SOC Automation Project: AI-Driven Email & Network Threat Detection

---

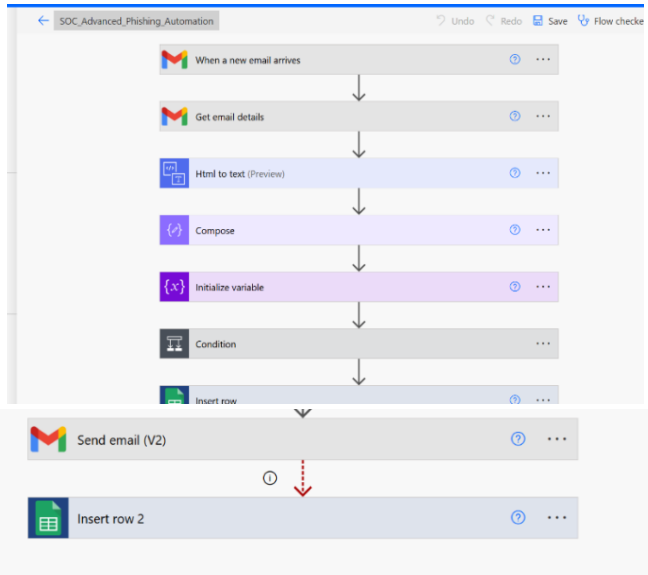
What is this project?

A practical, deployment-ready **Security Operations Center (SOC) automation system**—detects and responds to phishing emails and network threats using free, cloud-based tools. This bridges the gap between student projects and true industry SOC workflows.

What did we build?

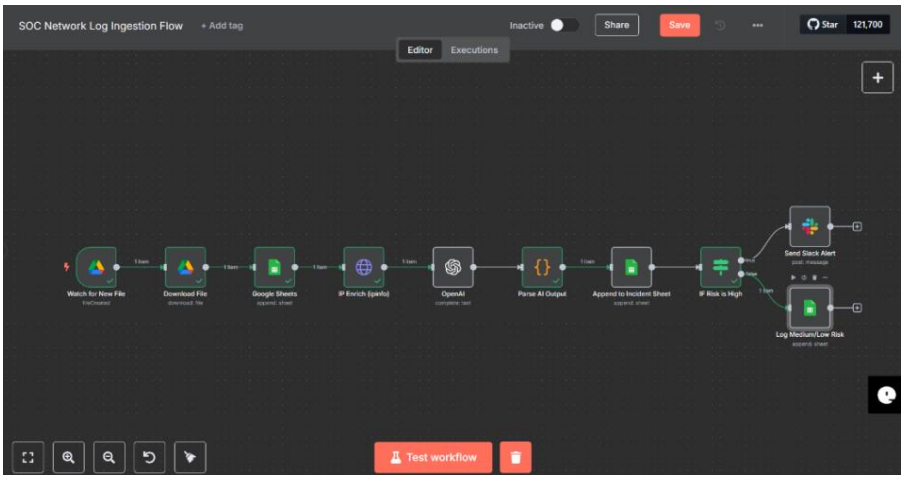
🔷 Automated Email Threat Detection (Power Automate):

- Watches for new phishing/malware emails.
- Parses, scores, and logs into a Google Sheet.



🔷 Smart Network Log Triage (n8n):

- Ingests, enriches, and analyzes network logs.
- Uses AI (OpenAI) for risk scoring and recommendations.



◆ Central Incident Database (Google Sheets):

- Acts as the "integration bus" between automations.
- Each incident/event is logged, tracked, and ready for analytics.
- **Google Apps Script** used for advanced header/metadata extraction and automation.

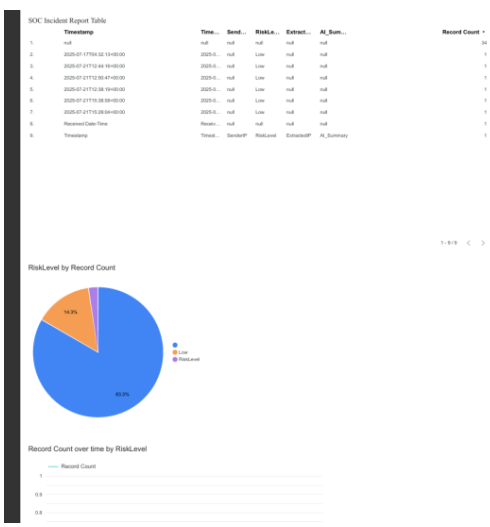
Name	Owner	Date modified	File size
network_alerts	me	1:09 AM	3 KB
insert row 2 SOC_Incident_Log	me	Jul 17	1 KB
SEC Alerts	me	Jun 11	1 KB
SOC_Incident_Log	me	1:49 AM	5 KB

A	B	C	D	E	F	G	H	I	J	K	L
Column1	Column2	Column3	Column4	Column5	Column6	Column7	Column8	Column9	Column10	Column11	Column12
Timestamp	IncidentType	Sender	Subject	SuspiciousURL	SenderIP	GeoLocation	ThreatReputati	AnalystNotes	ipinfo_link	abuseipdb_link	RiskLevel
Received Date-Time	Phishing	From	Subject	Outputs (from Compose)					Check ipinfo	Check AbuseIPDB	

