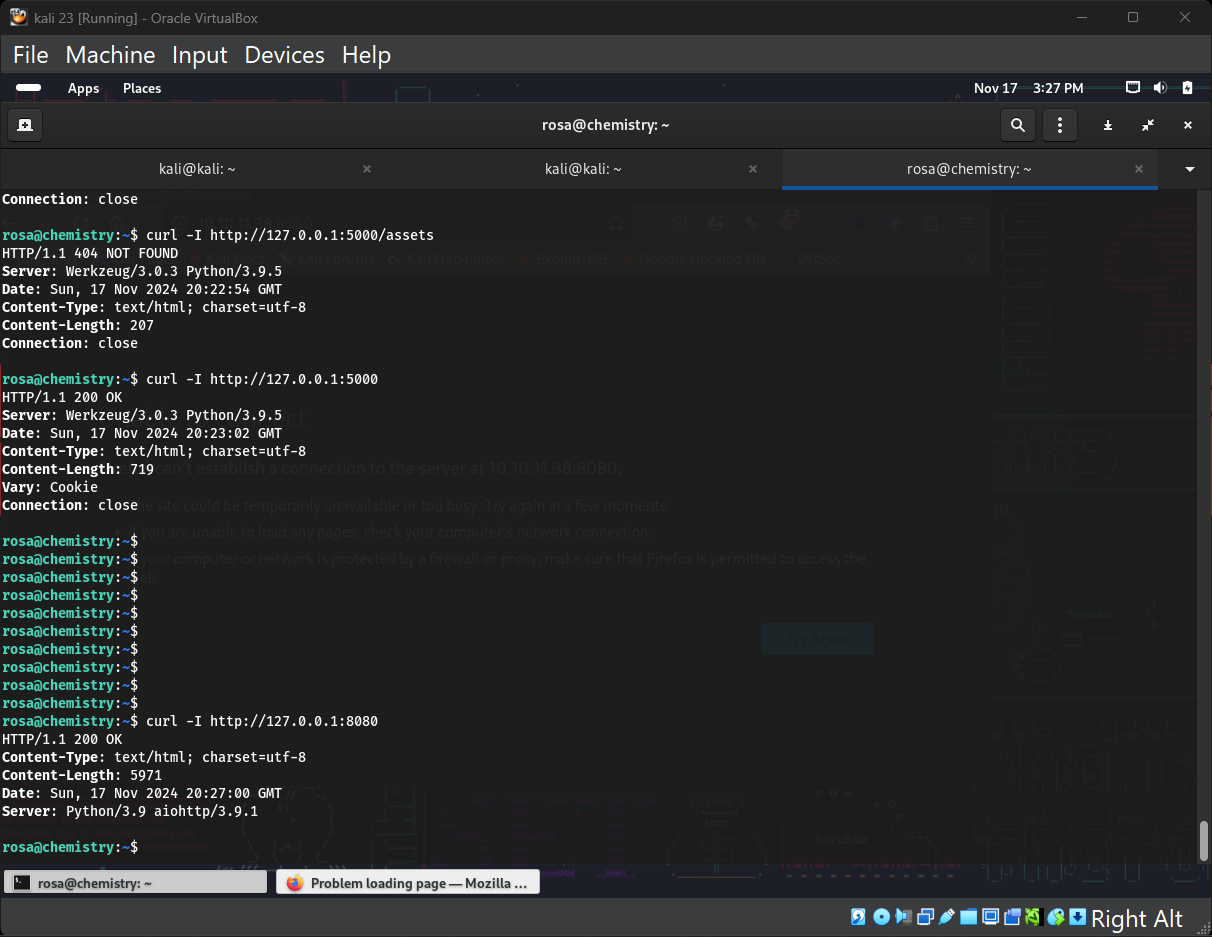
* **Credentials Extracted**: Successfully used the credentials to log in via SSH:

