SOFTWARE SURFERS

11/26/2024

Autonomous AI Agent Execute of MITRE ATT&CK Framework - EyeSpy AI

Team Leader: Oluwatobiloba Lawuyi

Kiara Nichols, Nia McNeal, Medinat Lawal, Christian Earle

Is this an industry-based project? YES

Table of Contents

Project Description

By 2025, it is expected that $10 trillion a year could be stolen by cybercriminals. Nowadays, advanced automated security solutions are essential because they allow us to detect and respond to incidents faster, reduce security workload, etc. Our solution is to automate the detection of threats and defend against various cyber attacks with our system EyeSypy AI using the MITRE ATT&CK wireframes. MITRE ATT&CK is a collection of information about tactics and techniques that hackers use based on real-life cases. What MITRE ATT&CK does is help companies and the government build strategies to protect against attacks and create tools to defend against those attacks. MITRE ATT&CK gives details on different attacks and techniques to combat those techniques. Some attacks listed are Reconnaissance, Resource Development, Initial Access, Execution, Persistence, Privilege Escalation, Defense Evasion, Credential Access, Discovery, Lateral Movement, Collection, Command and Control, Exfiltration, and Impact. Each tactic has a range from 8 to 43 techniques to combat it. For example, Reconnaissance is a tactic that the attacker is trying to get information they can use to plan future operations. Some techniques to combat reconnaissance are vulnerability scanning, scanning IP blocks, and scanning hardware and software. As the Blue Team, we want to evolve this agent by simulating Red Team tactics on our system and understand how they can be countered using various defensive strategies.

I. Background and Survey of Existing Work

Two similar platforms that are available are SafeBranch and Cortex Extended Detection and Response XDR.

SafeBrach is a breach-and-attack simulation platform that allows users to continuously validate all layers of security by simulating real-world attacks to identify gaps in the user's controls, prioritize remediation, and reveal actual risks. What a user can do with SafeBranch is simulate over 25,000 attacks

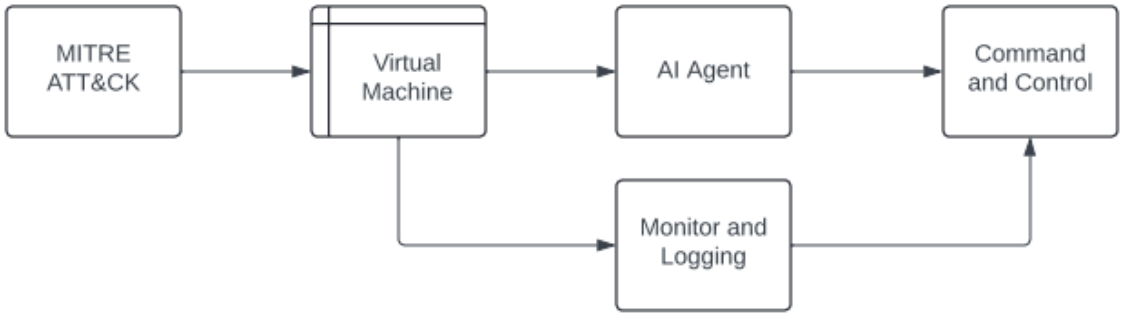
to test existing security controls. New threats are continuously being added to the platform test. What a user can do with SafeBreach is search, filter, and select attacks to run, execute preconfigured attack scenarios to replicate common threats, integrate with internal threat providers, and create customized attacks in the user's repertoire. On the defensive end, SafeBreach accurately presents the level of risk, presents a prioritized list of mitigation plans, and provides a cost assessment of risk reduction efforts.

Cortex Extended Detection and Response (XDR) is a platform that enhances security and stops attacks with complete visibility and analytics. Some features include proven endpoint protection, laser-accurate detection, and lightning-fast investigation and response. They block advanced malware, exploits, and file-less attacks for proven endpoint protection. The agent stops threats with Behavioral Threat Protection, AI, and cloud-based analysis. With laser-accurate detection, Cortex XDR uses machine learning to profile behavior and detect anomalies in the system. It lets users spot adversaries attempting to blend in with normal users. With its lightning-fast investigation and response, the agent investigates threats by getting a complete picture of each attack with incident management in a swift manner. Cortex XDR includes endpoint security, ML-driven threat detection, incident management, automated root cause analysis, deep forensics, flexible responses, and extended threat hunting.

II. System Overview

The system is made up of five main parts: the MITRE ATT&CK Framework, Virtual Machine, AI Agent, Command and Control Module, and the Monitor and Logging System. The MITRE ATT&CK Framework outlines various cyberattack techniques. The Virtual Machine serves as an environment where these techniques can be executed against our security setup. The AI Agent defends against these techniques automatically and adjusts its approach according to the feedback it receives. The Command and Control Module allows for the management of the AI, making it possible to modify defense strategies

on the fly. The Monitor and Logging System keeps track of all activities, providing essential data to analyze the effectiveness of defensive moves. Altogether, this setup creates a realistic defense scenario, making it useful for testing and improving cybersecurity strategies in a controlled setting.



III. Stakeholders

A stakeholder for this project is the Cybersecurity Operations Center (CSOC) members at Sandia, although Sandia as a whole is the main stakeholder. The CSOC is in charge of monitoring and defending against cyber threats, so utilizing an AI agent that uses the MITRE ATT&CK framework would help them catch threats faster and come up with better defensive strategies. This system would also let them test and improve their defenses against a variety of simulated attacks, which would give them valuable insight. The company as a whole would benefit. Overall, focusing on strengthening the defensive side would help make Sandia's security stronger and more prepared for any future cyber threats.

IV. Function and Nonfunction Requirements

EyeSpy AI must have functional and nonfunctional requirements defined to depict the behavior of how the system will operate and describe how the system must perform its tasks. These requirements are essential in the system's development. Functional requirements are incredibly important in outlining the

scope of the system and ensuring user satisfaction. Nonfunctional requirements, conversely, are significant in ensuring the quality meets the user's standards and ensures that the system is user-friendly. Furthermore, the combination of both requirements will guide the Software Surfers in designing and developing the EyeSpy AI agent effectively.

Functional Requirements

Subject	Requirement
Integration of MITRE ATT&CK Framework	The system must use automation to execute the MITRE ATT&CK tactics and sub-techniques.
Large Language Model (LLM) Implementation	The system must assist the user in understanding the MITRE ATT&CK tactics and sub-techniques.
Automation of Attacks	The system must automate the process of 3 selected MITRE ATT&CK techniques in a controlled virtual environment.
Live Attack Demonstration	The system must provide a live demonstration of how each MITRE ATT&CK technique and deliver real-time feedback on its success or failure.
Reliability	The system should have no unexpected failures or crashes.
Security	This system will ensure that the virtual attack

	environment is isolated from the host.
Alert System	The system will generate alerts when it detects attacks.

Nonfunctional Requirements

Subject	Requirement
Performance	The system should produce accurate and consistent results according to the user's input.
Compatibility	The system should be compatible with major operating systems to execute attacks.
Ethical	This system must ensure all attacks are conducted in a controlled manner.

V. Data Usage and Management Plan

EyeSpy AI will be designed with a structured and prioritized approach to ensure efficient data handling, security, and threat detection. The first phase focuses on Data Ingestion and Storage, where

Snowflake Snowpipe will be implemented for continuous data ingestion, alongside real-time streaming from critical data sources. This phase will include basic data cleansing and validation.

Next, in Data Processing and Analysis, a multi-cluster Snowflake warehouse will ensure 24/7 availability, with a basic stream processing framework for immediate responses to known threats through rule-based systems.

Security and Monitoring will be integral, with end-to-end encryption and strict access controls in place. Real-time monitoring of data pipelines, storage, and processing will be established, along with automated alerts for any critical system performance issues.

For Model Development and Serving, initial AI/ML models for threat detection will be developed and deployed, with a basic infrastructure for model serving and automated retraining.

Threat Response and Decision Making will involve a decision trail for all automated decisions and a simple gradual response system based on threat severity. Security orchestration tools will also be integrated to mitigate automated threats.

In terms of Data Management and Backup, continuous incremental backups of critical data will be implemented, and backup testing and validation will be conducted.

