National College of Ireland

ML-BASED ZERO-DAY ATTACK DETECTION

MSc Research Project
MSc Cybersecurity

Ans Maria Mathew
Student ID: 23173661

School of Computing
National College of Ireland

Supervisor: Jawad Salahuddin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Ans Maria Mathew
Student ID:	23173661
Programme:	MSc Cybersecurity **Year:** 2024
Module:	MSc Research Practicum Part 2
Supervisor:	Jawad Salahuddin
Submission Due Date:	12th December 2024
Project Title:	ML-Based Zero-Day Attack Detection
Word Count:	6959 **Page Count**: 20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Ans Maria Mathew

Date: 12th December 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

ML-Based Zero Day Attack Detection

Ans Maria Mathew
23173661

Abstract

In this paper discussing a new machine learning based approach and architecture for intrusion detection system (IDS) for detecting zero-day attack. By employing IoT23 dataset, the research employs supervised learning with Random Forest; unsupervised learning through Isolation Forest; deep Neural Network (DNN) for the high criticality data. Expanding the dataset by Smote as well as feature scaling helped in achieving a good performance by the models. Considering the achieved outcomes, having high values of accuracy level, as well as pioneers' high level of precision and recall, it is worth to concentrate on DNN as the most effective and accurate variant with over 99% of accuracy. Validity issues such as privacy and fairness were considered in the study. This multi-layering makes it quite effective to hold up as a model for practical applications in the field of cybersecurity.

Keywords: Zero-day attack, Intrusion Detection, Machine Learning, Anomaly Detection, Supervised learning, Unsupervised learning, Neural Network

1 Introduction

With attackers targeting unknown vulnerabilities, and such threats being not known to the software vendor or the concerned security community, zero-day attacks are turning out to be one of the most serious cyber threats today. Signature-based detection is ineffective against such unknown threats in a scenario whereby every method of detection used so far relies on this very principle. One of the promising solutions emerging for zero-day attack detection is machine learning. This can be learned from existing data thus adjusting to newer threats, and it recognizes patterns that seem familiar and have been exposed by unknown vulnerabilities before. The paper discusses an investigation in which embedding techniques of machine learning into Intrusion Detection and Prevention Systems may enhance the effectiveness of real-time detection and mitigation of zero-day attacks.

Research questions

In which way can a machine learning model be properly utilized in the context of Intrusion Detection/Prevention Systems (IDPS) in order to detect zero-day attacks and afterward deploy it?

Background study

Machine learning (ML), the detection of zero-day attacks are the active areas of research in cybersecurity. This trend is driven by the frequency and intensity of the cyberattacks. Essentially, a zero-day attack takes advantage of a software vulnerability unknown to the

vendor or the security community. The systems will remain open to vulnerability until such a time as a patch may be fashioned for subsequent deployment. Traditional detection methods are signatures-based methods that cannot detect these unknown attacks since these methods work on already documented patterns (Guo, 2023). That is to say, more adaptive and proactive approaches in the detection of zero-day attacks refer to how models learn from historical data, determine system behavior trends, and then go ahead to report any deviation as a potential new unknown attack. It applies approaches of supervised learning, unsupervised learning, and anomaly detection. The strengths of zero-day detection of these approaches include the detection of known and novel threats (Sarhan *et al*. 2023). The problem is that, by definition, zero-day attacks are largely unknown and are therefore not sufficiently represented by labeled data, which heavily makes feature selection and model training with such scarce resources particularly tricky. In addition, adverse manipulation can be used to cheat an ML model into misidentifying attacks.

Structure of the research paper

Section 1: Abstract and Research Question
Section 2: Literature Review: It offers a thorough examination of the advantages and disadvantages of each ML technique to zero-day detection, as well as a critical review of the various approaches.
Section 3: This section outlines the approach, including model training and data selection. It includes information on research ethics and, at the end, gives result and future recommendations.

2 Related Work

Overview of Zero-Day Attacks in the Modern Paradigm of Cyber Security. This section discusses an overview of zero-day attacks in the modern paradigm of cybersecurity. Zero-day vulnerabilities pose significant threats because they have not yet come to the notice of the software vendor or security community and, consequently, systems are exposed to the attack until a patch is developed. Because such attacks are complex and evolving, they cannot be successfully countered by traditional signature-based methods. ML-based detection approaches have proved to be promising candidates that overcome the above challenges. They can detect never-before-seen attacks and learn from patterns in data.

2.1 Advantages of ML over traditional detection methods

Traditional methods of detection include signature-based and heuristic approaches, which have dominated cyber security for many years. These approaches rely upon predefined patterns or known characteristics of attacks for identifying threats. Signature-based detection checks incoming data against a database of known attack signatures, whereas heuristic detection tries to identify suspicious activity based on predefined rules or behavior. However, these methods are always at an advantage with novel, zero-day attacks that exploit previously unknown vulnerabilities (Ahmad *et al*. 2023). Zero-day attacks are the hardest to detect because there aren't signatures and patterns in place to match against them. Thus, they keep devouring systems until a patch is developed or an update is published. Machine learning

(ML) models represent the dynamic and adaptive approach of cybersecurity that learns from historical attack data. They address the detection of emergent attack vectors through data-driven analysis, unlike the traditional method of prebuilt signatures.

