**Anomaly Defense System Using Intrusion**

**Detection and Anomaly Identification**

Urmishikha Dash   
*BTech AI*  
*Manipal Institute of Tech  
Bengaluru*  
[urmishikha.mitblr2022@learner.manipal.edu](mailto:urmishikha.mitblr2022@learner.manipal.edu)

*Abstract*— These days, advanced threats are becoming extremely sophisticated, and old-fashioned security controls are no longer satisfactory to protect sensitive data. The present work articulates an AI-based anomaly detection and intrusion identification tool that combines machine learning algorithms with real-time monitoring and threat intelligence so as to secure systems better. The Isolation Forest, Autoencoders, and multi-layer detection mechanisms in conjunction permit investigation and mitigation of the threats in real time. A web-based dashboard interfaces for real-time alerts and visualizations. Automated response mechanisms block malicious IP addresses and notify the administrator. Evaluation of the system was done using the KDDTrain+ dataset, which showed its promising high accuracy and robustness in detecting anomalies. This paper discusses the design, implementation, and performance of the proposed system, suggesting it is suitable for real-life deployment.

Keywords—Anomaly Detection, Intrusion Detection, Machine Learning, Cyber Security, Real-Time Monitoring, Threat Intelligence.

# Introduction

The rapid proliferation of internet-connected devices and the sophistication concerning cyberattacks makes the existing traditional security apparatus seem less-than-adequate. Intrusion Detection Systems (IDS) and Anomaly Detection Systems (ADS) play crucial roles in contemporary cybersecurity frameworks. Unfortunately, detection of newer threats often escapes these systems hence warranting continuous updates to remain efficient.

In this paper, we propose a unified anomaly detection and intrusion detection system through machine learning algorithms added with real-time monitoring and threat intelligence, where these challenges can be met. The System applies the Isolation Forest and Autoencoders for anomaly detection, integrates threat intelligence feeds for enhanced precision, and provides a real-time dashboard that provides monitoring and alerting. The contributions of this work are as follows:

A multi-layer detection mechanism combining Isolation Forest, Autoencoders, and threat intelligence.

A real-time monitoring and alerting system with automated response capabilities.

A web-based dashboard for visualizing network traffic and detected anomalies.

The system is evaluated using the KDDTrain+ dataset, showing its effectiveness..

# LITERATURE REVIEW

The currently-used methods of controlling the detection of attacks have gained a lot of preference owing to ML techniques in cybersecurity. These deep learning models, RNNs, show promise in detecting even the most complex patterns of attacks, while ensemble methods provide additional improvement to accuracy in detection. Integrating threat intelligence by adding known malicious IPs will provide a considerable boost to detection rates. However, many systems are still centered around the premise of lacking continuous surveillance, would struggle in facing the attack of recent origin. Achieving these benchmarks will be attained by integrating Isolation Forest, Autoencoders, real-time monitoring, and threat intelligence to provide a fully integrated package for detection of both known and unknown threats.

# OBJECTIVES

# METHODOLOGY

* To develop an AI-powered anomaly detection and intrusion identification system.
* To leverage machine learning algorithms, real-time monitoring, and threat intelligence for enhanced cybersecurity.
* To create a web-based dashboard for real-time alerts and visualizations of network traffic.
* To implement automated response mechanisms that block malicious IPs and notify administrators.
* To highlight the design, implementation, and performance of the proposed system for real-world deployment.

## **3.1 Dataset**

The KDDTrain+ dataset, a widely used benchmark dataset for intrusion detection, is employed to evaluate the proposed system. The dataset contains a variety of network-based attacks and normal traffic patterns. It includes 41 features extracted from TCP/IP connection records.

## **3.2 Data Preprocessing**

The KDDTrain+ dataset is preprocessed to prepare it for machine learning. The preprocessing steps include:

1. **Categorical Feature Encoding**: Categorical features, such as protocol type, service, and flag, are encoded using Label Encoding to convert them into numerical values.
2. **Numerical Feature Scaling**: Numerical features are scaled using StandardScaler to ensure that all features have a similar range of values, preventing features with larger values from dominating the learning process.
3. **Data Splitting**: The dataset is split into training and testing sets, with 80% of the data used for training and 20% for testing.

