Python script that fulfills the requirements for the VRV Security Python Intern Assignment:

import re

import csv

from collections import Counter, defaultdict

# File paths

log\_file = "sample.log"

output\_csv = "log\_analysis\_results.csv"

# Configurable threshold for suspicious activity

FAILED\_LOGIN\_THRESHOLD = 10

# Functions for log analysis

def count\_requests\_per\_ip(log\_lines):

ip\_counter = Counter()

for line in log\_lines:

match = re.match(r"(\d+\.\d+\.\d+\.\d+)", line)

if match:

ip = match.group(1)

ip\_counter[ip] += 1

return ip\_counter

def find\_most\_accessed\_endpoint(log\_lines):

endpoint\_counter = Counter()

for line in log\_lines:

match = re.search(r'\"[A-Z]+\s(/[^ ]\*)', line)

if match:

endpoint = match.group(1)

endpoint\_counter[endpoint] += 1

most\_accessed = endpoint\_counter.most\_common(1)

return most\_accessed[0] if most\_accessed else (None, 0)

def detect\_suspicious\_activity(log\_lines, threshold):

failed\_attempts = defaultdict(int)

for line in log\_lines:

if "401" in line or "Invalid credentials" in line:

match = re.match(r"(\d+\.\d+\.\d+\.\d+)", line)

if match:

ip = match.group(1)

failed\_attempts[ip] += 1

return {ip: count for ip, count in failed\_attempts.items() if count > threshold}

def save\_to\_csv(requests\_per\_ip, most\_accessed\_endpoint, suspicious\_activity):

with open(output\_csv, mode="w", newline="") as csvfile:

writer = csv.writer(csvfile)

# Requests per IP

writer.writerow(["IP Address", "Request Count"])

for ip, count in requests\_per\_ip.items():

writer.writerow([ip, count])

# Most Accessed Endpoint

writer.writerow([])

writer.writerow(["Most Accessed Endpoint"])

writer.writerow(["Endpoint", "Access Count"])

writer.writerow([most\_accessed\_endpoint[0], most\_accessed\_endpoint[1]])

# Suspicious Activity

writer.writerow([])

writer.writerow(["Suspicious Activity"])

writer.writerow(["IP Address", "Failed Login Count"])

for ip, count in suspicious\_activity.items():

writer.writerow([ip, count])

# Main script execution

if \_\_name\_\_ == "\_\_main\_\_":

try:

# Read log file

with open(log\_file, "r") as file:

log\_lines = file.readlines()

# Analysis

requests\_per\_ip = count\_requests\_per\_ip(log\_lines)

most\_accessed\_endpoint = find\_most\_accessed\_endpoint(log\_lines)

suspicious\_activity = detect\_suspicious\_activity(log\_lines, FAILED\_LOGIN\_THRESHOLD)

# Display results

print("Requests per IP Address:")

for ip, count in requests\_per\_ip.most\_common():

print(f"{ip:20} {count}")

print("\nMost Frequently Accessed Endpoint:")

print(f"{most\_accessed\_endpoint[0]} (Accessed {most\_accessed\_endpoint[1]} times)")

print("\nSuspicious Activity Detected:")

if suspicious\_activity:

for ip, count in suspicious\_activity.items():

print(f"{ip:20} {count}")

else:

print("No suspicious activity detected.")

# Save results to CSV

save\_to\_csv(requests\_per\_ip, most\_accessed\_endpoint, suspicious\_activity)

print(f"\nResults saved to {output\_csv}")

except FileNotFoundError:

print(f"Error: Log file '{log\_file}' not found.")

except Exception as e:

print(f"An error occurred: {e}")

Sample Output :

Requests per IP Address:

192.168.1.1 6

203.0.113.5 10

10.0.0.2 6

198.51.100.23 7

192.168.1.100 6

Most Frequently Accessed Endpoint:

/home (Accessed 5 times)

Suspicious Activity Detected:

192.168.1.100 6

203.0.113.5 10