To ensure the system is adaptable, System Refinement and Version Control will include version control for AI models and decision logic, as well as automated documentation of system changes.

Finally, Testing and Human Oversight will play a key role. Thorough testing of all automated systems will be done before deployment, and a basic alert system will notify human operators of significant issues, with a 24-hour human check-in and oversight process.

This foundational setup prioritizes the infrastructure needed for data handling, security, model development, and threat response, while also ensuring that key testing and human oversight measures are in place for future phases of expansion.

VI. Requirements Analysis and System Specifications

This project aims to learn about the MITRE ATT&CK framework and find ways to enhance its defensive capabilities. It involves implementing 3 MITRE ATT&CK tactics or sub -techniques in an automated process using the help of an LLM (Large Language Model). We're going to create a command line interface or GUI to demonstrate a live attack. We'll do that by creating an environment for these attacks using virtual machines or other systems we decide to use. In this project, we'll be playing as the Blue team which is the defensive side. The main purpose of this is to identify, monitor, and mitigate the attacks while using defensive security mechanisms.

We will be implementing three specific tactics/sub-techniques within the MITRE ATT&CK framework to simulate real-world attack scenarios. A Large Language Model (LLM) will assist in automating the simulation of attacks, allowing for dynamic variations of attack methods. The execution will be demonstrated live through a command-line interface or GUI for ease of monitoring and analysis. This can be seen as an interactive demonstration These attacks will be in controlled virtual environments monitored by our team that will mimic the structure and possible vulnerabilities of an actual system. Blue Team, our focus will be on the identification and defense mechanisms against these attacks, looking into how these systems can detect and respond to them in real time.

VII. Outline of Development Plan

The development plan for the project will follow Agile methodologies, specifically Scrum, with Kanban used for team accountability and progress tracking. The team will operate on 2-week Scrum sprints, with daily standups where Kanban will be used to visualize workflow and task statuses. This will allow the team to remain flexible, adapting to the demands of work as needed, especially given that everyone is a college student with varying schedules.

Kanban will also be used for handling daily operational and maintenance tasks, helping the team address issues as they arise without overcommitting to fixed amounts of work.

The team structure and roles are as follows:

- Tobi, the Project Lead, will handle overall project coordination, final decision-making, stakeholder presentations, technical implementation for large language models (LLMs), and user interface development.
- Christian, the Technical Lead, will oversee the technical implementation, conduct code reviews, and make decisions on code architecture.
- Kiara, Cybersecurity Specialist, and Documentation Lead will focus on MITRE ATT&CK research, implementation, and maintaining documentation.
- Nia, Cybersecurity Specialist, and Data Integration Lead, will work on security testing, data preprocessing, validation, and LLM integration.
- Medinat, Technical Specialist and AI Integration Lead will handle threat modeling, test plan creation, and AI prompt engineering.

The plan includes three weekly stand-ups lasting about 15 minutes to keep communication open and progress clear. There will also be a weekly progress review for an hour, with every other week marking the end and start of a sprint. A weekly mentoring update and review will be scheduled for 45 minutes. The plan allows for flexibility, including a break during spring break, and allows us to finish the project on time, and account for unforeseen circumstances.

The team will practice unit testing for all components to ensure functionality and quality. Version control will be maintained using GitHub and Git Desktop, ensuring collaborative coding. Testing for AI models will occur in a sandbox environment, and command-line development will be used for access.

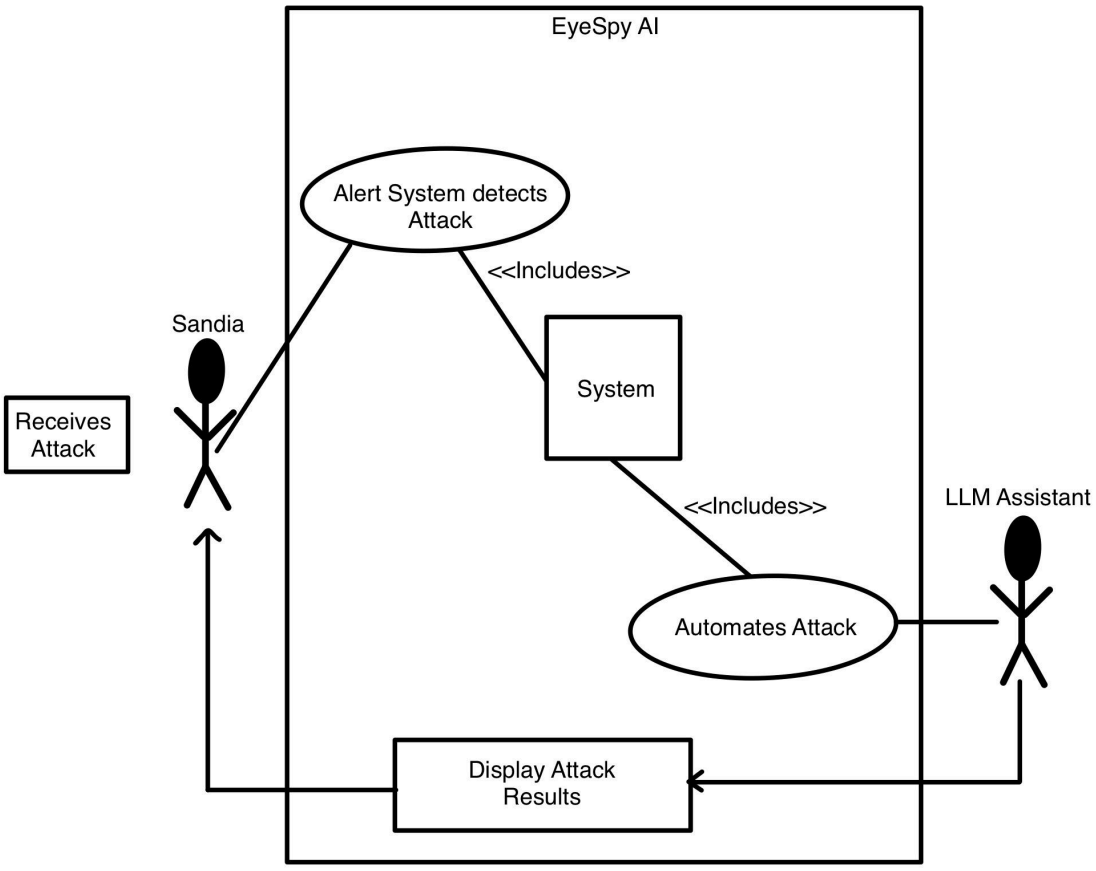
Throughout the project, technical documentation will be maintained, with special attention given to documenting AI prompts and instructions for effective usage. This plan ensures efficient, flexible development while keeping accountability through Scrum and Kanban, with clear roles and a strong focus on security, AI integration, and version control.