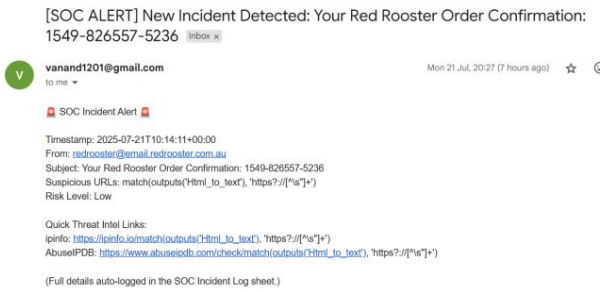
```gas
/**
 * SOC Incident Log - Fill EmailHeaders by matching Subject to Gmail.
 * Sheet must have columns 'Subject' and 'EmailHeaders' in Row 2.
 * Now searches last 7 days and matches subject case-insensitively.
 */

function addHeadersToSOCSheet() {
  const sheetName = 'Sheet1'; // Change if needed
  const sheet = SpreadsheetApp.getActiveSpreadsheet().getSheetByName(sheetName);
  const data = sheet.getDataRange().getValues();

  // Use the second row for actual headers (since the first row is "Column1", etc.)
  const headerRow = data[1]; // <-- This is row 2 (A2, B2, etc.)
  const subjectCol = headerRow.indexOf('Subject');
  const headersCol = headerRow.indexOf('EmailHeaders');

  if (subjectCol === -1 || headersCol === -1) {
    SpreadsheetApp.getUi().alert('Missing required columns: Subject or EmailHeaders');
    return;
  }

  // Build a subject-to-message map from recent emails (case-insensitive)
  const threads = GmailApp.search('newer_than:7d'); // last 7 days
  let subjectToHeaders = {};

  threads.forEach(thread => {
    thread.getMessages().forEach(msg => {
      const subj = (msg.getSubject() || "").toLowerCase().trim();
      const rawHeaders = msg.getRawContent();
      const headerMatch = rawHeaders.match(/^(.*?)\r?\n\r?\n/s);
      subjectToHeaders[subj] = headerMatch ? headerMatch[1] : rawHeaders;
    });
  });

  Logger.log("Found Gmail subjects: " + Object.keys(subjectToHeaders).join(", "));

  // For each row (starting from row 3, since row 2 is headers)
  for (let i = 2; i < data.length; i++) {
    const subj = (data[i][subjectCol] || "").toLowerCase().trim();
    const curHeader = data[i][headersCol];

    if (!curHeader && subj && subjectToHeaders[subj]) {
      Logger.log("Filling headers for sheet row " + (i+1) + " with subject: " + subj);
      sheet.getRange(i + 1, headersCol + 1).setValue(subjectToHeaders[subj]);
```

🔷 Real-Time SOC Alerts & Dashboard:

- Slack notifications for high-risk incidents.
- Google Looker Studio dashboard for live SOC analytics.



🔷 Real-Time SOC Alerts E-mails:



[SOC ALERT] New Incident Detected: Your Red Rooster Order Confirmation: 1549-826557-5236 Inbox ×

vanand1201@gmail.com Mon 21 Jul, 20:27 (7 hours ago)
to me

🚨 SOC Incident Alert 🚨

Timestamp: 2025-07-21T10:14:11+00:00
From: redrooster@email.redrooster.com.au
Subject: Your Red Rooster Order Confirmation: 1549-826557-5236
Suspicious URLs: match(outputs('Html_to_text'), 'https?://[^\s"]+')
Risk Level: Low

Quick Threat Intel Links:
ipinfo: https://ipinfo.io/match(outputs('Html_to_text'), 'https?://[^\s"]+')
AbuseIPDB: https://www.abuseipdb.com/check/match(outputs('Html_to_text'), 'https?://[^\s"]+')

(Full details auto-logged in the SOC Incident Log sheet.)

What tools did we use—and why?

Function	Our Solution	Enterprise-Grade Equivalent
Email Monitoring	Power Automate	Defender for O365, Splunk, Sentinel
Log Analysis & Enrichment	n8n + OpenAI	SOAR (XSOAR, Sentinel Playbooks)
Data Integration/Logging	Google Sheets + Apps Script	SIEM (Splunk, Sentinel, ELK, QRadar)
Alerts/Dashboard	Slack, Looker Studio	PagerDuty, Teams, SIEM Dashboards

Enterprise note:
Industry SOCs use SIEM/SOAR platforms with direct integrations (no Google Sheets workaround).

What did I learn?

- **Integrating no-code and pro-code tools for SOC automation**
- **Building modular, upgradeable automations**
- **Using AI for incident triage**
- **Presenting security data for analytics and management**

How can this be used?

- **For SOC teams:** A blueprint for automation with budget-friendly tools.
- **For recruiters:** Proof of automation, integration, and real-world security engineering skills.

See more details, full workflow JSON, and scripts in my GitHub repo:
[cybernishman/soc-automation:**]")**
