# Agentic-AI System: Agent Functionality Guide

**Purpose**  
This guide provides a comprehensive overview of the Agentic-AI system agents. Each section includes the agent’s name, function, and a realistic sample input to help security analysts, developers, and automation engineers integrate and operate the system effectively.

## 🔐 1. Cypher Query Generator

* **Function:** Generates Cypher queries for Neo4j databases.
* **Sample Input:**
* Generate a Cypher query to find all users with admin privileges.

## 📄 2. Log Summarizer

* **Function:** Summarizes threat logs for rapid analysis.
* **Sample Input:**
* Summarize the latest firewall logs for suspicious activity.

## 🧩 3. Neo4j Open Connect

* **Function:** Connects and interacts with your Neo4j database.
* **Sample Input:**
* Connect to the Neo4j database at bolt://localhost:7687.

## 🛡️ 4. Mitigation Finder

* **Function:** Finds and recommends mitigations for identified threats.
* **Sample Input:**
* Find mitigation strategies for CVE-2021-44228.

## 📰 5. Article Summarizer

* **Function:** Summarizes articles and documents.
* **Sample Input:**
* Summarize the attached PDF about ransomware trends.

## 🎥 6. YouTube Summarizer

* **Function:** Summarizes YouTube videos for threat intelligence or training.
* **Sample Input:**
* Summarize the key points from this YouTube video: https://youtu.be/example

## 🖼️ 7. Image Summarizer

* **Function:** Summarizes and interprets images relevant to security.
* **Sample Input:**
* Analyze and summarize the attached screenshot of a phishing email.

## 🔥 8. Flame Graph Summarizer

* **Function:** Analyzes and summarizes flame graphs for performance or threat analysis.
* **Sample Input:**
* Summarize the main findings from this flame graph image.

## 🚀 9. Cypher Query Executor

* **Function:** Executes Cypher queries in Neo4j.
* **Sample Input:**
* Run this Cypher query: MATCH (n:User) RETURN n LIMIT 10

## 🧠 10. Threat Generator

* **Function:** Generates detailed threat chains from attack paths.
* **Sample Input:**
* Generate a threat chain for: CVE-2021-44228 > CWE-502 > CAPEC-248 > TTP-001 > MITIGATION-LOG4J-PATCH

## ⚖️ 11. Attack Risk Assessor

* **Function:** Assesses the risk and criticality of threat chains and provides prevention steps.
* **Sample Input:**
* Assess the risk of this threat chain: TTP-001 > TTP-002 > TTP-003 > TTP-004

## 🧭 12. Threat Chain Visualizer

* **Function:** Visualizes threat chains as diagrams or graphs.
* **Sample Input:**
* Visualize the following threat chain: TTP-001 > TTP-002 > TTP-003 > TTP-004

## 🛠️ 13. Incident Response Agent

* **Function:** Suggests and automates incident response actions based on threat chains.
* **Sample Input:**
* Suggest incident response actions for a high-risk ransomware attack chain.

## 🕵️‍♂️ 14. Threat Intelligence Aggregator

* **Function:** Aggregates and correlates the latest threat intelligence.
* **Sample Input:**
* Aggregate the latest threat intelligence on phishing attacks.

## 📥 15. Report Downloader

* **Function:** Downloads comprehensive reports as PDF or Word documents.
* **Sample Input:**
* Download a report summarizing all recent threat chains and risk assessments.

## 🗂️ 16. Mermaid Opener Agent

* **Function:** Opens the Mermaid online diagram tool.
* **Sample Input:**
* Open the Mermaid Live Editor for this diagram code.

## 🧰 Tool: Mitigation Finder

* **Function:** Finds mitigation strategies for threats.
* **Sample Input:**
* Search for mitigations for CVE-2022-12345.

