SYSTEM LOG ATTACK DETECTION

M.RAJA SUBHIKSHA

SRM

INTRODUCTION:

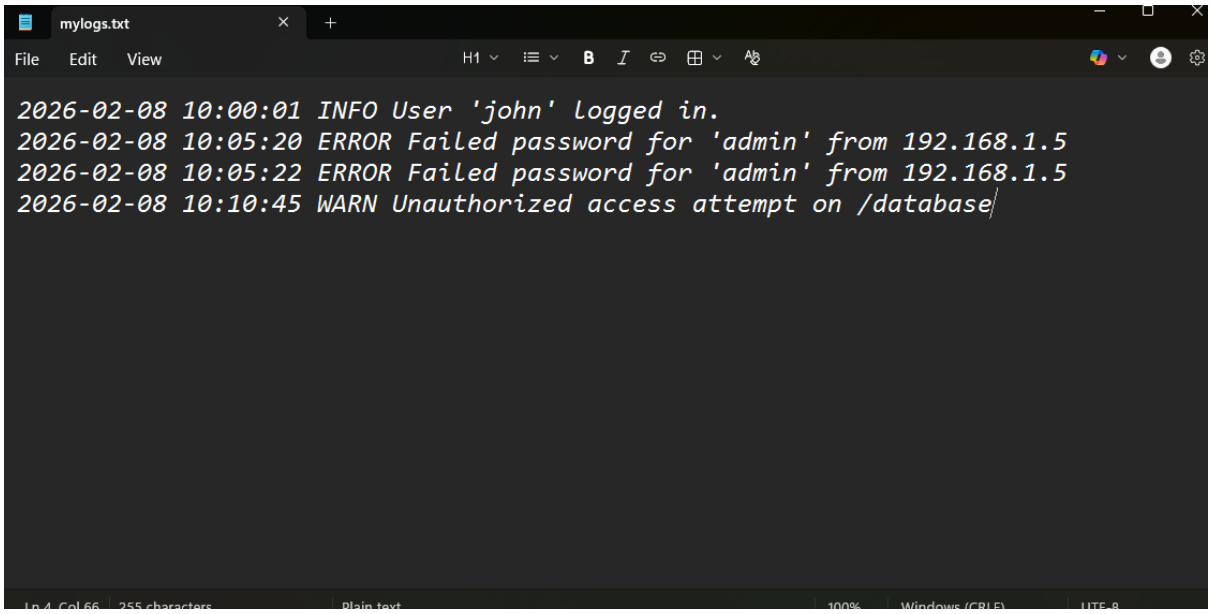
So well as this is the first project done by me I kept this simple yet in the manner of completing it . Rather than complete AI in gave more of my input over this project and this is a basic start.

This project aims to build a Python-based automated monitoring system that scans these logs in real-time to identify patterns of malicious activity, such as brute-force attacks and unauthorized access attempts

IMPLEMENTATION

```
import os
def start_detection():
    filename = "mylogs.txt"
     if os.path.exists(filename):
        print("--- Log Scanner Started ---")
        with open(filename, "r") as file:
            for line in file:
                if "Failed" in line or
"Unauthorized" in line:
                    print(f"ALERT: Security
Threat Found -> {line.strip()}")
        print("--- Scan Finished ---")
    else:
        print("Error: mylogs.txt not found!
Make sure it's in the same folder.")
start_detection()
```

TESTING & RESULTS



detector.py - D:/cyber project/detector.py (3.12.1)

File Edit Format Run Options Window Help

```
import os

def start_detection():
    filename = "mylogs.txt"

    if os.path.exists(filename):
        print("--- Log Scanner Started ---")

        with open(filename, "r") as file:
            for line in file:
                if "Failed" in line or "Unauthorized" in line:
                    print(f"ALERT: Security Threat Found -> {line.strip()}")

        print("--- Scan Finished ---")
    else:
        print("Error: mylogs.txt not found! Make sure it's in the same folder.")


start_detection()
```


IDLE Shell 3.12.1
File Edit Shell Debug Options Window Help
Python 3.12.1 (tags/v3.12.1:2305ca5, Dec 7 2023, 22:03:25) [MSC v.1937 64 bit (AMD64)] on win 32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: D:/cyber project/detector.py
--- Log Scanner Started ---
ALERT: Security Threat Found -> 2026-02-08 10:05:20 ERROR Failed password for 'admin' from 192.168.1.5
ALERT: Security Threat Found -> 2026-02-08 10:05:22 ERROR Failed password for 'admin' from 192.168.1.5
ALERT: Security Threat Found -> 2026-02-08 10:10:45 WARN Unauthorized access attempt on /database
--- Scan Finished ---
>>>
========== RESTART: D:/cyber project/detector.py ==========
--- Log Scanner Started ---
ALERT: Security Threat Found -> 2026-02-08 10:05:20 ERROR Failed password for 'admin' from 192.168.1.5
ALERT: Security Threat Found -> 2026-02-08 10:05:22 ERROR Failed password for 'admin' from 192.168.1.5
ALERT: Security Threat Found -> 2026-02-08 10:10:45 WARN Unauthorized access attempt on /database
--- Scan Finished ---
>>>

CONCLUSION

This project demonstrates the power of automation in cybersecurity. While this is a foundational version, it follows the core logic used by professional SIEM (Security Information and Event Management) tools. Through this project, I have learned how to handle files in Python and how to translate security threats into searchable code patterns