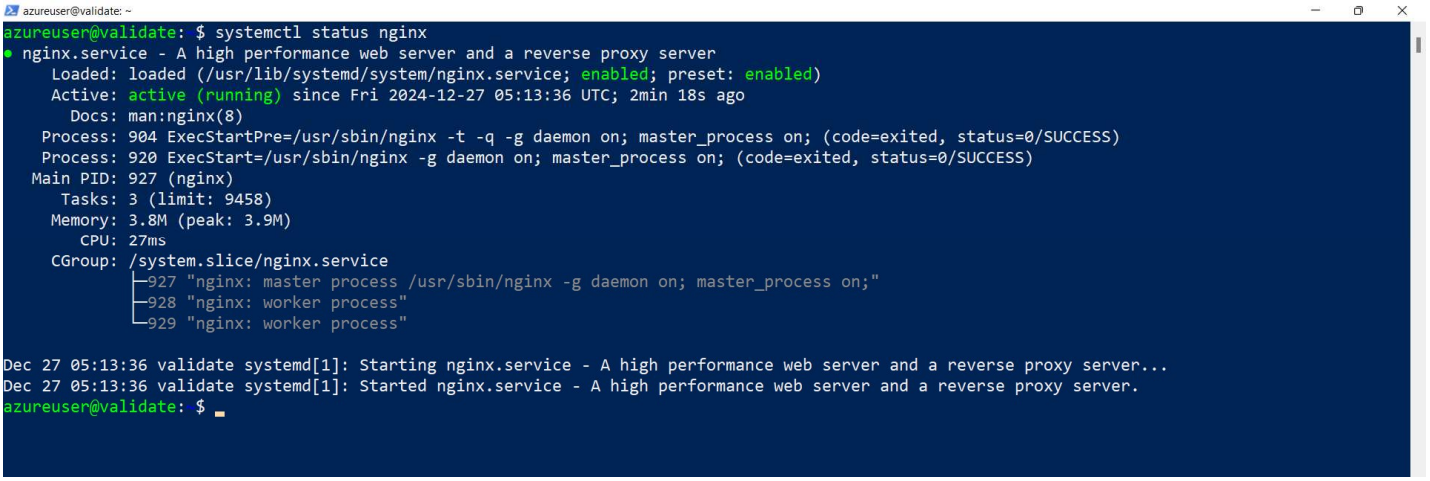
Invalid Login Attempts

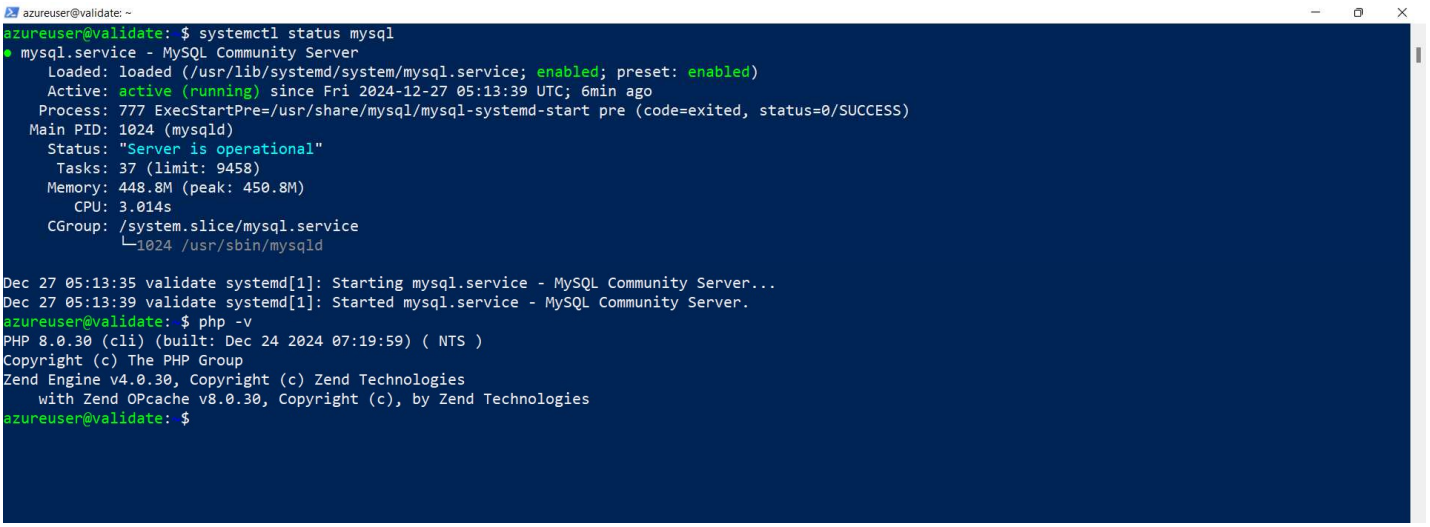
Create a Python script to:

- Parse logs from a web server.
- Identify IPs causing the most failed login attempts.
- Block those IPs by dynamically updating firewall rules.

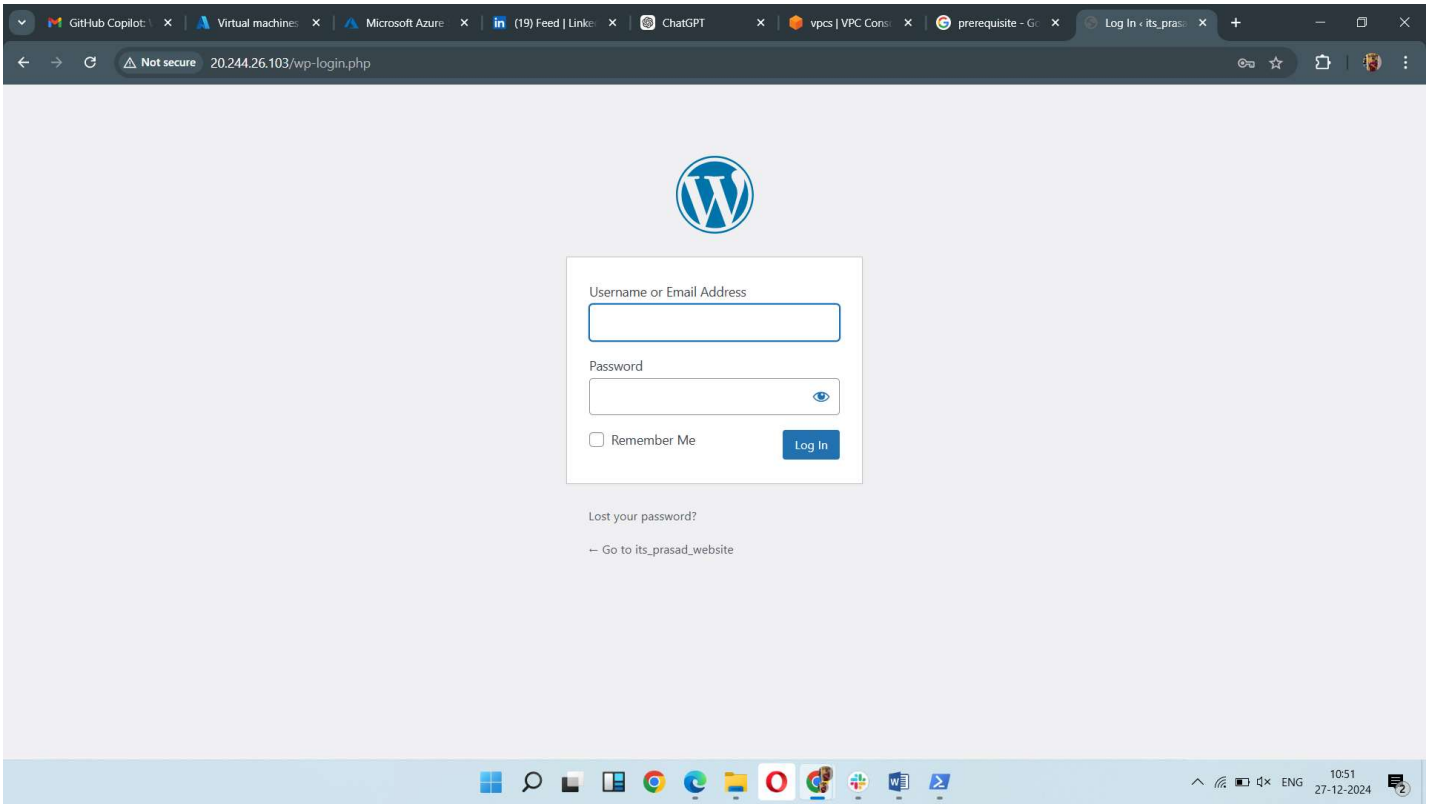
Install nginx that acts as proxy server.