**End of Guide**

For additional support, integrations, or deployment examples, please refer to the Agentic-AI System documentation or contact your system administrator.

| **Agent Name** | **Function Description** |
| --- | --- |
| cypher\_query\_generator | Cypher query generation for Neo4j databases |
| log\_summarizer | Summarizes threat logs for rapid analysis |
| neo4j\_open\_connect | Connects and interacts with Neo4j database |
| article\_summarizer | Summarizes articles/documents relevant to security |
| youtube\_summarizer | Summarizes YouTube videos for threat intelligence/training |
| image\_summarizer | Summarizes and interprets images relevant to security |
| flame\_graph\_summarizer | Analyzes and summarizes flame graphs |
| cypher\_query\_executor | Executes Cypher queries on Neo4j |
| threat\_generator | Generates detailed threat chains from attack paths |
| attack\_risk\_assessor | Assesses risk and criticality of threat chains |
| threat\_chain\_visualizer | Visualizes threat chains as diagrams/graphs |
| incident\_response\_agent | Suggests and automates incident response actions |
| threat\_intelligence\_aggregator | Aggregates and correlates latest threat intelligence |

**System Overview**

* **Purpose:** Centralized security automation and orchestration using specialized agents.
* **Architecture:** Modular, agent-based design with a manager agent delegating tasks to sub-agents.

**2. Main Components**

**Manager Agent**

* **File:** [agent.py](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
* **Class:** [Agent](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) (from [google.adk.agents](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o "))
* **Role:** Orchestrates and delegates user requests to sub-agents.
* **Startup Message:** Displays a message indicating readiness to coordinate tasks.

**Sub-Agents**

* **Location:** [sub\_agents](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ")
* **Imported Agents:**
  + [cypher\_query\_generator](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [log\_summarizer](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [neo4j\_open\_connect](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [article\_summarizer](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [youtube\_summarizer](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [image\_summarizer](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [flame\_graph\_summarizer](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [cypher\_query\_executor](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [threat\_generator](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [attack\_risk\_assessor](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [threat\_chain\_visualizer](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [incident\_response\_agent](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [threat\_intelligence\_aggregator](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
  + [mitigation\_finder](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) (as a tool)

**Agent Functions**

| **Agent Name** | **Function Description** |
| --- | --- |
| cypher\_query\_generator | Generates Cypher queries for Neo4j databases |
| log\_summarizer | Summarizes threat logs |
| neo4j\_open\_connect | Connects/interacts with Neo4j database |
| article\_summarizer | Summarizes articles/documents |
| youtube\_summarizer | Summarizes YouTube videos |
| image\_summarizer | Summarizes/interprets images |
| flame\_graph\_summarizer | Analyzes flame graphs |
| cypher\_query\_executor | Executes Cypher queries |
| threat\_generator | Generates threat chains |
| attack\_risk\_assessor | Assesses risk/criticality of threat chains |
| threat\_chain\_visualizer | Visualizes threat chains |
| incident\_response\_agent | Suggests/automates incident response |
| threat\_intelligence\_aggregator | Aggregates threat intelligence |
| mitigation\_finder (tool) | Finds mitigation strategies |

**3. Workflow**

* **Initialization:** Manager agent displays a startup message.
* **Input Handling:** Accepts user requests via option number or description.
* **Delegation:** Routes requests to the appropriate sub-agent/tool.
* **Mitigation:** Requests involving mitigation are handled by [mitigation\_finder](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ").

**4. Key Files & Imports**

* **Main Manager:** [agent.py](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
* **Sub-Agents:** Each in [agent.py](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)
* **AgentTool:** Used for tool integration (e.g., mitigation finder)

**5. Extensibility**

* **Adding Agents:** Import new agent modules and add to [sub\_agents](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") list.
* **Adding Tools:** Import and add to [tools](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html) list using [AgentTool](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ").

**6. Error Handling**

* **Aggregate Outputs:** [aggregate\_agent\_outputs(input\_data)](vscode-file://vscode-app/c:/Users/Shreya%20Kunda/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") runs each sub-agent and captures errors per agent.

**7. Dependencies**

* **google.adk.agents**
* **google.adk.tools.agent\_tool**
* **Other sub-agent modules**

**8. Entry Point**

* **Run Command:** When executed as main, starts the manager agent.