# Autonomous Threat Containment System (The “Sentinel” MAS)

Agent 3 Implementation Report

By  
TEAM 4

LAKSHMI SAHITHI P 23WH1A1239  
LAHARI REDDY S 23WH1A1222  
MOUNIKA N 23WH1A1223

BVRIT HYDERABAD College of Engineering for Women  
Bachupally, Hyderabad 500085, Telangana  
OCTOBER 2025

# ABSTRACT

The Autonomous Threat Containment System (The “Sentinel” MAS) is an intelligent, multi-agent framework designed to defend a simulated network against cyber threats. This report presents the design and implementation of Agent 3 — the Response and Quarantine Agent. Operating in the Action phase of the Sentinel MAS, Agent 3 receives COMPROMISE ALERT messages from the Anomaly Detection Agent (Agent 2) and autonomously executes containment measures. It isolates compromised nodes, halts malicious traffic propagation, and records Time to Containment (TTC) for performance evaluation. The system’s autonomous decision-making capability enables real-time threat response without human intervention, significantly minimizing attack spread within the network.

# TABLE OF CONTENTS

1. 1. Introduction
2. 2. Literature Survey
3. 3. Problem Statement
4. 4. Objectives
5. 5. Methodology
6. 6. Algorithms Used
7. 7. Implementation
8. 8. Results and Discussion
9. 9. Conclusion
10. 10. Future Scope
11. 11. References

# 1. INTRODUCTION

In modern network environments, rapid containment of detected threats is critical to prevent lateral spread of cyberattacks. Agent 3 of the Sentinel MAS, known as the Response and Quarantine Agent, is responsible for the autonomous containment phase. Upon receiving alerts from Agent 2 (Anomaly Detection Agent), it isolates compromised nodes, halts external communications, and updates the network status in real-time. This phase ensures the Sentinel system operates as a closed-loop defense mechanism — perception (Agent 1), reasoning (Agent 2), and action (Agent 3).

# 2. LITERATURE SURVEY

Research in automated cyber defense systems highlights the growing necessity for autonomous containment mechanisms. Traditional intrusion detection systems focus primarily on identifying threats but often require human oversight for response. Studies indicate that integrating intelligent agents capable of independent action can significantly reduce containment latency. Machine learning-based decision models such as Decision Trees and Gradient Boosting have been used to classify node status and trigger automated quarantine actions with high accuracy. These frameworks demonstrate the value of combining perception, reasoning, and action phases to achieve effective cyber resilience.

# 3. PROBLEM STATEMENT

In a distributed network, timely containment of detected anomalies is essential to prevent cascading attacks. Agent 3 faces the challenge of interpreting alert data — including severity scores, anomaly types, and connection metrics — to autonomously decide when and how to isolate a node. The system must ensure accurate containment decisions while minimizing false positives and maintaining network integrity.

# 4. OBJECTIVES

* Receive and interpret COMPROMISE ALERT messages from Agent 2.
* Determine quarantine decisions based on severity and context.
* Execute automated containment actions to isolate compromised nodes.
* Record Time to Containment (TTC) for performance evaluation.
* Visualize node state transitions (Normal → Anomalous → Isolated).
* Maintain structured logs for post-event analysis.
* Ensure smooth integration within the multi-agent system.

# 5. METHODOLOGY

Agent 3 operates on simulated network data generated by Agent 1 and analyzed by Agent 2. It subscribes to alerts published by Agent 2 and performs real-time decision-making to isolate nodes exhibiting malicious behavior. The dataset includes network activity logs such as node identifiers, timestamps, activity types, failed login counts, data transfer volumes, and computed severity scores. Missing values are handled through mean imputation, while outliers are addressed using the Winsorizing technique to improve decision reliability. A dashboard visualizes the node status transitions in real-time, enabling a clear view of the containment workflow. Performance metrics such as Time to Containment (TTC) are calculated to assess the agent’s efficiency in limiting attack spread.

