

Author: Bhargav Raj Dutta

LinkedIn: [www.linkedin.com/in/bhargav-raj-dutta-80251a1b4](https://www.linkedin.com/in/bhargav-raj-dutta-80251a1b4)

# **Log Analyzer Tool: Enhancing Security through Log Analysis**

**A Python-based Tool for Detecting Suspicious  
Login Attempts with Visualization**

**Created by: Bhargav Raj Dutta**

**Date: 5 December 2024**

# Introduction

## Overview:

This Log Analyzer Tool automates the process of analyzing server logs to detect any potential security threats. By focusing on failed login attempts, the tool identifies the suspicious IPs and provides an actionable insight for IT administrators.

## Purpose:

To assist system administrators in monitoring the login activities.

To provide a use case of how Python can be applied in cybersecurity.

## Scope:

Applicable for analyzing any text-based server log files, such as Linux SSH logs or web server logs.

Customizable for different type of log formats by modifying the regular expression patterns.

## Relevance in Cybersecurity:

With the increasing cyber threats, tools like the Log Analyzer helps to detect any brute-force or unauthorized access attempts quickly, reducing incident response time.

Author: Bhargav Raj Dutta

LinkedIn: [www.linkedin.com/in/bhargav-raj-dutta-80251a1b4](https://www.linkedin.com/in/bhargav-raj-dutta-80251a1b4)

## **Features**

### **Interactive GUI:**

I made a Tkinter-based interface simplifies the log selection and report generation for non-technical users.

### **Regex-Powered Parsing:**

It Extracts the IP addresses with failed login attempts from any text-based log file.

### **Detailed Reporting:**

It generates a text report (suspicious\_activity\_report.txt) that summarizes all the failed login attempts per IP address.

### **Visualization:**

It creates a bar chart of failed login attempts (suspicious\_activity\_graph.png) for easy interpretation.

### **Customizability:**

The regex pattern can be adjusted to support various log file formats.

### **Error Handling:**

Invalid or missing files trigger user-friendly error messages.

Ensures smooth user experience even with unexpected inputs.

## Technical Overview

### Core Functionality:

Reads log files line-by-line to identify patterns.

Extracts IP addresses associated with failed login attempts.

Aggregates and visualizes the results.

### Key Modules Used:

- re: For regular expression-based log parsing.
- tkinter: For the graphical interface.
- matplotlib: For data visualization.

### Log Format:

Expected format for failed login attempts:

**[Date Time] Failed password for [user] from [IP Address]**

```
Dec 04 10:01:15 server sshd[10235]: Failed password for user root from 192.168.1.2 port 22 ssh2
Dec 04 10:02:45 server sshd[10236]: Failed password for invalid user test from 192.168.1.3 port 22 ssh2
Dec 04 10:03:20 server sshd[10237]: Accepted password for user john from 192.168.1.4 port 22 ssh2
Dec 04 10:04:05 server sshd[10238]: Failed password for user guest from 192.168.1.5 port 22 ssh2
```

## High-Level Algorithm

### 1. Initialize GUI:

Create an interface using Tkinter.

Add a button to allow users to select a log file.

### 2. Log Parsing:

Use the re to search for the failed login patterns in the selected log file that is selected by user

Extract and store IP addresses of suspicious activities.

### 3. Data Aggregation:

Use collections. Counter to count occurrences of each IP address.

### 4. Generate Outputs:

Create a human-readable report in the suspicious\_activity\_report.txt.

Generate and save a bar chart in the suspicious\_activity\_graph.png.

### 5. Error Handling:

Validate the selected file's format and content.

Inform users if no suspicious activity is found.

