Autonomous Access Control System with Hierarchical Intrusion Detection*

Yunji Kim
Artificial Intelligence
Dongguk University
Seoul, Korea
2019112107@dgu.ac.kr

Jihyeon Kim
Computer Science and Engineering
Dongguk University
Seoul, Korea
jiwlgus048@dongguk.edu

Dongho Kim
Software Education Institute
Dongguk University
Seoul, Korea
dongho.kim@dgu.edu

ABSTRACT

We present a new machine learning-based hierarchical approach for autonomous access control in computer systems, aiming to overcome traditional static methods' limitations. It builds on our previous work with hierarchical classification and aims to create an autonomous access control system for real-time security enhancement.

KEYWORDS

Intrusion Detection System, Machine Learning, Hierarchical Classification, Access Control

1 Introduction

1.1 Research Background and Objectives

With the increasing complexity of computer systems, network intrusion detection and access control policy management have become challenging. Recent research introduced machine learning techniques to address these issues[1]. Our previous study[2] observed significant performance enhancements in the intrusion detection machine learning model utilizing a hierarchical architecture. We design a system to detect intrusions in real time and automatically generate and apply access control policies based on the detection results without human intervention.

1.2 Previous Work

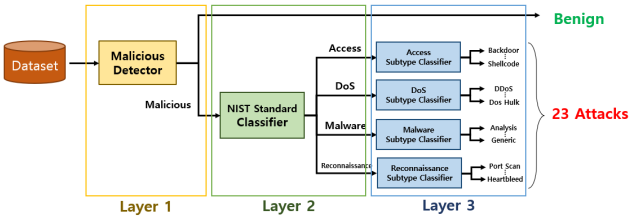

Figure 1: **Hierarchical Intrusion Detection Model**

As shown in Figure 1, [2] is hierarchically structured, first classifying traffic into malicious and benign. Malicious data is further classified into four major categories and divided into specific intrusion types. This hierarchical structure not only improves classification performance but also holds promise for application in real-time systems and provides effective access control.

Classifying high-volume data into specific intrusion scenarios can be time-consuming due to the complexity of the model. Therefore, when implementing an intrusion detection model, it is crucial to swiftly identify malicious traffic. [2] is designed to effectively address this situation. We performed feature selection to utilize the model with fewer features, simplifying it and reducing complexity to enable faster analysis.

We developed a model capable of detecting 23 distinct intrusion scenarios using two benchmark datasets. These scenarios were grouped into four categories based on the NIST standard. Each category comprises similar intrusion scenarios, which share access control policies.

2 Intrusion Detection and Automatic Response

2.1 Real-time Intrusion Detection

Network traffic data is extensive and demands real-time processing. In particular, cyberattacks can lead to disruptions by producing a significant volume of traffic. This introduces spatiotemporal challenges in systems requiring event-based architectures and asynchronous processing. We propose a solution to these issues using Apache Kafka and Spark[3], introducing a system capable of managing large-scale traffic. This platform supports machine learning libraries and is efficiently designed to implement [2]'s structure, where multiple models are integrated. The framework is shown in Figure 2.

The framework follows these steps:

1. Captures packets in real-time through pfSense[4], an open-source firewall and router platform.
2. Load data into Kafka and preprocess data for [2].
3. [2]'s malicious detector determines whether network traffic is malicious, and the output is transmitted to pfSense, which blocks intrusions.

4. For traffic determined to be malicious in step 4, [2]'s intrusion type classifier outputs a refined intrusion scenario. This result is relayed to pfSense, which implements specific access control policies.