VIII. Design

Use Case and Sequence Diagrams

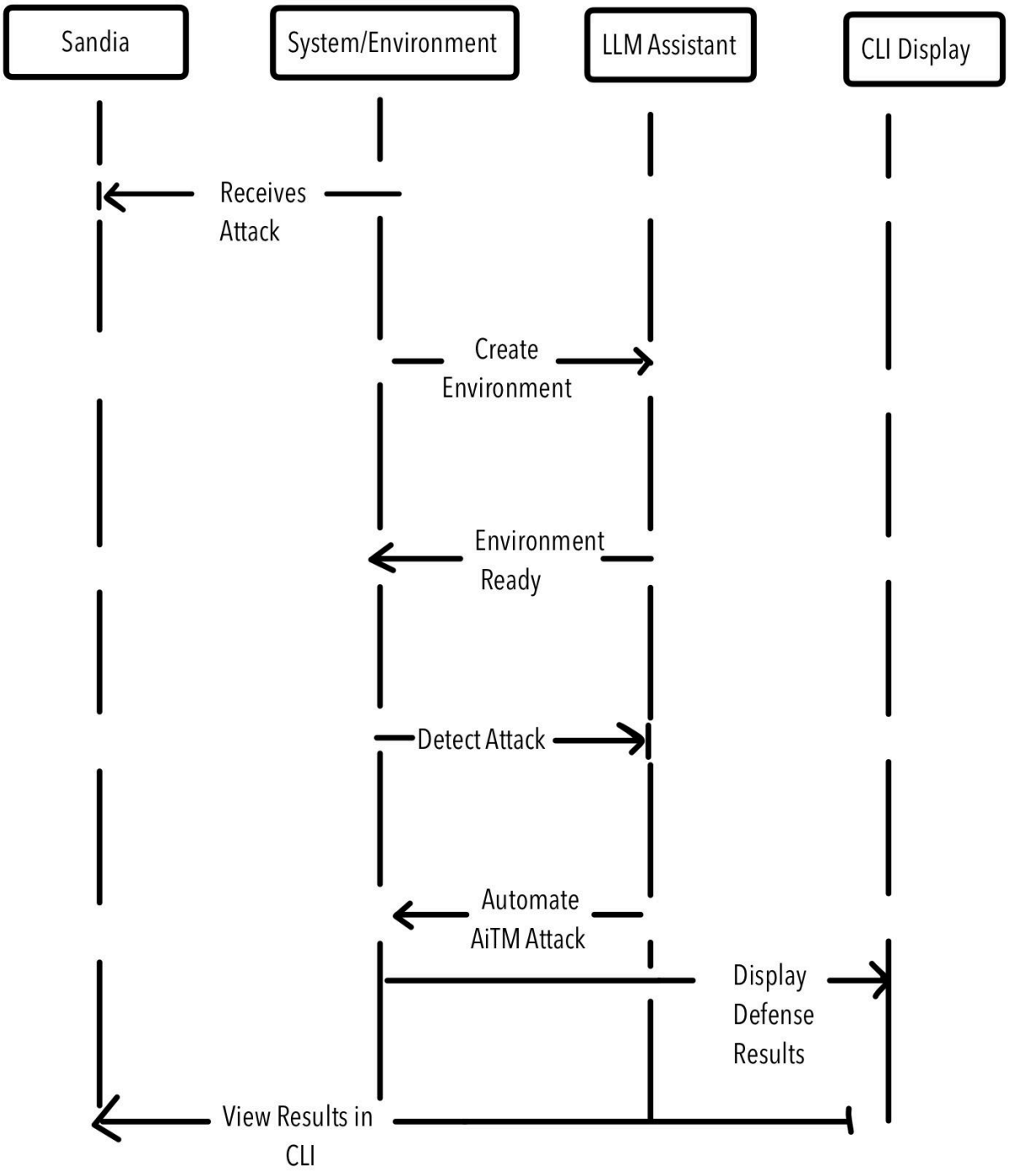
This use case diagram shows interactions that will be used for handling multiple attack types (Content Injection, Remote Access, AiTM) using our EyeSpy AI system. It depicts that Sandia will initiate attacks and the system/environment will receive those attacks. The LLM Assistant can automate detection and defense for each attack type, and the results of the attacks are displayed for the user to see and be able to analyze.

Use Case Diagram



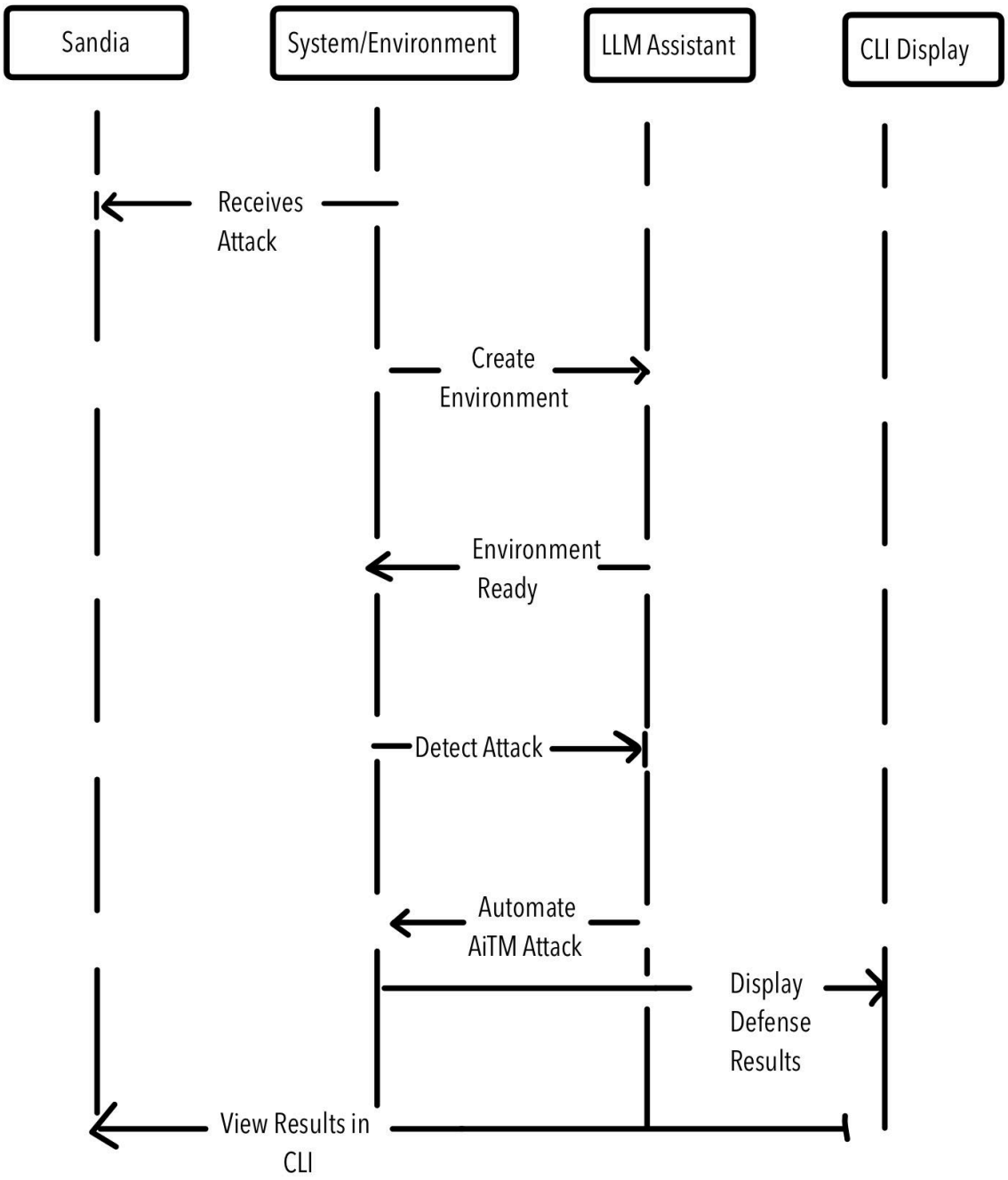
The diagram below illustrates the process used for Content Injection Attacks. We can see the attack flow starting at the top with Sandia to the system/environment, where the environment is set up, it detects the attack, and will then automate a response using an LLM Assistant. The results will be displayed on the CLI.