#Install the prerequisite for WordPress as PHP and MySQL.

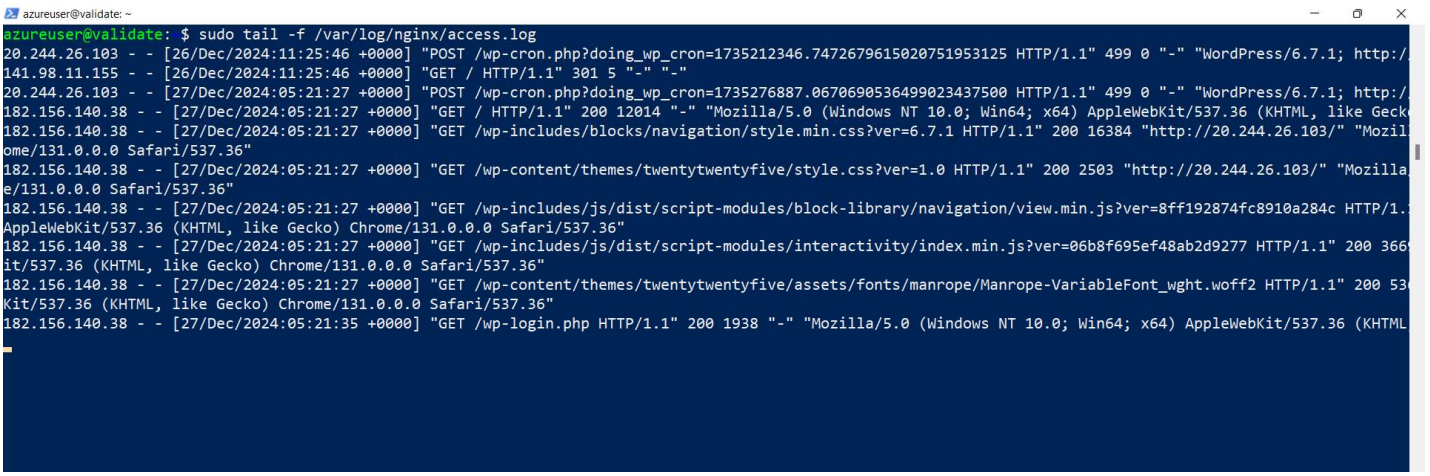


Installing wordpress



By checking the access log of the nginx I understood the HTTP status code

- 2XX:success
    - 200: OK (The request was success)
    - 201: Created (A new resource was successfully created)
    - 204: No Content (Request succeeded, but there's no content to send back)
- 4xx: Client Errors
    - 400: Bad Request (The request could not be understood)
    - 401: Unauthorized (Authentication is required)
    - 403: Forbidden (The server understood the request but refuses to authorize it)
    - 404: Not Found (The requested resource is not found)



Python script that :

