No-Code Security Event Detection & Response Using Microsoft Power Automate Desktop

Saturday Afternoon Project by [Manish Pulluru]

Executive Summary

This project demonstrates how anyone—even with zero coding experience—can automate security monitoring and basic incident response on a Windows computer using **Microsoft Power Automate Desktop (PAD)**. In a single afternoon, I built a working demo that detects suspicious logins, alerts the user, logs the event, and can even lock the workstation—all through simple drag-and-drop automation.

Table of Contents

---

1. Introduction

Why this project?
Cybersecurity is critical—even for individuals and small businesses. Most can’t afford expensive security tools, but Power Automate Desktop offers a free, no-code way to build powerful automations. This project shows how to create a simple "mini-SOC" (Security Operations Center) at home or in any SMB setting.

---

2. Project Objectives

- **Monitor a log file** for suspicious activity (like failed login attempts).
- **Alert the user** in real-time when a suspicious event is detected.
- **Log the incident** to a separate alert file for later review or audit.
- **Optionally lock the workstation** automatically as a protective measure.

---

3. Tools Used

- **Microsoft Power Automate Desktop** (latest version as of July 2025)
- **Windows 10/11 PC**
- **Notepad** (to create/edit sample log files)
- **Screenshots** for documentation
- *(Optional: Sample log file for demo testing)*

---

4. Step-by-Step Implementation

A. Preparation

1. **Installed Power Automate Desktop** on my computer.
2. **Created a sample log file** named security_log.txt with entries like:
3. 2024-07-08 14:10:00 LOGIN_FAILED User: Vivek IP: 192.168.1.15
4. 2024-07-08 14:12:05 LOGIN_SUCCESS User: Admin IP: 192.168.1.10

B. Building the Automation in PAD

1. Read the Log File

- Used the **Read text from file** action to load the entire contents of security_log.txt into a variable.

2. Split the Log into Lines

- Used **Split text** action to break the file into a list of lines using the "New line" delimiter and a reasonable split limit.

3. Loop Through Each Log Entry

- Used **For each** action to process every line from the log.

4. Check for Suspicious Activity

- Used an **If** condition to check if the current line contains "LOGIN_FAILED".

5. Respond to Detection

- **Display message:** Show a pop-up security alert with details.
- **Append line to text:** Store the suspicious entry.
- **Write text to file:** Log the event to a new file alert_log.txt.
- **(Optional) Lock workstation:** Use **Run application** to execute the Windows command that locks the device.

C. Visual Flowchart

Read log file → Split by new line → For each line:

If line contains LOGIN_FAILED:

Show alert

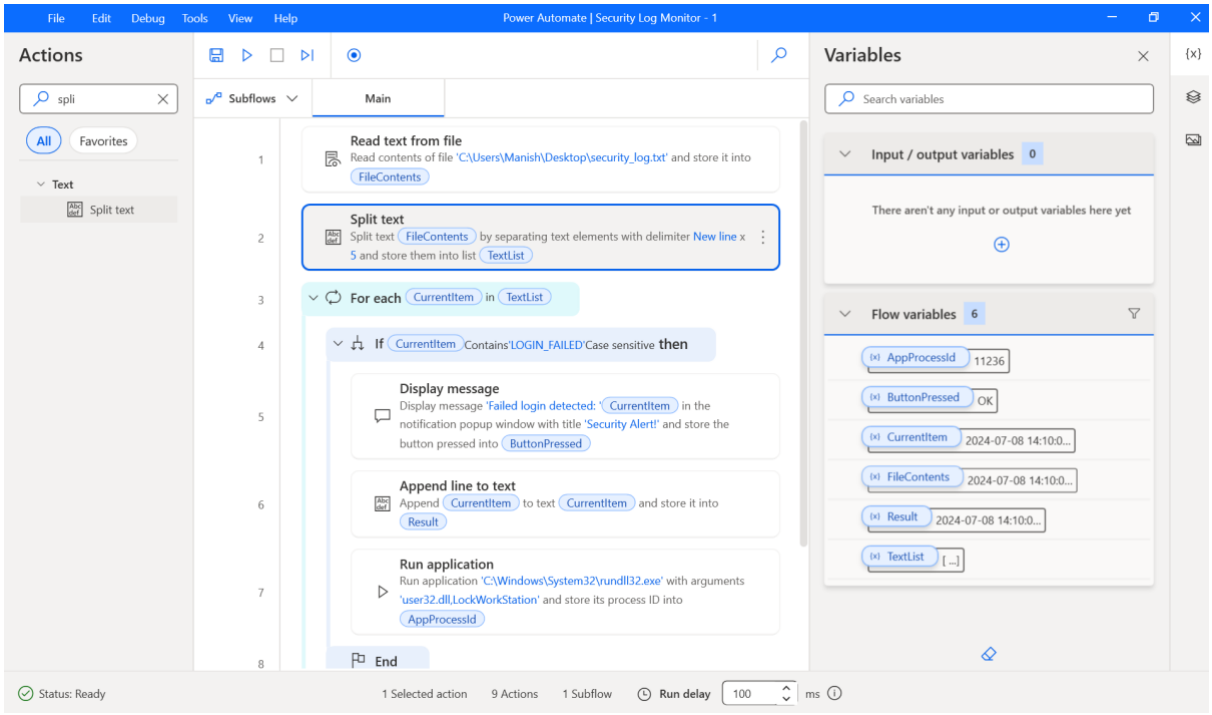
Log event

Lock workstation (optional)

Else:

Continue

D. Screenshots



---

5. Testing and Results

Screen Recording
2025-07-12 203226.r

- Ran the flow in PAD.
- Added a new LOGIN_FAILED entry to security_log.txt and saved it.
- **Observed:**
    - Pop-up alert appeared instantly.
    - New event was added to alert_log.txt.
    - Workstation locked automatically (when enabled).

---

6. Real-Time Use Case Examples

- **Small business:** Instantly alert staff to suspicious logins on shared office PCs.
- **School or lab:** Auto-locks a workstation and notifies IT if brute-force login attempts are detected.
- **Home user:** Adds a layer of protection against password guessing or unauthorized local access.
- **Audit log:** Maintains a simple, human-readable incident log for compliance or troubleshooting.

---

7. Lessons Learned

- **Power Automate Desktop** can create powerful security automations with zero code.
- Pay attention to delimiter and "times" settings when processing text to avoid errors.
- “Export/import” for PAD flows is not natively supported, so thorough documentation is essential for sharing and reproducibility.
- No-code tools are a fantastic entry point for learning cybersecurity basics!

---

8. Possible Improvements

- **Send email or Teams alerts** for remote notification.
- **Export alerts to Excel** for analysis.
- **Monitor additional log sources** (e.g., Windows Event Logs, firewall logs).
- **Trigger automated remediation** (disable user, notify admin, etc.).
- **Continuous monitoring** using scheduled or always-on PAD flows.

---

9. Conclusion

This Saturday afternoon project proves you don't need a big budget or coding background to automate security monitoring and response. Using only free, beginner-friendly tools, you can protect your PC or business—and learn a ton about IT and cybersecurity automation in the process.

---

10. Appendix

A. Sample Log File (security_log.txt)

2024-07-08 14:10:00 LOGIN_FAILED User: Vivek IP: 192.168.1.15

2024-07-08 14:12:05 LOGIN_SUCCESS User: Admin IP: 192.168.1.10

2024-07-08 14:15:22 LOGIN_FAILED User: TestUser IP: 192.168.1.25

2024-07-08 14:18:31 LOGIN_FAILED User: Guest IP: 192.168.1.88

2024-07-08 14:19:45 LOGIN_SUCCESS User: Vivek IP: 192.168.1.15

B. Flowchart Illustration

(Include a simple diagram if possible.)

C. Resources

- Microsoft Power Automate Desktop Official Documentation
- Project Link
- Manish Pulluru LinkedIn

Contact: manishpulluru@gmail.com

- *For questions, feedback, or collaboration, connect with me on* Manish Pulluru LinkedIn

---

Resume/Portfolio Summary

"Designed and implemented a no-code security incident detection and response system using Power Automate Desktop. Automated the detection of suspicious logins, instant alerting, and incident logging—enabling real-time security for small businesses or personal devices."