Sequence Diagram: Content Injection Attack



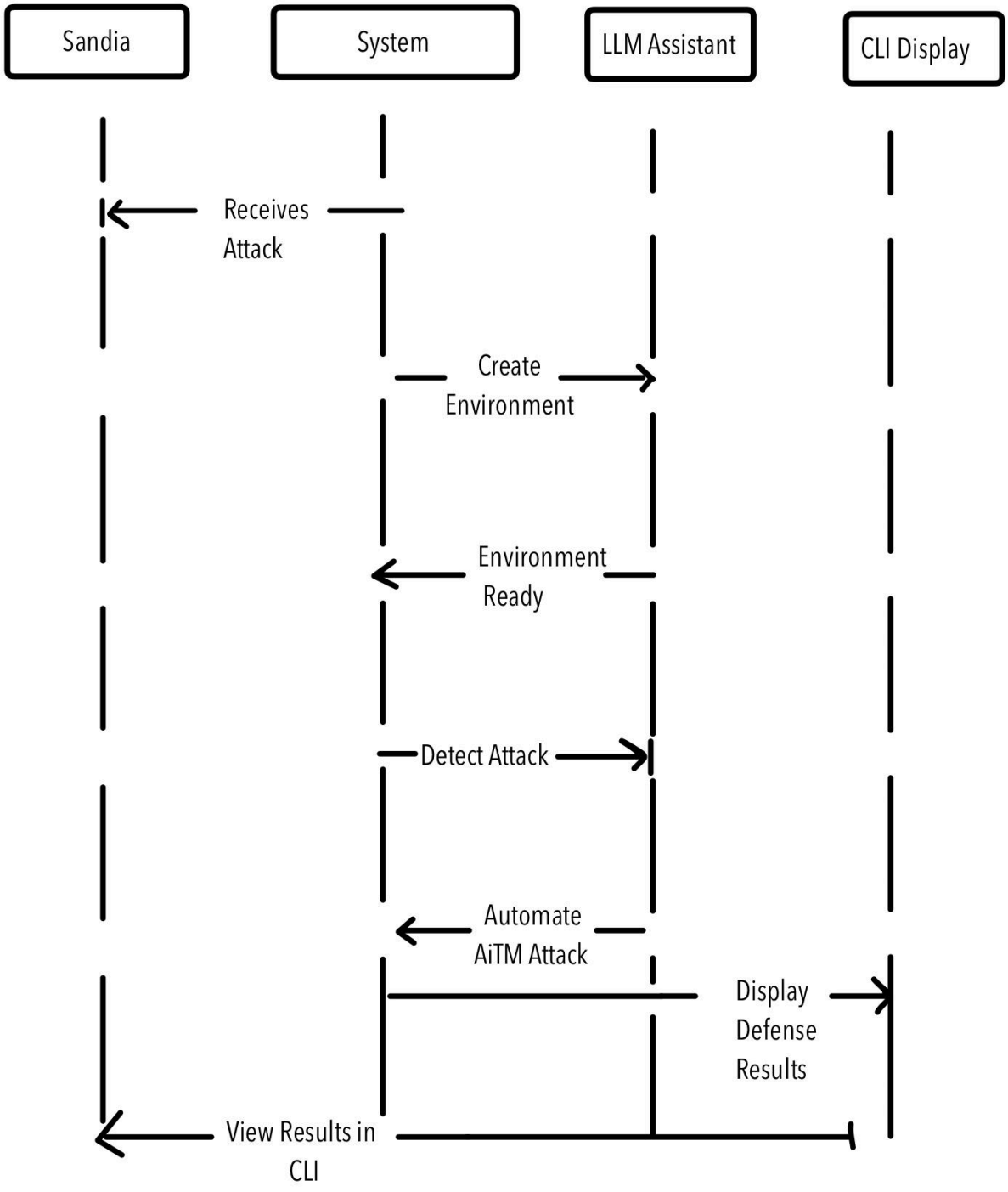
This diagram represents the flow in which Remote Access Attack would take. It will start with Sandia sending the attack to the system/environment, which then prepares the environment. The LLM Assistant will then automate the response needed for the attack, and everytime the results are displayed on the CLI. This process involves attack detection, environment setup, automation, and defense result visualization.

Sequence Diagram: Remote Access Attack



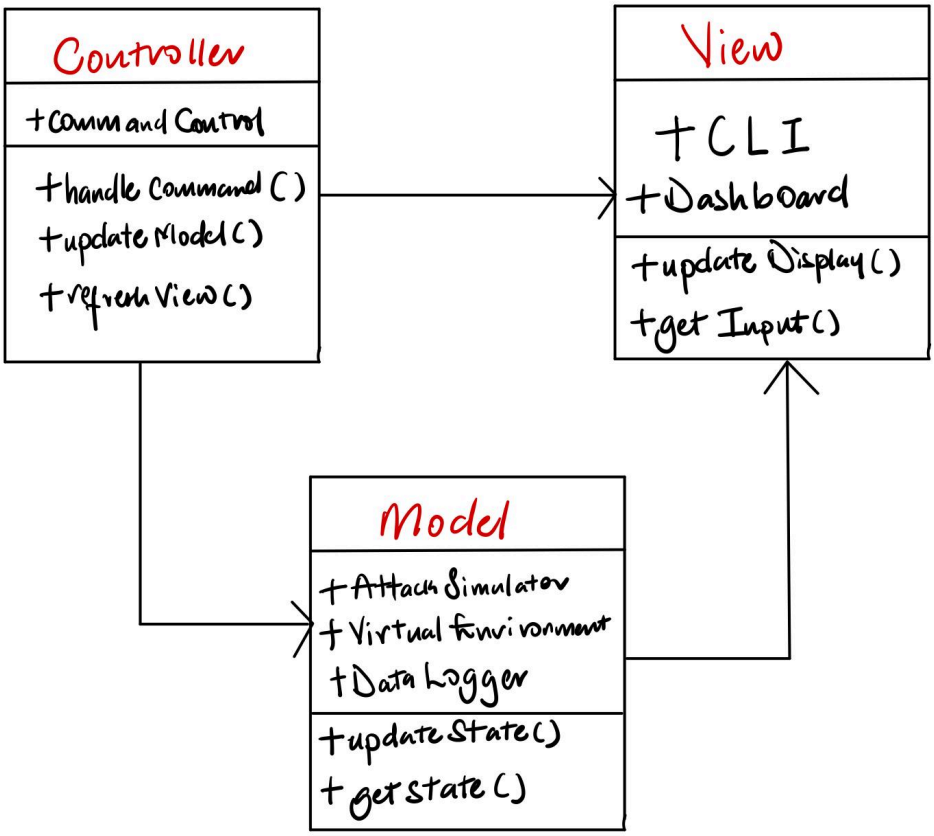
The diagram depicts an Adversary-in-the-Middle (AiTM) attack process. It involves Sandia initiating the attack, the system/environment will handle the attack, and the LLM Assistant automates the attack response. The attack is received, an environment to defend the attack is created, and results are displayed on the CLI, showing the defense status.

Sequence Diagram: Adversary in the Middle Attack



Model View Controller Architecture

The Model-View-Controller (MVC) architecture in the EyeSpy AI system organizes components to promote the separation of concerns. The **Model** manages business logic and data, including attack simulation and environment states. The **View** handles user interfaces, displaying real-time feedback and system metrics, while the **Controller** processes user commands, orchestrates data flow, and ensures seamless communication between the Model and View. This design enhances maintainability and scalability.