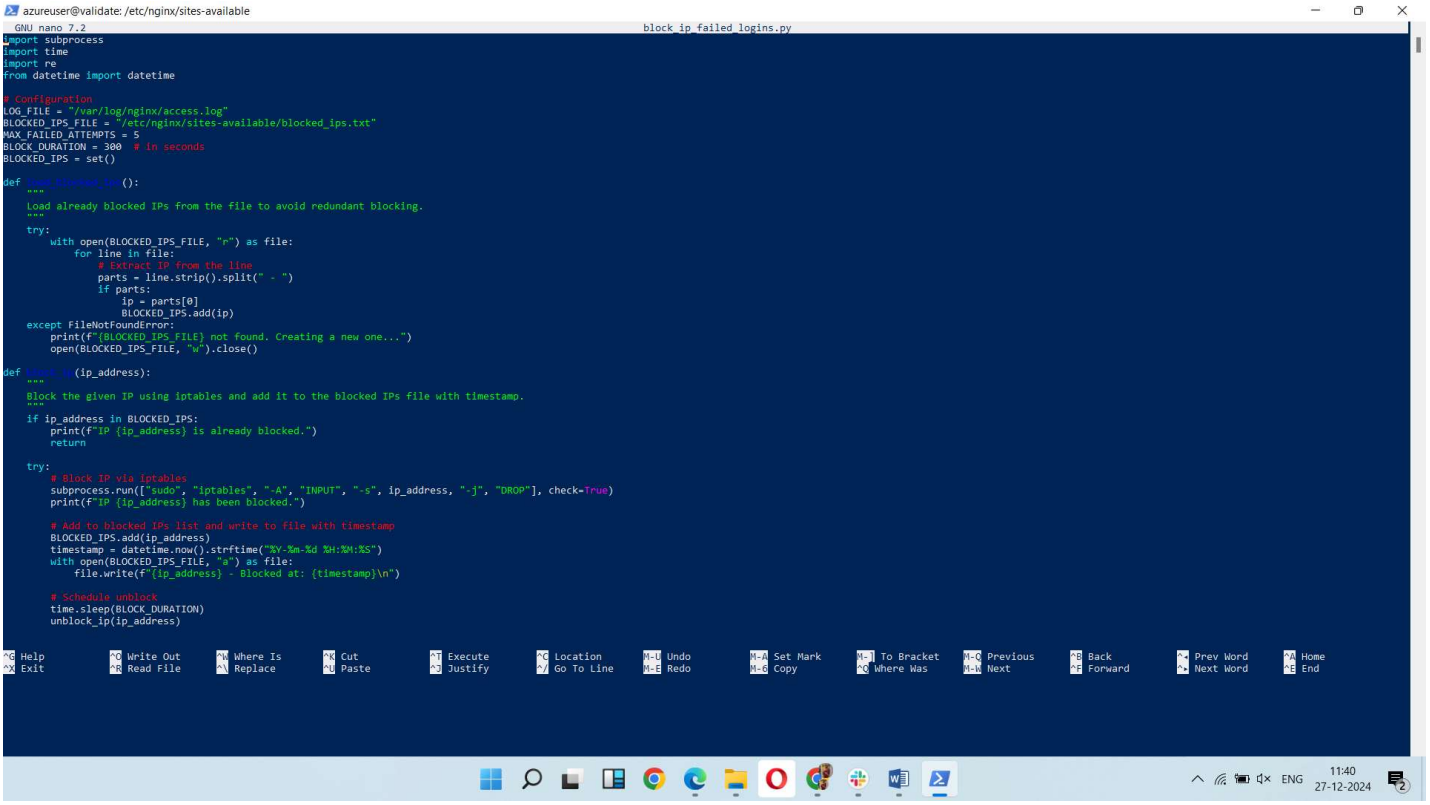
Accessed the nginx logs

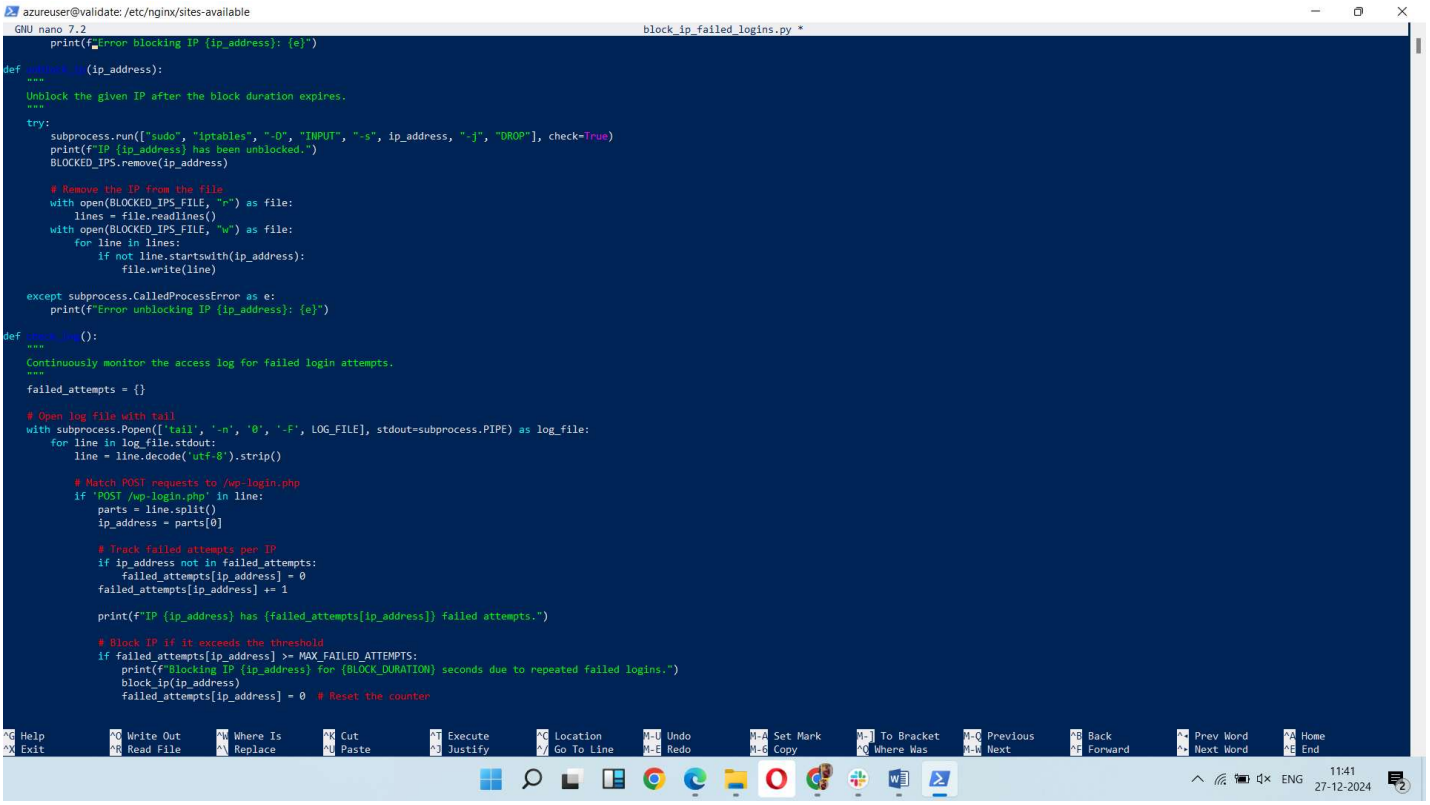
Created the blocked_ip.txt file if not there and use to store the history of blocked IP addresses.

Max attempted per user is set as 5 & Max duration that IP will Remain blocked is 5 minutes.