bash

Copy code

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**

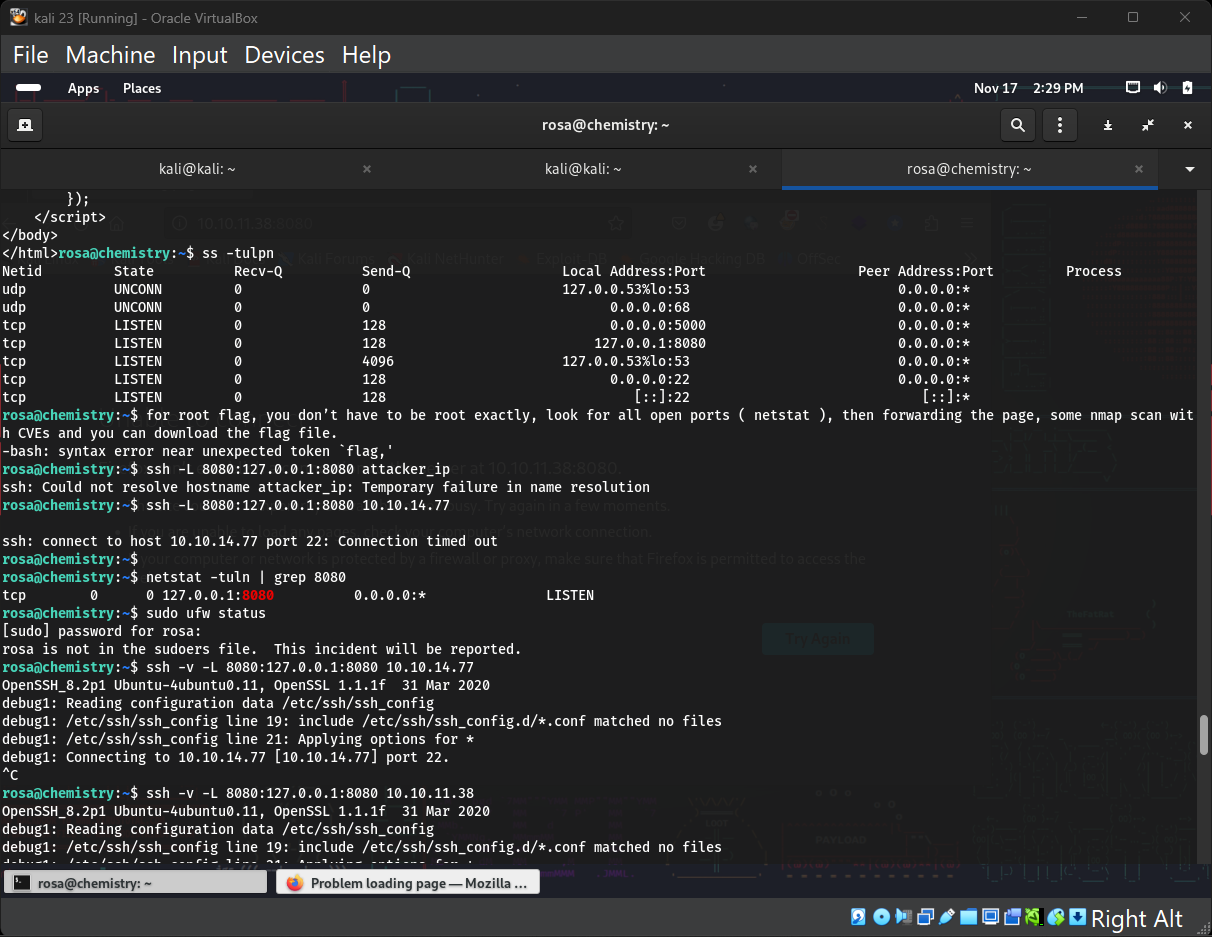
While enumerating the system, a locally hosted service was identified on port **8080**.

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

bash

Copy code

netstat –tuln



**6. Port Forwarding**

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

bash

Copy code

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.