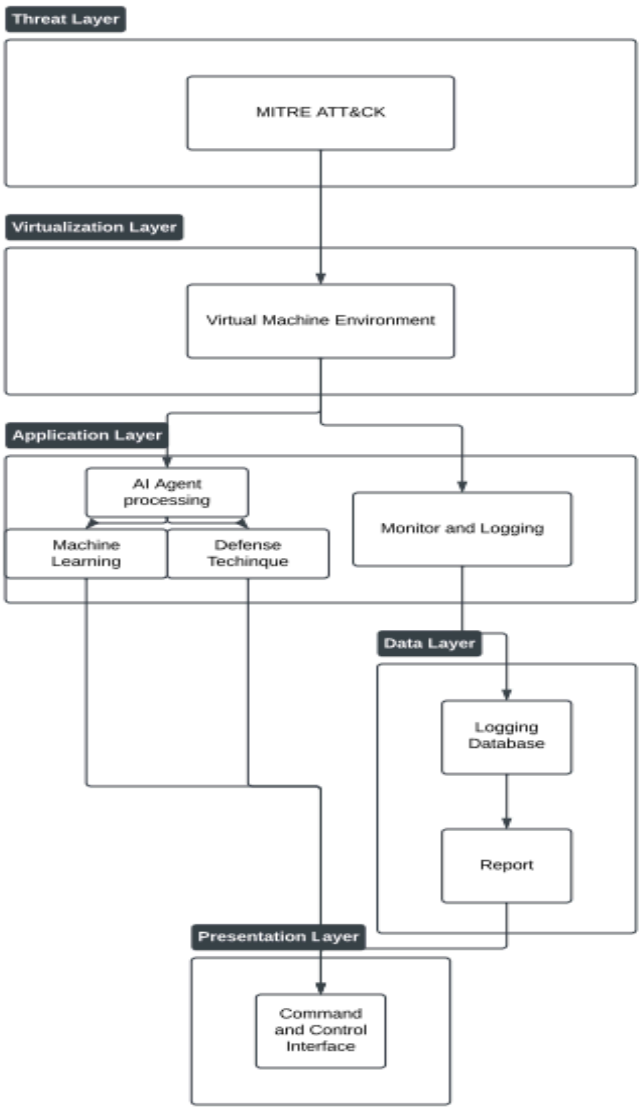
Model View Controller Pattern

Name	**MVC (Model-View-Controller)**
Description	The MVC pattern separates concerns in software design by dividing the application into three interconnected components: Model, View, and Controller. The Model encapsulates the business logic and data of the application, such as the AttackSimulator, VirtualEnvironment, and DataLogger. The View manages the presentation layer, which includes components like CLI and Dashboard for displaying real-time feedback and user interfaces. The Controller handles user input, processes commands, and communicates updates between the Model and View to ensure system functionality and interaction flow.
Example	The EyeSpy AI system's architecture, as shown in the diagram, uses MVC for structured design. Controllers manage simulation commands, the Model handles data and states, and the View ensures proper output and user experience.
When Used	MVC is suitable when there is a need to maintain a clear separation between the application's data management and its user interface, especially in scenarios where the presentation and data interactions may change or evolve over time.
Advantages	This pattern supports flexible data handling and simplifies updates, allowing independent modifications to data logic or user interfaces. It enhances maintainability and scalability, especially for complex applications.
Disadvantages	Implementing MVC can increase code complexity, especially for simple interactions or applications. Additional design layers may also complicate debugging and extend development time.

Layered Architecture

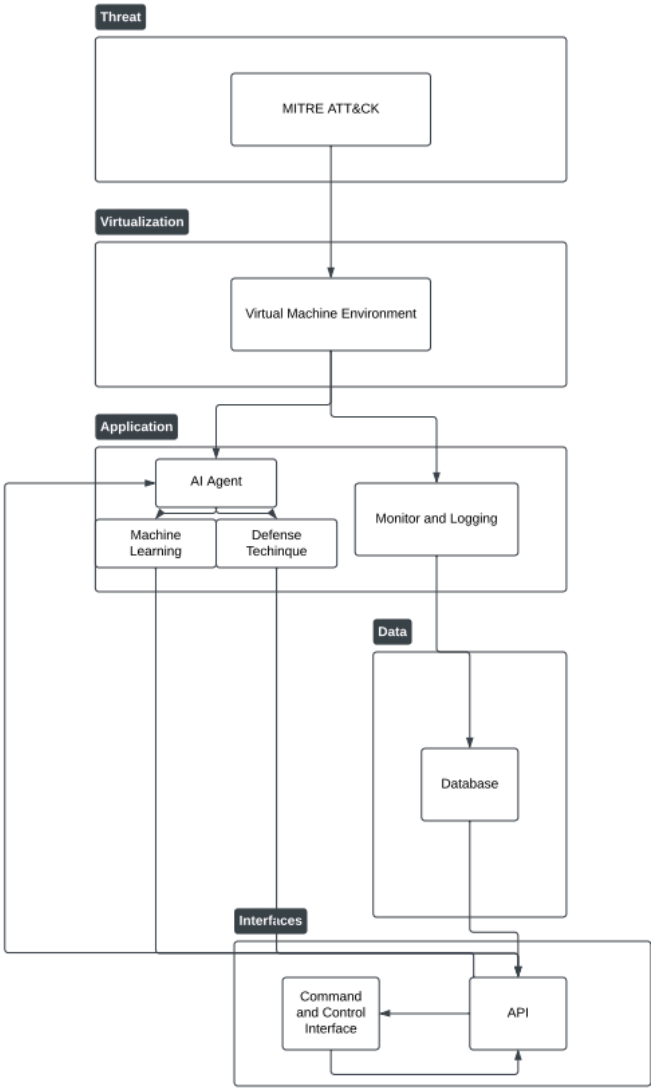
This diagram represents a multi-layered cybersecurity architecture incorporating various components and functionalities across distinct layers. At the top, the Threat Layer includes the MITRE ATT&CK framework, which is the adversary's tactics and techniques. This feeds into the Virtualization Layer, where a virtual machine environment simulates and isolates the processing environment.

The Application Layer is responsible for AI agent processing and monitoring/logging activities, which are essential for real-time analysis and response. Data collected is stored in the Data Layer, specifically in a logging database, which is used to generate reports for further insights. Finally, the Presentation Layer provides a command and control interface, allowing users to interact with and manage the system based on gathered intelligence and analysis. This layered structure enhances security by compartmentalizing functions and enabling detailed monitoring and control.



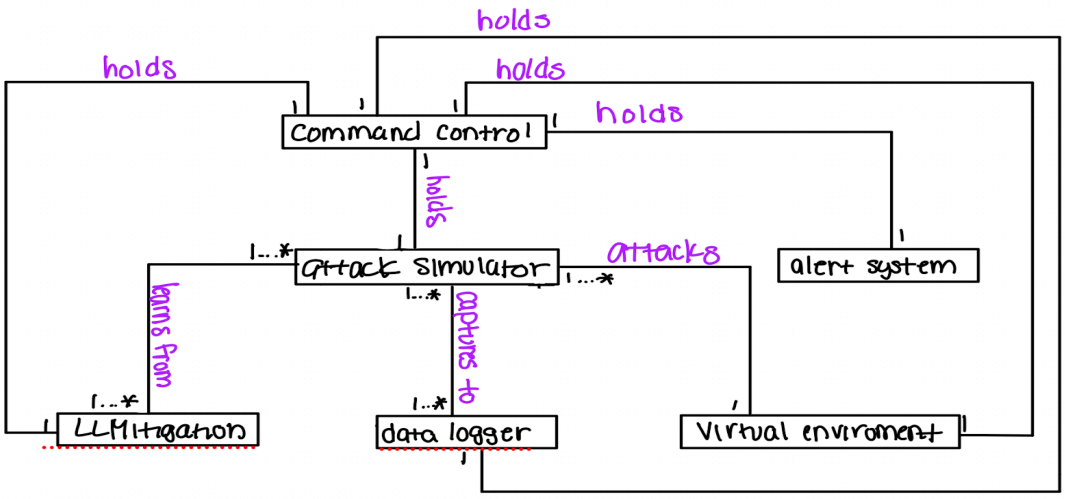
Generic System Architecture

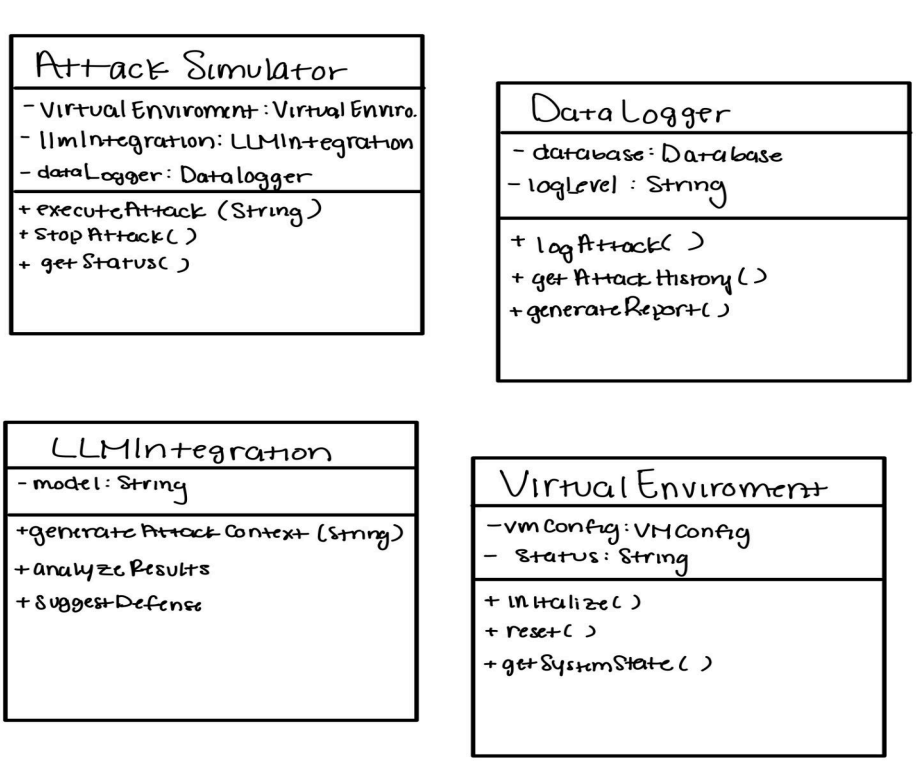
This cybersecurity framework leverages virtualization and artificial intelligence to monitor, detect, and respond to cyber threats, with a strong reliance on the MITRE ATT&CK framework for threat intelligence. The use of a virtualized environment, AI-driven defense techniques, and an organized database ensures the system can handle and respond to attacks dynamically. The command and control interfaces and API support secure and efficient management, enabling administrators to maintain control over the system and integrate it with other tools as needed.

