Unified SOC Analyst & Threat Hunter Tool - User Manual

1. Introduction

The **Unified SOC Tool** is an all-in-one Graphical User Interface (GUI) written in Python. It is designed to assist Security Operations Center (SOC) analysts and Threat Hunters by integrating:

- **AI-Powered Assistance:** Uses OpenAI (GPT-4) to answer security questions and generate code.
- **Azure Sentinel Integration:** Connects to Azure Log Analytics to run KQL queries.
- **Threat Hunting:** Analyzes PDF reports and log files for indicators of compromise (IOCs) or CTF flags.
- **Automated Reporting:** compiles findings into professional Word documents (`.docx`).

2. Installation & Prerequisites

Step 1: Install Python

You need **Python 3.8 or higher** installed on your system.

- **Windows:** Download from python.org. Ensure you check the box **"Add Python to PATH"** during installation.
- **Mac/Linux:** Python 3 is usually pre-installed. Verify by running `python3 --version` in your terminal.

Step 2: Download the Tool

Save the provided script file as `unifiedsoctool.py` in a folder of your choice (e.g., `Documents/SOC_Tool`).

Step 3: Install Required Libraries

The tool relies on several external Python libraries. Open your Command Prompt (Windows) or Terminal (Mac/Linux) and navigate to the folder where you saved the script.

Run the following command to install all dependencies:

```
pip install openai pypdf azure-identity azure-monitor-query pandas colorama tiktoken python-docx
```

Note for Linux Users: If you encounter errors related to `tkinter`, you may need to install it separately: `sudo apt-get install python3-tk`

3. Configuration & Credentials

To use the tool's full potential, you need to configure access to OpenAI and (optionally) Microsoft Azure.

A. OpenAI API Key (Required for AI features)

1. Go to platform.openai.com.
2. Sign up/Log in and create a new **API Key**.
3. **Recommended:** Set this as an environment variable on your computer named `OPENAI_API_KEY`. The tool will automatically detect it.
    - *Windows Powershell:* `$env:OPENAI_API_KEY="sk-..."`
    - *Mac/Linux:* `export OPENAI_API_KEY="sk-..."`
4. **Alternative:** You can paste the key directly into the tool's **Configuration** tab every time you launch it.

B. Azure Log Analytics (Required for Live Logs)

If you want to query Azure Sentinel/Log Analytics, you need two things:

1. **Workspace ID:** Found in the Azure Portal under your Log Analytics Workspace > Overview.
2. **Authentication:** The tool uses `DefaultAzureCredential`. The easiest way to authenticate locally is to install the Azure CLI and log in:
    - Install Azure CLI: Guide Here.
    - Run command: `az login`
    - Follow the browser prompt to sign in with your Azure account.

4. How to Run the Tool

1. Open your terminal or command prompt.
2. Navigate to the directory where `unifiedsoctool.py` is saved.
3. Run the script:

```
# Windows
python unifiedsoctool.py

# Mac / Linux
python3 unifiedsoctool.py
```

5. Feature Walkthrough (Tab by Tab)

Tab 1: SOC Agent (The Chatbot)

- **Purpose:** Your primary interface for asking questions and querying Azure logs.
- **How to use:**
    1. **Chat:** Type a question like "What is the syntax for a KQL query to find failed logins?" and press **Send**. The AI will reply in the console.
    2. **Azure Integration:**

        - Ensure your **Workspace ID** is set in the Configuration tab.
        - Ask a natural language question about your logs (e.g., "Show me the last 10 heartbeat events").
        - Click **"Generate KQL"**. The AI will write the query for you.
        - Click **"Run Query"**. The tool sends the KQL to Azure and displays the results in the console.

Tab 2: Threat Hunter (File Analysis)

- **Purpose:** analyzing static files (logs, PDFs, text) for hidden threats or CTF flags.
- **How to use:**
    1. **Load File:** Click **"Load File(s)"** and select a PDF report or a text log file.
    2. **Set Focus:** Type what you are looking for in the "Hunt Focus" box (e.g., "SQL Injection", "Base64 strings", or "CTF Flags").
    3. **Analyze:** Click **"Analyze for Threats/Flags"**.
    4. **Review:** The AI will read the file and output a summary of findings, highlighting suspicious parts.

Tab 3: Report Editor (Documentation)

- **Purpose:** Build a report as you work, then export it to Microsoft Word.
- **How to use:**
    1. **Title & Category:** Give your finding a title (e.g., "Suspicious IP Found") and category (e.g., "Network").
    2. **Description:** Write the details of your finding.
    3. **Add Evidence:** Copy-paste logs or KQL queries into the "Evidence" box.
    4. **Add to Queue:** Click **"Add Entry to Report"**. It will appear in the list on the right.
    5. **Export:** When finished, click **"GENERATE WORD DOC REPORT"**. A `.docx` file will be created in your folder.

Tab 4: Incident Report (AI Writer)

- **Purpose:** Generate a formal incident response summary.
- **How to use:**
    1. **Select Template:** Choose an incident type from the dropdown (e.g., Phishing, Malware).
    2. **Generate:** Click **"Generate Report"**.
    3. **Result:** The AI will draft a formal executive summary based on the current session context and the chosen template.

Tab 5: Configuration

- **Purpose:** Manage settings.
- **Settings:**
    - **OpenAI API Key:** Paste your key here if not set in environment variables.
    - **Workspace ID:** Your Azure Log Analytics Workspace ID.
    - **Model:** Select `gpt-4o` (recommended) or other available models.

- **Save/Load Session:** Use these buttons to save your current investigation state (API keys, report entries, chat history) to a JSON file, allowing you to resume later.

6. Troubleshooting Common Issues

Error: `ModuleNotFoundError: No module named '...'`

- **Cause:** You missed installing the dependencies.
- **Fix:** Run `pip install openai pypdf azure-identity azure-monitor-query pandas colorama tiktoken python-docx`.

Error: Azure Query Fails with "CredentialUnavailableError"

- **Cause:** The tool cannot find valid Azure credentials.
- **Fix:** Open your terminal, run `az login`, and sign in. Restart the tool.

Error: "RateLimitError" from OpenAI

- **Cause:** You have exceeded your API quota or have no credits.
- **Fix:** Check your billing status at https://www.google.com/search?q=platform.openai.com.

The GUI looks small or blurry

- **Fix:** This can happen on high-DPI screens. The tool tries to auto-scale, but you may need to adjust your system scaling settings or maximize the window manually.