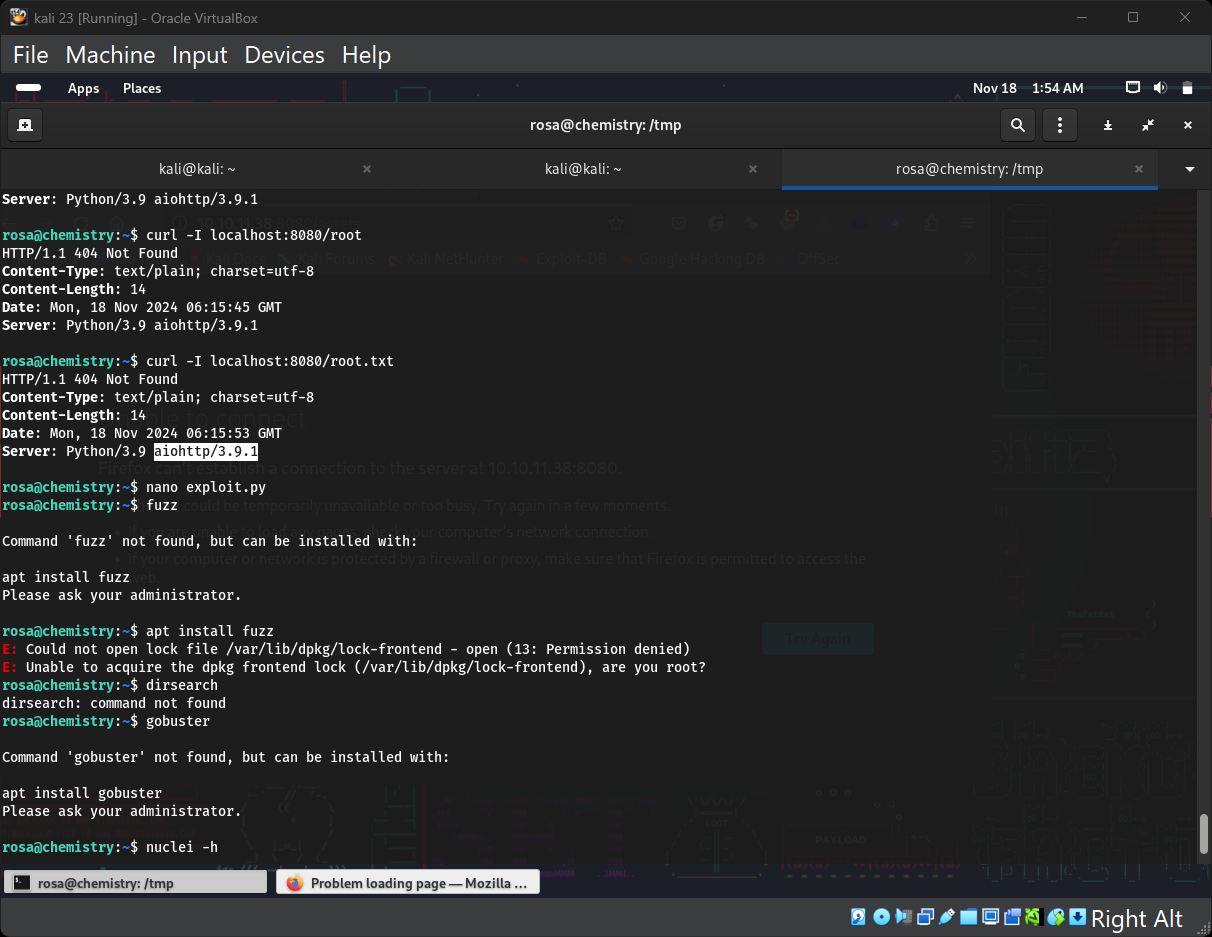
**7. Vulnerability Discovery**

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

bash

Copy code

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.