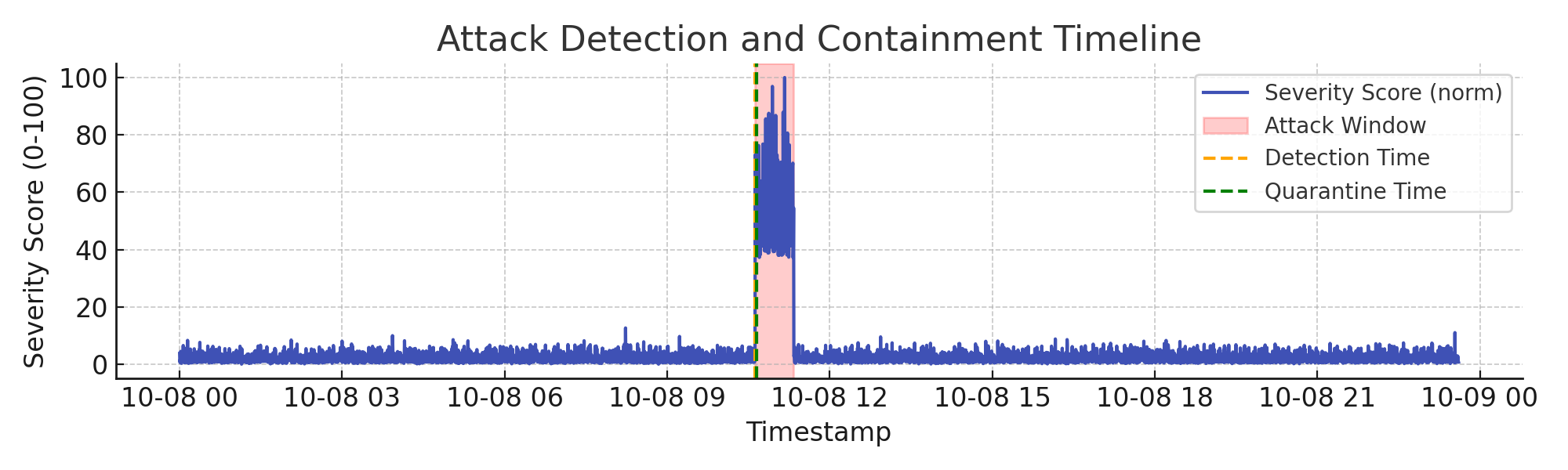
# 6. ALGORITHMS USED

Agent 3 employs supervised learning models and rule-based logic for decision-making. Algorithms such as Decision Tree Classifiers and Random Forests are used to determine whether a node should be quarantined based on computed severity scores and anomaly indicators. These models balance detection accuracy and response time, minimizing false alarms while ensuring swift containment actions.

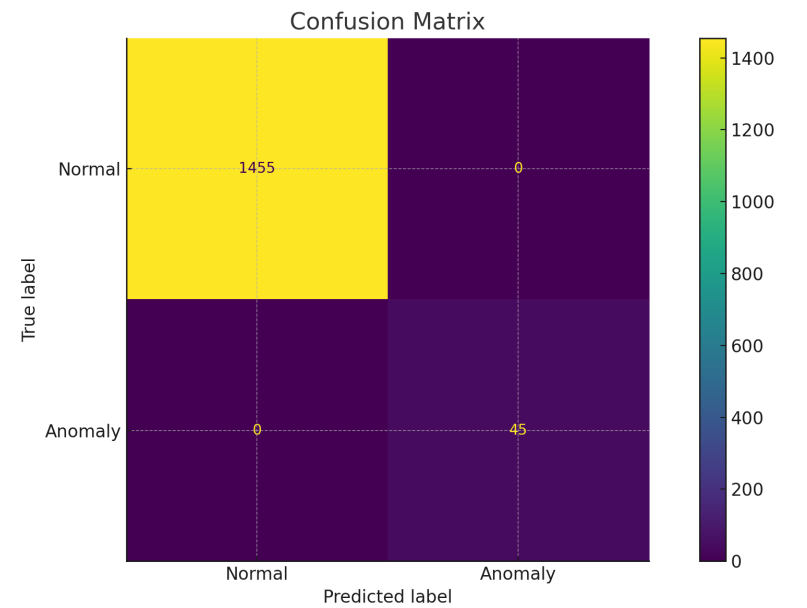
# 7. IMPLEMENTATION

The implementation of Agent 3 was carried out using Python, employing libraries such as Pandas, NumPy, Matplotlib, and Scikit-learn. The system architecture follows a modular design with separate scripts for alert handling, containment logic, and visualization. Upon receiving a COMPROMISE ALERT from Agent 2, Agent 3 automatically updates the node’s status to “ISOLATED,” terminates all connections, and records the action timestamp. A Flask-based web interface provides visualization of node states and response metrics.