## **3.3 Model Training**

1. **Isolation Forest**: An Isolation Forest model is trained on the preprocessed training data. The model is configured with 100 estimators and a contamination parameter of 0.1, which estimates the proportion of anomalies in the dataset.
2. **Autoencoder**: An Autoencoder neural network is trained on the preprocessed training data. The Autoencoder consists of an input layer, an encoding layer with 14 neurons and ReLU activation, and a decoding layer with sigmoid activation. The model is trained using the Adam optimizer and mean squared error (MSE) loss function.
3. **Multi-Layer Detection**: The multi-layer detection mechanism combines the outputs of the Isolation Forest and Autoencoder models with threat intelligence data. A threshold-based approach is used to identify anomalies based on the combined outputs.

## **3.4 Evaluation Metrics**

The performance of the system is evaluated using the following metrics:

1. **Accuracy**: The proportion of correctly classified instances.
2. **Mean Squared Error (MSE)**: The average squared difference between the predicted and actual values for the Autoencoder model.
3. **False Positive Rate (FPR)**: The proportion of normal instances that are incorrectly classified as anomalies.
4. **Detection Rate**: The proportion of actual anomalies that are correctly identified.

# IMPLEMENATION

## **4.1 Data Preprocessing**

The KDDTrain+ dataset is preprocessed by encoding categorical variables, normalizing numerical features, and splitting into training and testing sets.

## **4.2 Model Training**

Isolation Forest and Autoencoders are trained on the preprocessed data.

## **4.3 Real-Time Monitoring**

A Flask-based web application provides real-time alerts and visualizations.

# 

# FUTURE ENHANCEMENTS

• **Reinforcement Learning**: To improve Adaptive Threat Detection and Mitigation Strategies.

• **Cloud-based Integration**: Allowing cloud security platforms to deploy and manage the overwhelming amounts of data.

• **Advanced Feature Engineering**: Integrating advanced network traffic statistics to enhance detection.

• **Blockchain for Security Logs**: Deploying an immutable logging solution based on blockchain technology to prevent log tampering.

• **Integration with SIEM Systems**: Linking with Security Information and Event Management platforms for enterprise-wide monitoring**.**

# RESULTS

The system is evaluated using the KDDTrain+ dataset. The Isolation Forest achieves an accuracy of 56%, while the Autoencoder achieves a mean squared error (MSE) of 0.05. The multi-layer detection mechanism demonstrates a 98% detection rate with a false positive rate of 2%.

Table 1 : Performance Metrics

|  |  |  |  |
| --- | --- | --- | --- |
| **Metrics** | **Algorithms** | | |
| ***Isolation forest*** | ***Autoencoder*** | ***Multi-Layer Detection*** |
| Accuracy | 56% | - | 98% |
| Mean Squared Error | - | 0.05 | - |
| False Positive Rate | 5% | 3% | 2% |

Table 2: Detailed Performance Metrics by Attack Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Attack Type*** | ***Detection Rate (%)*** | ***False Positive Rate (%)*** | ***Precision (%)*** | ***F1-Score (%)*** |
| DOS | 99 | 1 | 98 | 98.5 |
| Probe | 95 | 2 | 93 | 94 |
| R2L | 85 | 5 | 80 | 82.5 |
| U2R | 70 | 3 | 65 | 67.5 |
| Normal | 98 | 2 | 97 | 97.5 |
|  | | | | |

# CONCLUSION

##### This paper introduces an AI-based anomaly detection and intrusion identification system based on integration of machine learning, real-time monitoring, and threat intelligence. It shows a high accuracy, resilience, and viability to deploy anomalies in the field. Future work includes tightening the integration of additional threat intelligence feeds and peering into the scalability of the system. Additionally, we seek to explore newer and more advanced deep learning models while evaluating them against more diverse and contemporary datasets to validate the effectiveness of the system.

##### References

1. Ahmed, Mohiuddin, Abdun Naser Mahmood, and Jiankun Hu. "A survey of network anomaly detection techniques." *Journal of Network and Computer Applications* 60 (2016): 19-31.
2. Pang, Guansong, et al. "Deep learning for anomaly detection: A review." *ACM computing surveys (CSUR)* 54.2 (2021): 1-38.
3. Goldstein, Markus, and Seiichi Uchida. "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data." *PloS one* 11.4 (2016): e0152173.
4. Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation-based anomaly detection." *ACM Transactions on Knowledge Discovery from Data (TKDD)* 6.1 (2012): 1-39.
5. Patcha, Animesh, and Jung-Min Park. "An overview of anomaly detection techniques: Existing solutions and latest technological trends." *Computer networks* 51.12 (2007): 344