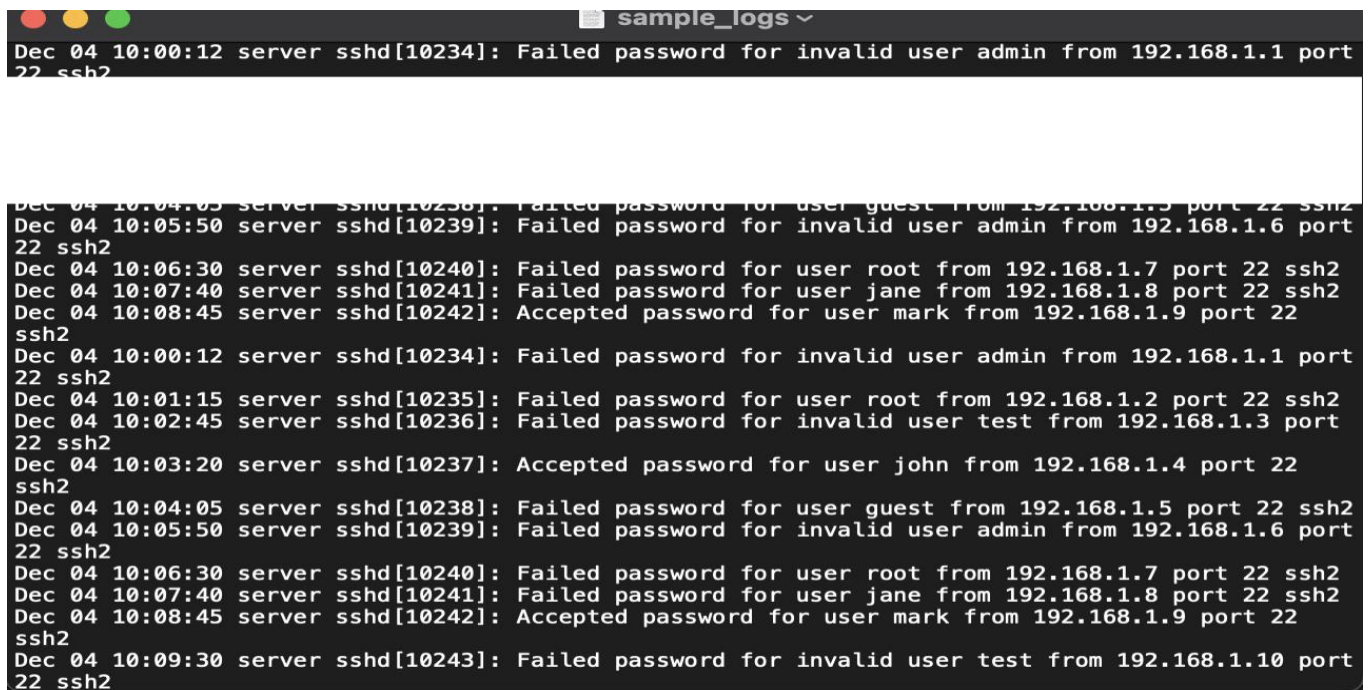
Author: Bhargav Raj Dutta

LinkedIn: [www.linkedin.com/in/bhargav-raj-dutta-80251a1b4](https://www.linkedin.com/in/bhargav-raj-dutta-80251a1b4)

## Testing and Results

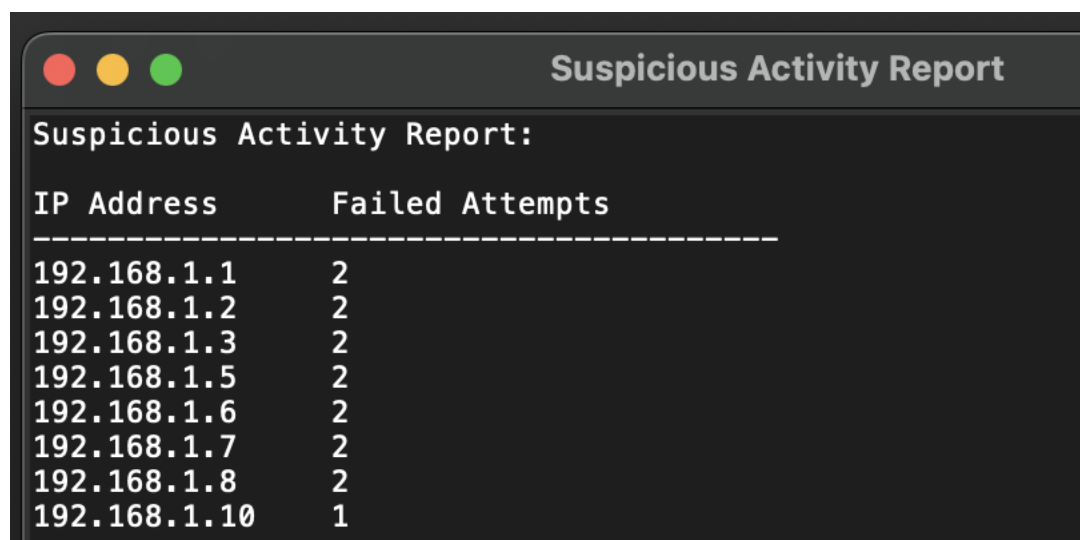
### Sample Log File:

Sample logs (sample\_logs.txt) were used for testing. This is an example of the sample log file.