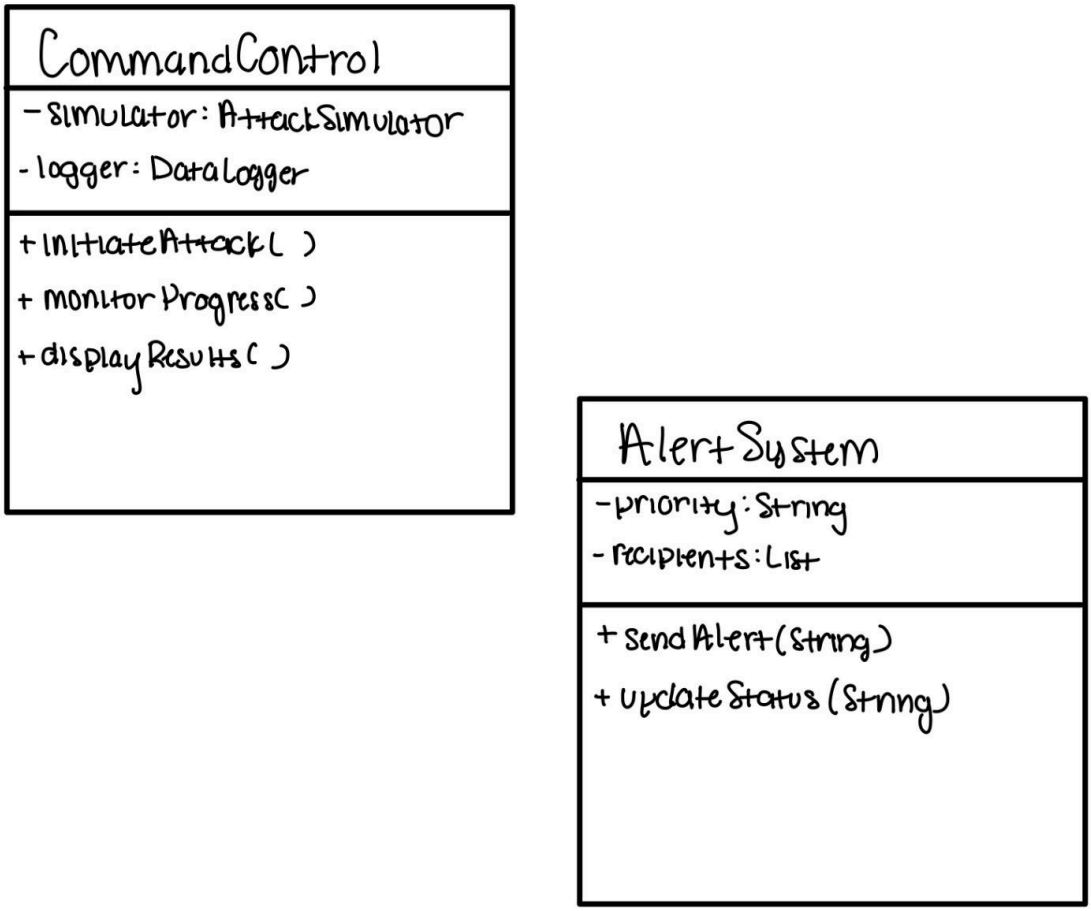


Relationship-Class Diagrams

This UML class and association diagram demonstrates how each class in our system interacts and connects to each other.







IX. Module and Sub-Module Design

The EyeSpy AI system is organized into three modules: Attack, Security, and Monitoring. The Attack Module consists of components like the AttackSimulator, which executes MITRE ATT&CK techniques and interfaces with the VirtualEnvironment for isolated simulation control, and MITREIntegration for technique validation and report generation. The Security Module includes Authentication and Authorization for user management and Encryption to secure data. The Monitoring Module features a DataLogger for event tracking, an alert system to notify of anomalies, and a ReportGenerator for analytics.

Function Description

The AttackSimulator class provides core methods, such as executeAttack(), for initiating attacks with success checks and stopAttack() to manage operations. The virtual environment component initializes and resets system states. Security functions include login() for user sessions and checkPermission() for role validation, while the Monitoring Module logs events, triggers alerts, and generates reports to ensure comprehensive data tracking and threat response.

X. Requirement Validation Plan

The validation strategy ensures thorough testing at various levels. Unit testing will provide 95% coverage, focusing on individual functions, while integration testing will validate data flow and module interactions. System testing will assess end-to-end operations, emphasizing performance and security. Success criteria include zero critical vulnerabilities, a response time under two seconds, and a task completion rate above 90%, with responsibilities in various sections assigned to all team members.

XI. Risk Assessment and Planning

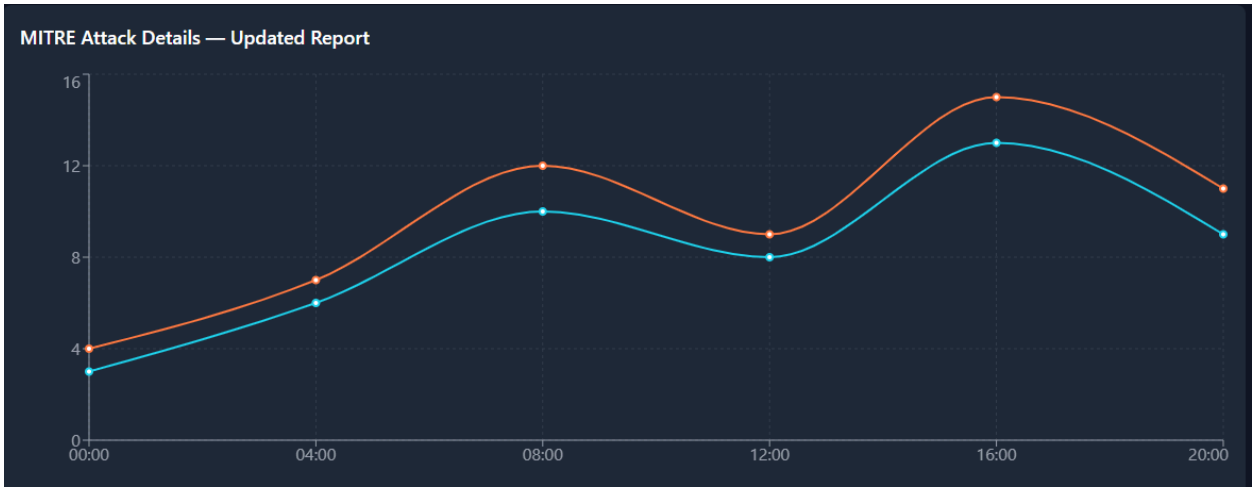
A comprehensive risk matrix categorizes risks by severity and probability, with strategies for mitigation. High-impact risks, like data breaches, are addressed through encryption and audits, while system failures are countered with redundancy plans. Technical, security, and operational risks are continually reviewed and mitigated with actions like load testing, resource monitoring, and fallback mechanisms, ensuring robust system performance and security.

