AI-Driven Cybersecurity Threat Intelligence and Response Platform

This document outlines the architecture of an integrated platform designed for AI-driven cybersecurity threat intelligence and incident response. The platform combines functionalities from two previously separate projects to create a holistic solution for proactive threat monitoring, analysis, and automated response.

1. Overview

The AI-Driven Cybersecurity Threat Intelligence and Response Platform is a highly customizable and modular system leveraging the General Theory of Information (GTI) to create a self-regulating, adaptive cybersecurity solution.

It integrates multiple AI agents, a unified knowledge graph, and a comprehensive configuration layer. This allows businesses to tailor the tool to their specific needs, addressing all aspects of cybersecurity requirements from threat hunting to incident resolution.

2. Core Architectural Components

2.1 Customization Layer (Preliminary Configuration)

This layer allows users to define their organization's specific context, influencing the behavior of all other components.

- **Business Profiling Module:** Gathers information about the business type, operations, hierarchy, compliance needs, past cybersecurity events, risk appetite, etc. This information is used to customize every other component of the system.
- **Interactive Questionnaire:** An interface that asks detailed questions about business operations, regulatory requirements, existing tools, vulnerabilities, and corporate governance.
- **Risk Assessment Module:** Analyzes collected information to create a risk profile that determines priorities in threat intelligence gathering and incident response planning.
- **Framework Recommendation Module:** Recommends specific security frameworks (e.g., NIST, ISO 27001, CIS) based on the gathered data.
- **Customization Output:** Configures threat intelligence focus areas, incident playbooks, policies, and security settings tailored to the organization's needs.

2.2 AI Agents (Microservices Architecture)

These intelligent agents operate as independent microservices to perform specific tasks.

- **OSINT Analyst Agent:** Collects real-time information about emerging threats from open-source intelligence (blogs, forums, news, etc.).
  - **Tools:** WebSearchTool and WebScraperTool for extracting and analyzing relevant cybersecurity data.
- **Data Extraction Agent:** Extracts comprehensive threat data from various sources

identified by the OSINT Analyst.

- **Validation Agent:** Verifies the accuracy, relevance, and completeness of information gathered by the OSINT and Data Extraction Agents. Ensures intelligence meets quality standards before being added to the knowledge graph.
    - **Tool:** NLPTool for processing and analyzing data for consistency and completeness.
- **Knowledge Graph Agent:** Transforms validated threat intelligence into a structured knowledge graph and updates the Neo4j database. Links threat actors, vulnerabilities, tactics, and other related entities.
    - **Tool:** GraphUpdateTool for interacting with the Neo4j database.

2.3 Autopoietic and Cognitive Management

These layers ensure the platform's self-regulation and efficient operation.

- **Autopoietic Network Manager (APM):** Utilizes Kubernetes for provisioning, managing, and self-healing containerized microservices (AI agents). Handles scalability, failover, and container lifecycle management.
- **Cognitive Network Manager (CNM):** Orchestrates data flows and manages interactions between microservices to maintain efficient operations. Uses policies derived from GTI to optimize data flow and dynamically manage workflows.

2.4 Knowledge Graph and Long-Term Memory

The knowledge graph serves as the platform's central repository of threat intelligence.

- **Knowledge Graph Integration: Neo4j** graph database maintains a real-time representation of the cybersecurity knowledge base, including entities such as threat actors, vulnerabilities, tactics, and relationships among them.
- **Associative Memory and Event History:** All interactions and processed intelligence are stored, forming an evolving history of events used to enhance system learning and adaptiveness.
- **Event-Driven Associative Memory:** Maintains an event-driven history of all system interactions, creating long-term associative memory that aids in optimizing response strategies and threat predictions.

2.5 Microservices Design

The platform adopts a microservices architecture for flexibility and scalability.

- **Containerized Services:** Each AI agent runs as an independent microservice within a Docker container.
- **Kubernetes Orchestration:** Used to manage and scale the microservices infrastructure, providing high availability and fault tolerance.
- **API Gateway:** Routes requests to relevant microservices while handling authentication,

rate limiting, and aggregation.