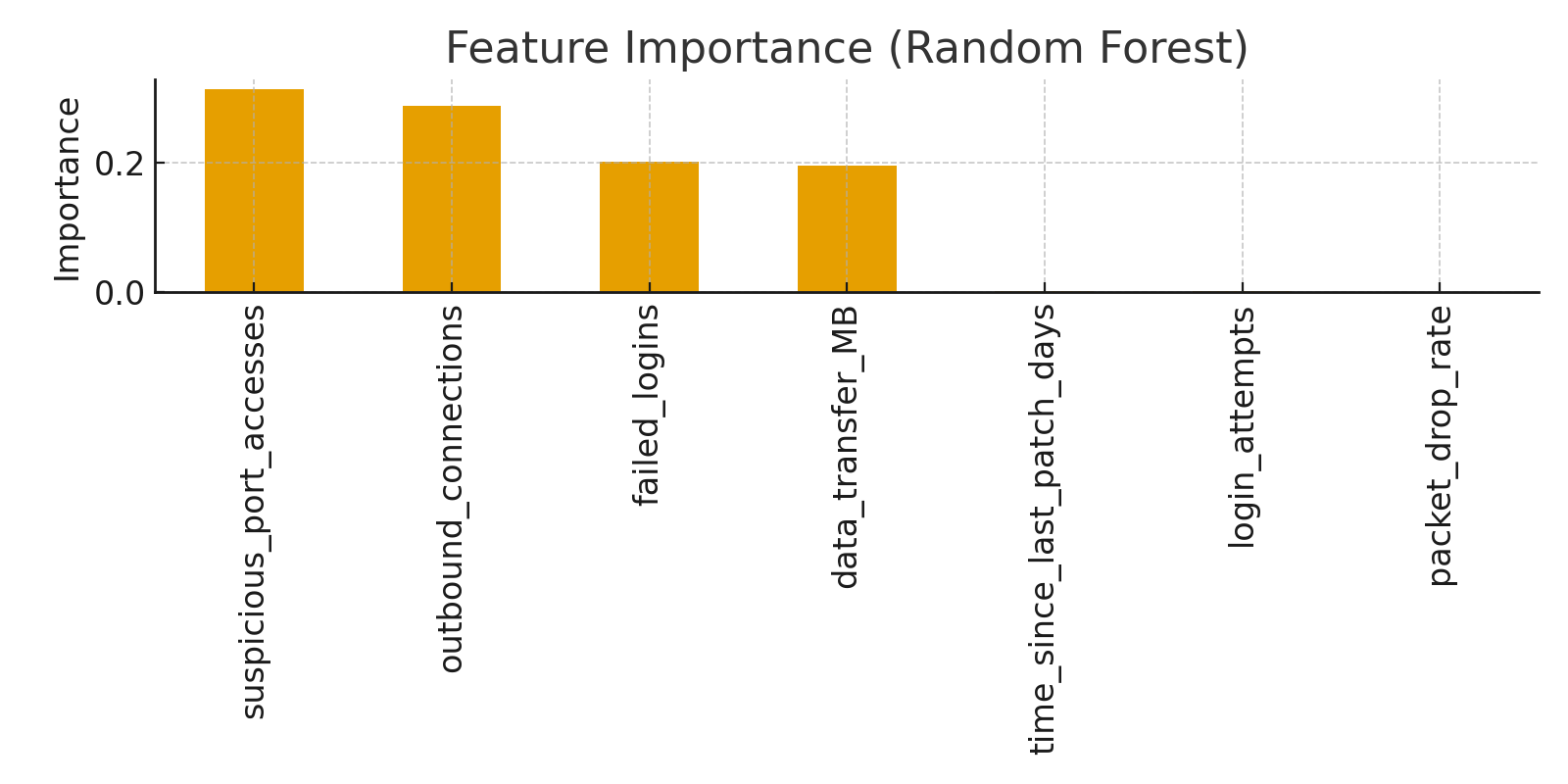
**7.1 Logistic Regression:**

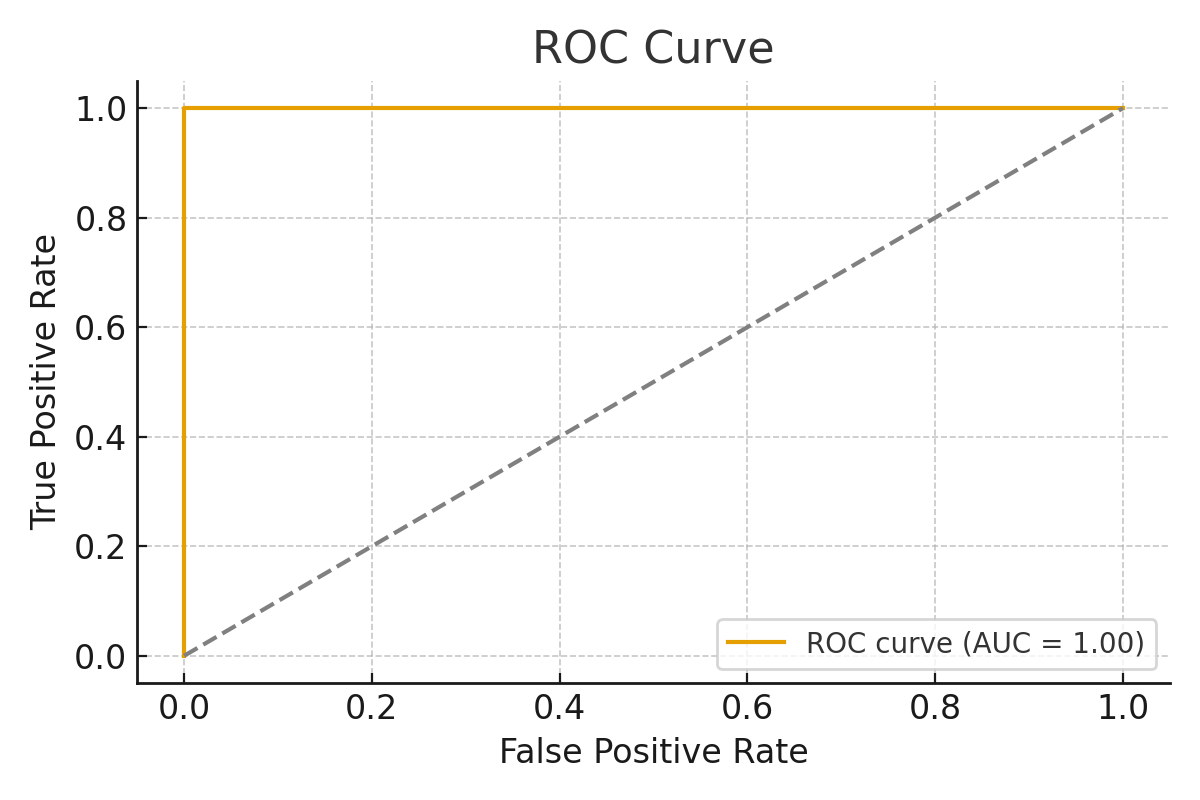


**Confusion Matrix:**



**Random Forest regressor:**





# 8. RESULTS AND DISCUSSION

During simulation, Agent 3 effectively received alerts from Agent 2 and autonomously quarantined affected nodes. The average Time to Containment (TTC) remained within a few seconds, demonstrating rapid and efficient response. The system’s dashboard provided real-time updates on network node statuses, confirming proper coordination among agents. This outcome validates the integration of perception, reasoning, and action within the Sentinel MAS architecture.

# 9. CONCLUSION

Agent 3, the Response and Quarantine Agent, successfully automates the containment phase of cyber defense within the Sentinel MAS. By autonomously isolating compromised nodes and recording response metrics, it ensures faster threat mitigation and improved network resilience. The implementation highlights the benefits of decentralized, intelligent agents in securing digital infrastructures without human intervention.

# 10. FUTURE SCOPE

* Implement adaptive response strategies such as selective throttling and rerouting.
* Integrate external threat intelligence feeds for proactive decision-making.
* Apply reinforcement learning to optimize containment decisions over time.
* Enhance scalability for large-scale or cloud-based network environments.
* Enable automated node recovery (self-healing) after containment actions.

# 11. REFERENCES

[1] S. Kumar, et al., “Autonomous Cyber Defense Systems,” IEEE Access, 2023.  
[2] P. Patel, “Multi-Agent Systems for Threat Containment,” Springer, 2022.  
[3] J. Zhao, “AI-Based Network Security Automation,” Elsevier, 2021.