XII. User Interface

The user interface for EyeSpy AI will be an interactive dashboard designed to monitor and analyze attack patterns and events in real time. The interface includes visualizations of event/attack details, integration with MITRE Attack frameworks, and performance rate metrics. It features updated reports, a live monitoring system for threat detection, and a breakdown of attack occurrences by type. EyeSpy AI combines analytics with interactive visual elements to enhance situational awareness and support proactive threat management. We will explain more about each component and how they will be used for this project.

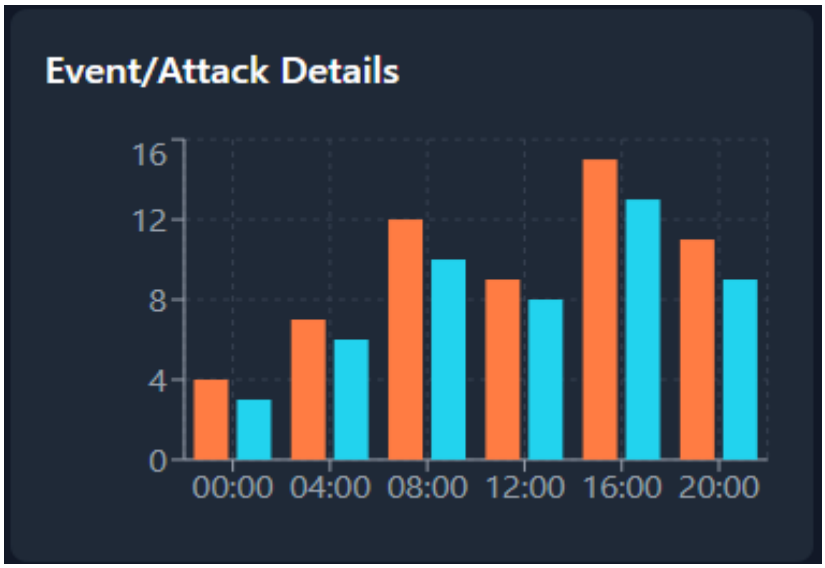
XIII. MITRE Attack Details

This chart represents data related to MITRE Attack Details, tracking two different metrics over time, we can follow these metrics by looking at the orange and blue lines. The X-axis shows the timeline from midnight (00:00) to 8:00 PM (20:00), while the Y-axis measures the quantity or intensity of the tracked metrics, ranging from 0 to 16. The data highlights activity trends, with both metrics showing a similar increase in the early hours of the day, it peaks around noon before it declines later in the day. This suggests that the most critical period for activity, whether it's attack frequency or defense response, occurs between 8:00 AM and 12:00 PM. The graph is a part of our security dashboard, designed to monitor trends, compare the effectiveness of defenses, and identify anomalies.



XIV. Event/Attack Detail

The "Event/Attack Details" chart displayed is a critical component of our cybersecurity testing platform, providing a visual comparison of threats faced versus threats mitigated over a specific time period. The orange bars represent the volume of threats encountered at different intervals, while the blue bars indicate the corresponding threats successfully mitigated. This data allows security teams to assess the platform's effectiveness in real time, identify patterns in attack frequency, and pinpoint any timeframes where the mitigation rate is suboptimal. By leveraging this insight, the platform helps optimize response strategies, enhance rule configurations, and improve the overall accuracy of detection mechanisms, ultimately strengthening the organization's security posture.



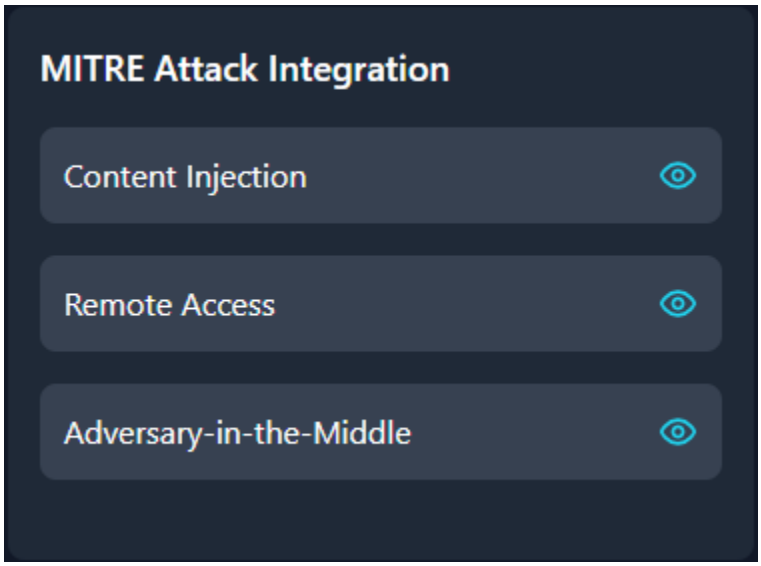
KEY

Threats Faced - Orange

Threats Mitigated - Blue

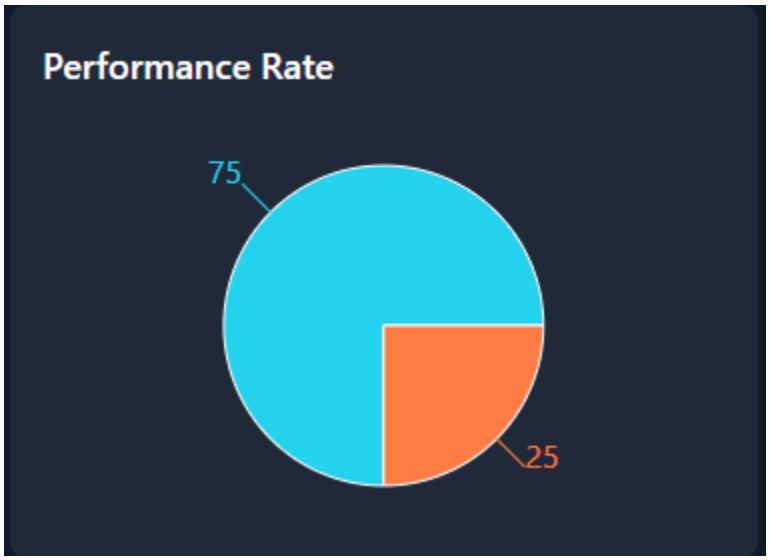
XV. MITRE Attack Integration Interface

The MITRE ATT&CK Integration Interface will seamlessly connect the MITRE ATT&CK framework to the EyeSpy AI interface. The framework will act as the structured point of interaction, mapping techniques, and sub-techniques from the defined MITRE ATT&CK tactics, enabling the system to output specific defense and mitigation context upon request. Furthermore, this interface will interpret threat information and map actions for users of the EyeSpy AI system. After processing user commands, the EyeSpy AI interface will display attack data while identifying relevant tactics and techniques that match the observed behavior. The MITRE ATT&CK Integration Interface will enhance the user's understanding of the attack by providing a mapped context of the attack's tactics and techniques.