2.6 Adaptive Security Framework Recommendation

The platform provides dynamic guidance on security best practices.

- **Adaptive Engine:** Suggests optimal security frameworks and action plans based on the business profile and current risk landscape.
- **Integration with Customization Layer:** Uses the information gathered during customization to recommend frameworks like ISO 27001 or CIS, adjusting for specific industry needs.

2.7 Incident Detection and Response Module

This module enables proactive threat identification and guided response.

- **Incident Detection:** AI agents continuously monitor the environment, leveraging the knowledge graph and analyzing data for potential threats.
- **Incident Playbook Generation:** Tailored playbooks are generated based on detected incidents, the company's structure, compliance requirements, and risk tolerance, drawing upon predefined templates and the knowledge graph.
- **Containment and Recovery:** Provides guidance on isolation, recovery, and root cause analysis based on previous incidents and the knowledge graph, utilizing predefined response actions.

3. File Structure Diagram for Implementation

```
cybersecurity_saas_tool/
├── config_layer/                  # Preliminary Configuration Layer
│   ├── business_profiling.py          # Business Profiling Module
│   ├── risk_assessment.py             # Risk Assessment Module
│   ├── questionnaire/                 # Questionnaires
│   │   ├── industry_questions.json     # Industry-Specific Questions
│   │   └── compliance_questions.json   # Compliance Requirements
│   └── framework_recommendation.py     # Security Framework Recommendation
│
├── ai_agents/                     # AI Agents (Microservices)
│   ├── osint_agent/                   # OSINT Analyst Agent
│   │   ├── scraper.py                 # Web scraping tools for open-source data
│   │   └── analysis.py                # Threat data analysis (initial)
│   ├── data_extraction_agent/         # Detailed Data Extraction Agent
│   │   └── extractor.py               # Tools for comprehensive data extraction
│   ├── validation_agent/              # Validation Agent
│   │   └── validate.py                # Data validation and enrichment tools
│   └── knowledge_graph_agent/         # Knowledge Graph Agent
│       ├── graph_updater.py           # Updates Neo4j with new threat data
│       └── graph_schema.json          # Schema for graph database
│
├── management_layers/                 # Autopoietic and Cognitive Management
│   ├── autopoietic_manager.py          # Kubernetes Management (APM)
│   ├── cognitive_manager.py           # Workflow Orchestration (CNM)
│   └── event_monitor.py              # Event-driven memory monitoring
│
├── knowledge_graph/                  # Neo4j Knowledge Graph
│   ├── graph_db.py                  # Interfaces for interacting with Neo4j
│   └── schema/                     # Knowledge Graph Schema Definitions
│       ├── entities.json            # Entities Definitions
│       └── relationships.json         # Relationships Definitions
│
├── incident_response/                # Incident Readiness and Response
│   ├── detection.py                 # Threat detection modules
│   ├── response_playbooks/           # Incident Playbooks
│   │   ├── playbook_template.json      # Base template for playbooks
│   │   └── tailored_playbook_gen.py    # Tailors playbooks for specific incidents
│   └── recovery.py                  # Containment and Recovery actions
│
├── adaptive_security/               # Adaptive Security Framework Engine
│   ├── framework_engine.py           # Recommends security frameworks
│   ├── compliance_mapping.json        # Maps compliance to business profile
│   └── maturity_roadmap.py           # Security maturity enhancement plans
│
├── api_gateway/                     # API Gateway
│   ├── gateway.py                   # Handles routing and aggregation
│   └── auth.py                     # Authentication and rate limiting
│
├── docker/                         # Docker Configurations
│   ├── Dockerfile                   # Base Docker configuration
│   └── docker-compose.yml            # Docker Compose for multi-container setup
│
├── kubernetes/                      # Kubernetes Configurations
│   ├── deployment.yaml               # Deployment specifications
│   ├── service.yaml                 # Service definitions
│   └── autoscaler.yaml               # Horizontal Pod Autoscaler settings
│
└── docs/                          # Documentation
    ├── user_guide.md                # User guide for customization and usage
    ├── architecture_overview.md       # Overview of architecture and components
    └── developer_guide.md            # Guide for developers working on the project
```