A terminal window titled 'sample\_logs' displaying a series of SSH log entries. The logs show failed password attempts for various users (admin, root, jane, test) from different IP addresses (192.168.1.1 through 192.168.1.10) and one successful login for user 'mark' from 192.168.1.9.

```
Dec 04 10:00:12 server sshd[10234]: Failed password for invalid user admin from 192.168.1.1 port 22 ssh2
Dec 04 10:04:05 server sshd[10238]: Failed password for user guest from 192.168.1.5 port 22 ssh2
Dec 04 10:05:50 server sshd[10239]: Failed password for invalid user admin from 192.168.1.6 port 22 ssh2
Dec 04 10:06:30 server sshd[10240]: Failed password for user root from 192.168.1.7 port 22 ssh2
Dec 04 10:07:40 server sshd[10241]: Failed password for user jane from 192.168.1.8 port 22 ssh2
Dec 04 10:08:45 server sshd[10242]: Accepted password for user mark from 192.168.1.9 port 22 ssh2
Dec 04 10:00:12 server sshd[10234]: Failed password for invalid user admin from 192.168.1.1 port 22 ssh2
Dec 04 10:01:15 server sshd[10235]: Failed password for user root from 192.168.1.2 port 22 ssh2
Dec 04 10:02:45 server sshd[10236]: Failed password for invalid user test from 192.168.1.3 port 22 ssh2
Dec 04 10:03:20 server sshd[10237]: Accepted password for user john from 192.168.1.4 port 22 ssh2
Dec 04 10:04:05 server sshd[10238]: Failed password for user guest from 192.168.1.5 port 22 ssh2
Dec 04 10:05:50 server sshd[10239]: Failed password for invalid user admin from 192.168.1.6 port 22 ssh2
Dec 04 10:06:30 server sshd[10240]: Failed password for user root from 192.168.1.7 port 22 ssh2
Dec 04 10:07:40 server sshd[10241]: Failed password for user jane from 192.168.1.8 port 22 ssh2
Dec 04 10:08:45 server sshd[10242]: Accepted password for user mark from 192.168.1.9 port 22 ssh2
Dec 04 10:09:30 server sshd[10243]: Failed password for invalid user test from 192.168.1.10 port 22 ssh2
```

## Output of the Report

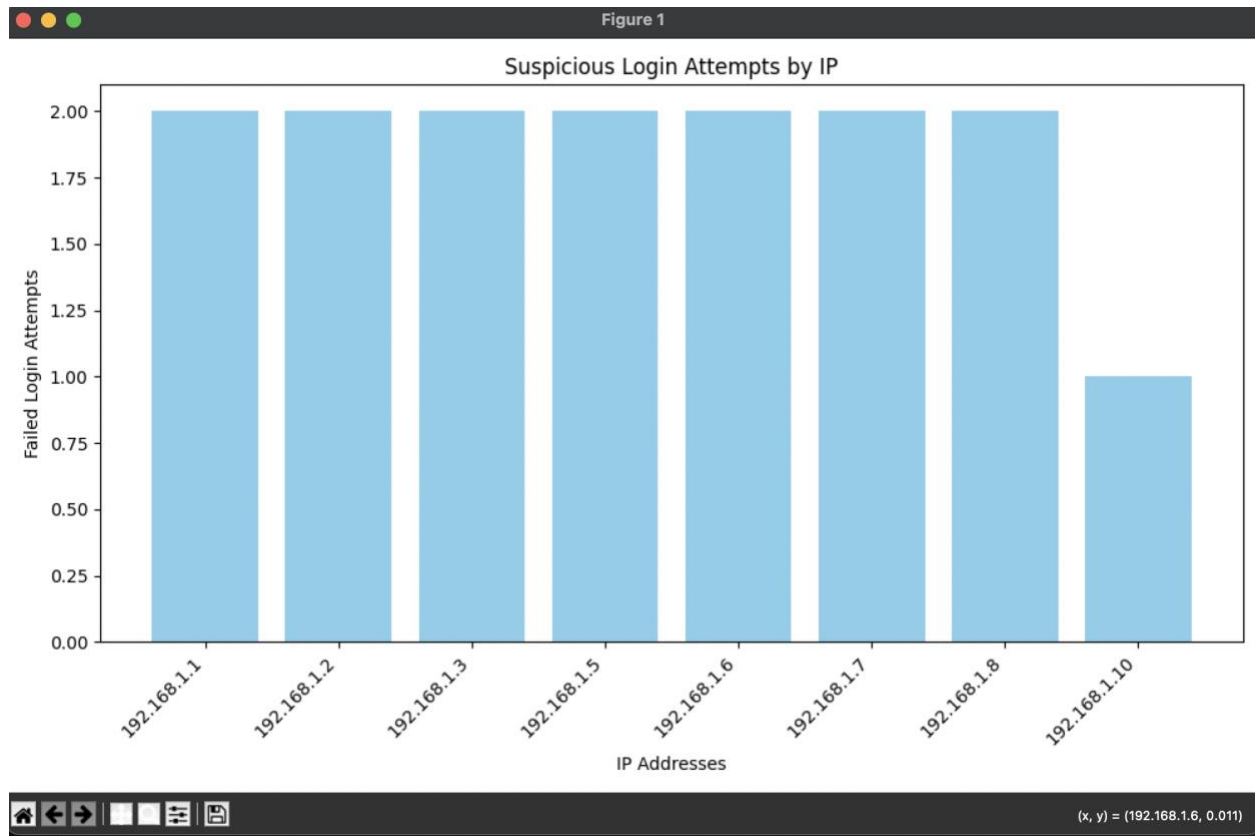
A window titled 'Suspicious Activity Report' displaying a table of failed login attempts. The table has two columns: 'IP Address' and 'Failed Attempts'. The data shows multiple failed attempts from various IP addresses in the 192.168.1.x range, with IP 192.168.1.10 having only one failed attempt.