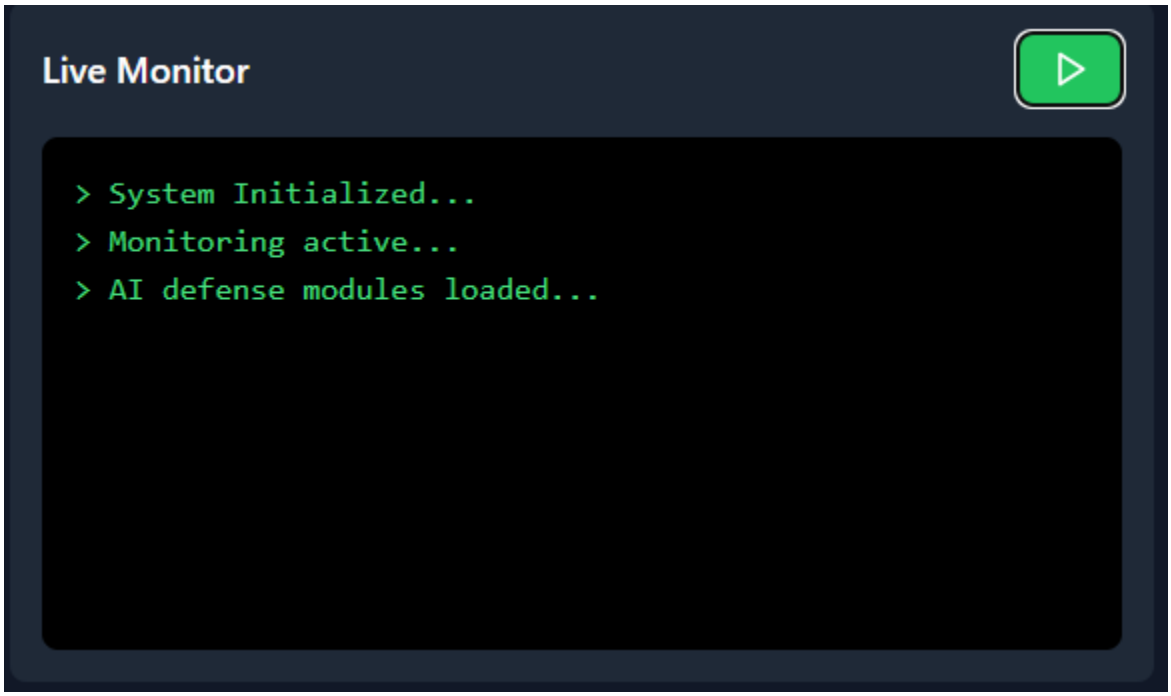
XVI. Performance Rate

The performance rate monitor plays a crucial role in ensuring the accuracy and reliability of threat detection mechanisms by continuously refining their performance. It minimizes false positives by identifying and addressing overly strict detection rules or misconfigured settings that cause benign activities to be flagged as threats. This reduces wasted resources and prevents alert fatigue among security analysts. Simultaneously, it combats false negatives by analyzing historical data, incorporating threat intelligence, and identifying gaps in detection mechanisms, ensuring real threats are not overlooked. Through adaptive learning, the monitor fine-tunes algorithms and rules based on real-time feedback and emerging patterns, making the system more intelligent and responsive to evolving threats.



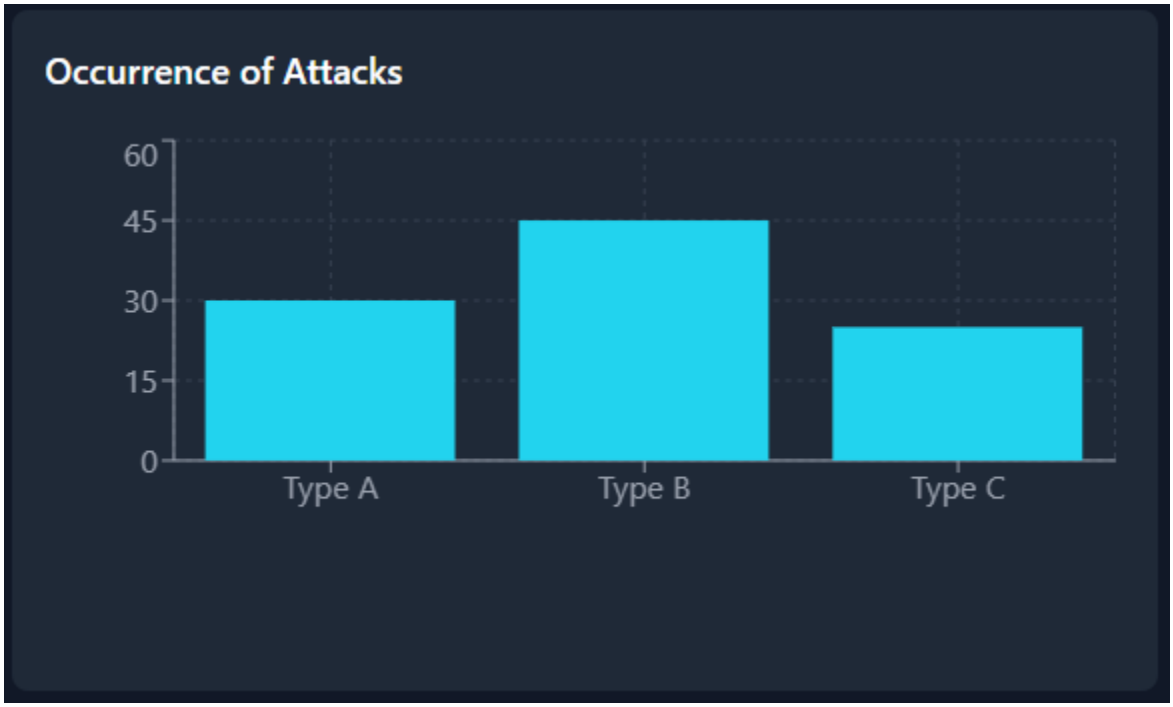
XVII. Live Monitor

This interface represents a Live Monitor displaying system initialization and status updates for our AI-based defense system. The displayed text shows that the system has been successfully initialized, active monitoring is in progress, and the AI defense modules have been loaded. The green text helps provide real-time feedback, confirming that the system is operational and ready to detect and respond to potential threats it receives. The green play button on the top right of the interface serves as a control to activate or can also pause monitoring whichever is needed. This live monitoring feature is crucial for providing immediate insights into system readiness and operational status. It is a user-friendly tool for tracking the AI's functionality at a glance.



XVIII. Occurrence of Attacks

The Occurrence of Attacks panel on the dashboard displays the number of times an attack has been made to the system in a bar chart format. Type A, Type B, and Type C on the x-axis will focus on the three main attacks we decided with the Red Team: Content Injection, Adversary-In-The-Middle, and Remote Access. The y-axis is simply the total number of times those attacks occur.



References

ATT&CK Training | MITRE ATT&CK®. attack.mitre.org/resources/learn-more-about-attack/training.

"Kanban Vs Scrum | Atlassian." *Atlassian*, www.atlassian.com/agile/kanban/kanban-vs-scrum.

MITRE ATT&CK®. attack.mitre.org

"Cortex XDR- Extended Detection and Response." *Palo Alto Networks*, www.paloaltonetworks.com/cortex/cortex-xdr.

Cybercrimemag. "Cybercrime to Cost the World $10.5 Trillion Annually by 2025." *Cybercrime Magazine*, 27 Apr. 2021, cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/#:~:text=Special%20Report%3A%20Cyberwarfare%20In%20The%20C%2DSuite.&text=Cybersecurity%20Ventures%20expects%20global%20cybercrime,%243%20trillion%20USD%20n%202015

GeeksforGeeks. "Functional vs. Non-Functional Requirements." GeeksforGeeks, October 11, 2024. https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/.

Ipa-Lab. "GitHub - Ipa-lab/hackingBuddyGPT: Helping Ethical Hackers Use LLMs in 50 Lines of Code or Less.." *GitHub*, github.com/ipa-lab/hackingBuddyGPT.

Matrix - Enterprise | MITRE ATT&CK®. attack.mitre.org/matrices/enterprise.

Multi-cluster Warehouses | Snowflake Documentation. docs.snowflake.com/en/user-guide/warehouses-multicluster?gad_source=1&gclid=CjwKCAjwx4O4BhAnEiwA42SbVEv7zakPNLgQ0ezwXEI77YGH_iZUozZIRzyaeQQ7rZIeE4Zyd510kRoCoN0QAvD_BwE.

SafeBreach. "SafeBreach | Breach and Attack Simulation Platform." *SafeBreach*, 17 Oct. 2024, www.safebreach.com

"What Is Security Automation? Types and Best Practices." *SentinelOne*, 16 Oct. 2024, www.sentinelone.com/cybersecurity-101/services/what-is-security-automation/#:~:text=Organizat

ions%20need%20security%20automation%20to%20enable%20faster%20incident%20detection%20and,compliance%2C%20and%20improve%20security%20ROI

Vankrunkelsven, Ann. “Functional vs Non-Functional Requirements. Everything You Need to Know.” *Matrix Requirements*, Matrix Requirements GmbH, 6 Aug. 2024, matrixreq.com/blog/functional-vs-non-functional-requirements-everything-you-need-to-know.

Codecademy. “MVC: Model, View, Controller.” *Codecademy*, www.codecademy.com/article/mvc

Presentations - Software Engineering. software-engineering-book.com/slides.