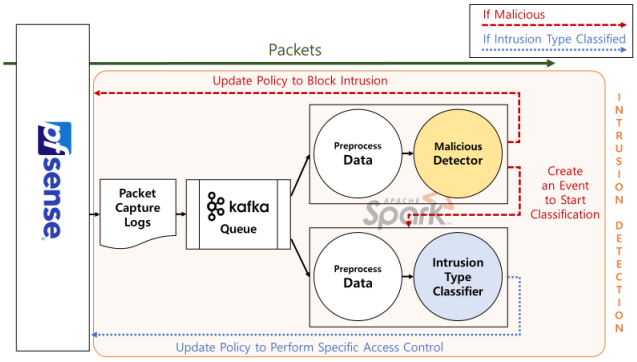

Figure 2: **Framework for Real-time Intrusion Detection**

2.2 Comprehensive Access Control

The access control policies are categorized into host-based and network-based. Host-based access control functions at the individual host system level to manage security and operate at the application layer. Network-based access control analyzes network traffic to control access and works at the network layer and transport layer. Table 1 shows each access control method by NIST intrusion type.

Table 1: **Access Control Method by NIST Intrusion Type**

NIST Type	Host-based Access Control	Network-based
Reconnaissance	Apply application-level input validation to filter out malicious code and disable unnecessary services and ports on the system.	Detect and block incoming traffic from untrusted sources or containing malicious code.
Access	Verify the digital signatures of executable files and control execution permissions. Block specific IP addresses or user agents associated with known malicious bots.	Block unauthorized access attempts and monitor and block requests that resemble those from bots.
DoS	Filter and block malicious traffic, especially HTTP requests exceeding specific thresholds, through continuous monitoring and even network traffic distribution.	Set IP connection limits for web servers and implement rate-limiting rules to control request rates per IP address.
Malware	Limit the permissions of regular user accounts without administrator privileges.	Use an Intrusion Detection System to detect and block common intrusions.

We establish access control policies by leveraging pfSense's various access control capabilities. pfSense supports both host- and network-based access control while offering a wide range of security functionalities[5].

For host-based access control, the Snort package for pfSense enables application-centric detection by enabling the OpenAppID module, providing access control at the application level[5]. We utilize the OpenAppID module to detect intrusions at the application level and establish access control policies. For network-based access control, we establish intrusion category-based access control policies using pfSense's access rule system and convert this information into command scripts to build an automated system. The framework is shown in Figure 3.

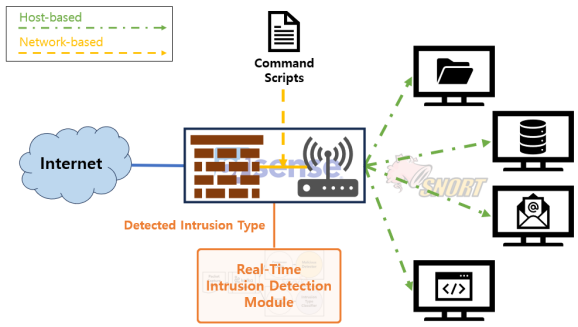

Figure 3: **Framework for Autonomous Access Control**

3 Conclusion

The proposed system allows for real-time intrusion detection and automatic response with access control policies across various intrusion scenarios, without human intervention. This enhancement contributes to bolstering security by increasing the efficiency and stability of the security system with swift responses to intrusion attempts. Furthermore, it can prevent security issues caused by human error, further strengthening system security.

ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (No. S-2021-A0496-00167).
This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2023-2020-0-01789), and the Artificial Intelligence Convergence Innovation Human Resources Development (IITP-2023-RS-2023-00254592) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation).

REFERENCES

[1] Ilhan Firat Kilincer, Fatih Ertam and Abdulkadir Sengur. 2021. Machine learning methods for cyber security intrusion detection: Datasets and comparative study. *Computer Networks 188, 107840.* DOI:https://doi.org/10.1016/j.comnet.2021.107840.
[2] Kim, Y.; Kim, J.; Kim, J.; Kim, D. Hierarchical Intrusion Classification using Machine Learning on a Consolidated Benchmark Dataset., Submitted to Future Internet,

https://github.com/CSID-DGU/MLAC/blob/main/docs/HierarchicalIntrusionClassificationusingMachineLearningonaConsolidatedBenchmarkDataset.pdf
[3] Apache Spark. Apache Spark Documentation Retrieved from https://spark.apache.org
[4] pfSense. pfSense-World's Most Trusted Open Source Firewall. Retrieved from https://www.pfSense.org