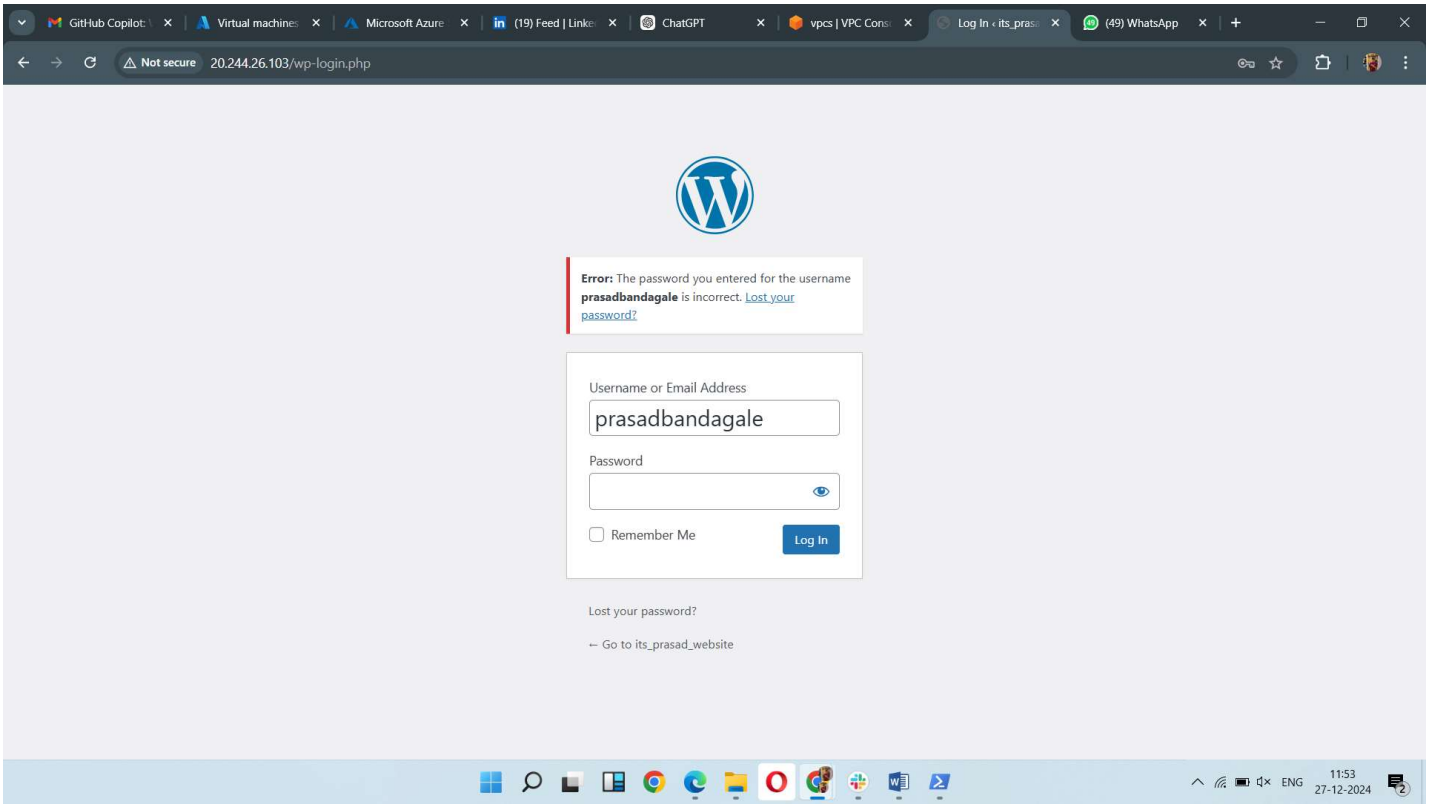
Blocking the User IP using iptables and stored it blocked_ip.txt file with date and time.