IP Address	Failed Attempts
192.168.1.1	2
192.168.1.2	2
192.168.1.3	2
192.168.1.5	2
192.168.1.6	2
192.168.1.7	2
192.168.1.8	2
192.168.1.10	1

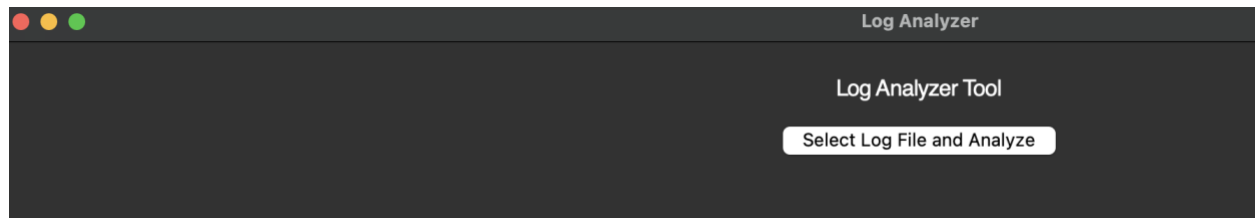
Author: Bhargav Raj Dutta

LinkedIn: [www.linkedin.com/in/bhargav-raj-dutta-80251a1b4](https://www.linkedin.com/in/bhargav-raj-dutta-80251a1b4)

## Graph Screenshot:



## View of the interface



Select the log file and then it will analyze accordingly.

## Challenges Faced

- **Regex Complexity:**

Designing a pattern that correctly matches any failed login attempts while ignoring any unrelated log lines.

- **GUI Design:**

Ensuring that the interface was intuitive for users with limited technical knowledge.

- **Error Handling:**

Managing unexpected file inputs and ensuring clear feedback to users.

## Conclusion

The Log Analyzer Tool is an effective solution for log file analysis, helping identify potential security threats quickly. Its simplicity, coupled with the power of Python, makes it a valuable addition to any administrator's toolkit. Future enhancements will further improve its versatility and utility.



Author: Bhargav Raj Dutta

LinkedIn: [www.linkedin.com/in/bhargav-raj-dutta-80251a1b4](https://www.linkedin.com/in/bhargav-raj-dutta-80251a1b4)

## References

1. Python Official Documentation: <https://docs.python.org/3/>
  2. Tkinter GUI Documentation: <https://docs.python.org/3/library/tkinter.html>
  3. Matplotlib Documentation: <https://matplotlib.org/>
- 

CHEERS! 😊

If you got any feedback, please feel free to email me  
@ **bhargavrajdutta685@gmail.com**

Author: Bhargav Raj Dutta

LinkedIn: [www.linkedin.com/in/bhargav-raj-dutta-80251a1b4](https://www.linkedin.com/in/bhargav-raj-dutta-80251a1b4)