Use tail –f for continuously monitoring the access.log file

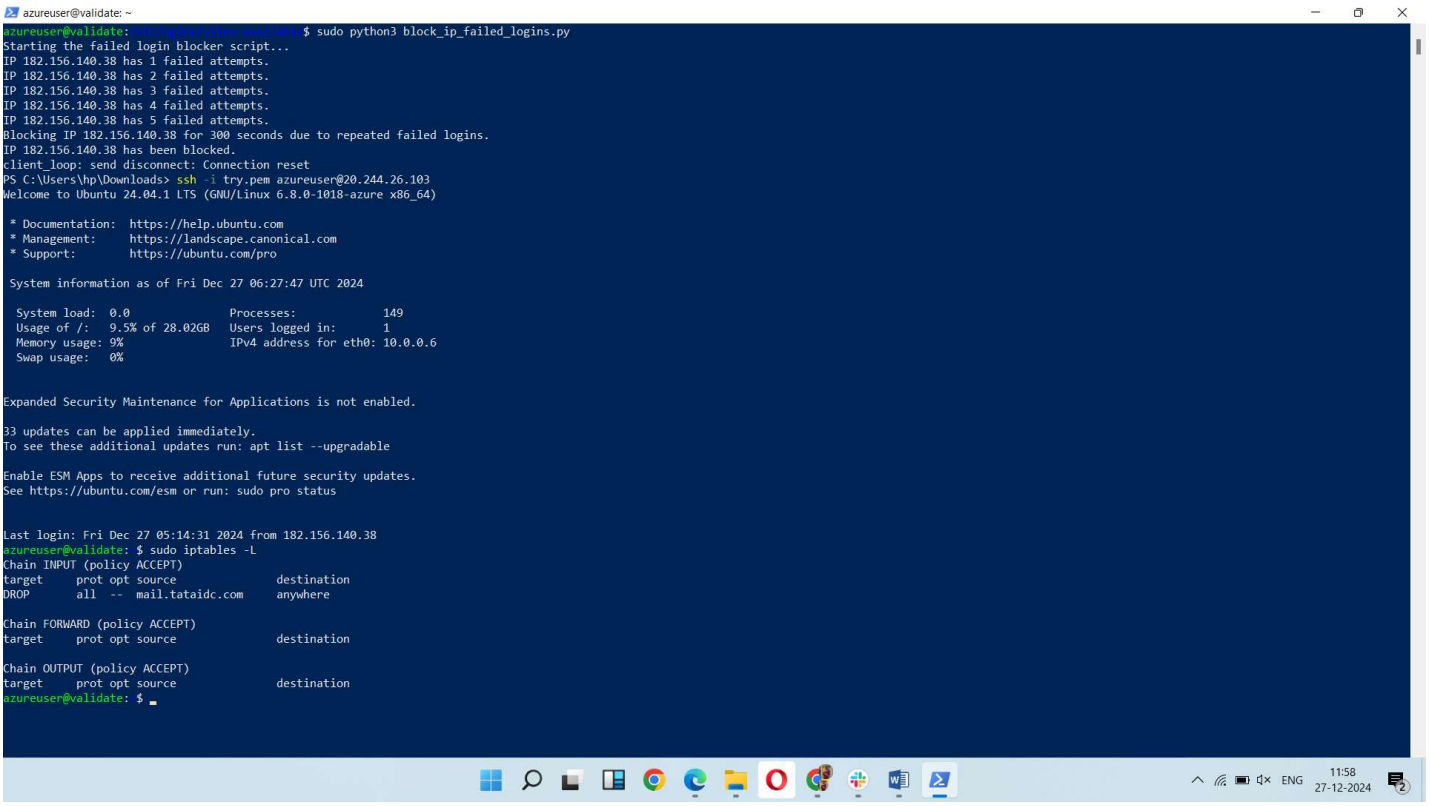




Intentionally attempted to log in 5 times with incorrect credentials to check how the script would react.



The script blocked the IP after 5 failed login attempts, as verified in iptables. Since the IP was blocked, SSH access to the server using the same IP was not possible. Therefore, the network was changed, and the server was accessed again.



The Script also stored the blocked IP address with Date and time in the Blocked_ips.txt file

After 5 minutes, the IP address blocked by the script is automatically unblocked.