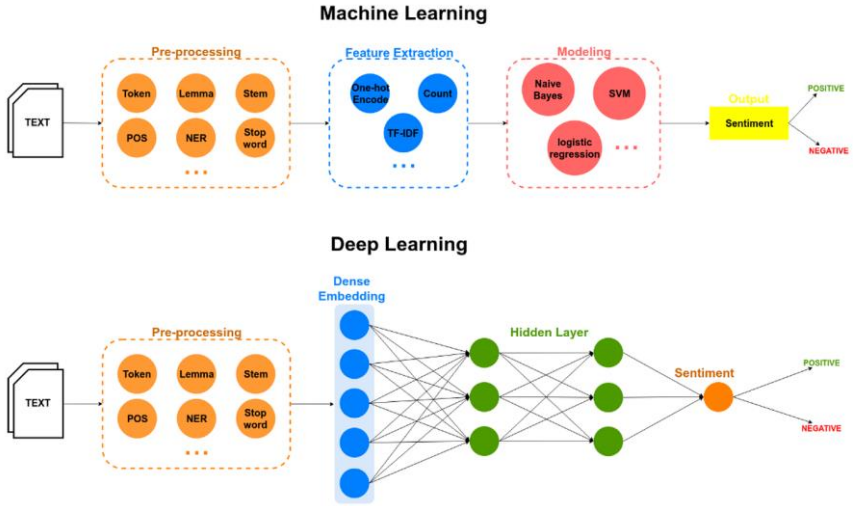

Figure 1 The Advantages of Deep Learning

(Source: Ali *et al*. 2022)

In return, the supervised learning methods identify known threats as well as new variants, while the unsupervised learning methods identify anomalies within the network or system behavior without any labeled data (Ali *et al*. 2022). Deep learning models, mainly those based on neural networks, could process large volumes of complex data and identify subtle patterns of attacks, which is very effective in discovering sophisticated zero-day threats. ML-based detection tends to become better with time as newer data becomes available and processed, enhancing the detection capabilities of systems regarding new attack vectors and further refining the algorithms for detection over time (Mbona and Eloff, 2022). The ML-based systems are extremely scalable and capable of coping with ever-growing amounts of data, making them fit for modern cybersecurity environments.

2.2 Importance of Proactive Detection Methods

The ability to detect possible threats before they materialize into real attacks is the primary distinguishing characteristic of proactive detection methods compared to traditional reactive systems. Whereas the reactive systems react after having experienced an incidence of known threats, proactive detection focuses on predicting or trying to pre-empt potential vulnerabilities before exploitation (Igugu, 2024). This is very important in the context of zero-day attacks as the exploit exploits an unknown vulnerability to a software vendor and the public at large. Zero-day vulnerabilities are quite alarming because they are usually discovered and exploited before patches or fixes are made available, which means that the systems remain open to exploitation (Mohamed, Taherdoost and Madanchian, 2024). Therefore, machine learning becomes an essential tool in the identification of zero-day attacks because it helps in the management of large data that provides insights. For example, ML would identify anomalous patterns in system behavior so that the threat could be detected in real-time before its attack escalates to a breach.

The ML-based system, therefore, learns continuously from incoming data and recognizes new attack vectors based on emerging trends (Wang *et al*. 2020). Proactive detection reduces potential damage from zero-day attacks through preventive action, for example, isolating compromised systems or blocking malicious activities, thus reducing breach recovery costs and long-term business operations impact. In the sophisticated era of today's data breaches and cyberattacks, proactive detection techniques based on ML are now the keys to further strengthening defenses regarding cybersecurity and system integrity.

2.3 Challenges in feature selection for zero-day detection

Feature selection is an important task related to the performance of the ML model, mainly about zero-day attack detection. The quality and relevance of the features applied in the training will have a direct impact on how likely an ability to detect new and unknown patterns of attacks might be. It becomes even more important in the case of zero-day attacks because such attacks target previously unknown vulnerabilities that, very often, evade traditional detection mechanisms (Soltani *et al*. 2023). Thus, the correct selection of feature sets targeted previously will significantly enhance the precision and robustness of ML detection against such attacks. Detection of zero-day attacks is challenging when labeled data do not exist for unknown attacks. It is hard to train supervised models over the requirement of labeled data. Thus, researchers are opting for unsupervised or semi-supervised learning techniques, sometimes less exact in identifying attacks without clear labels. Moreover, the dynamic nature of network traffic and system behavior complicates feature selection. Zero-day attacks can vary widely and have changed network behaviors or system logs based on different types of attacks (Gowthami and Priscila, 2024). High-dimensional network data can suffer from the "curse of dimensionality," wherein too many features decrease the model's performance due to noise. Domain-specific expertise is also necessary for feature selection because it recognizes which are the most relevant features. This domain knowledge filters out the irrelevant or redundant features and enhances the performance in detection provided by the model.

2.4 Vulnerabilities of ML models to adversarial manipulation

Machine learning has proved to be very promising for enhancing cyber security capabilities in detecting zero-day attacks, referring to newly unknown vulnerabilities within software. Still, despite the enormous potential of ML-based detection systems, there is no exemption to tampering. The reason is that the system is prone to attacks in the adversarial type. This is an attack where input data has been subtly modified in a way to deceive the ML model into making incorrect predictions or classifications (Zahoora *et al*. 2024). In the context of zero-day detection, adversarial attacks can create input data that are perceived by the system as innocuous, but, in the real scenario, is an imitation of the characteristics of malicious activities. These crafted inputs are called adversarial examples; they could bring about a failure of the ML model to recognize a true attack pattern, thereby leading to missed detections or false negatives (Hairab *et al*. 2022). The ability of adversarial examples to bypass detection drastically diminishes the reliability and performance of ML models in cybersecurity applications.

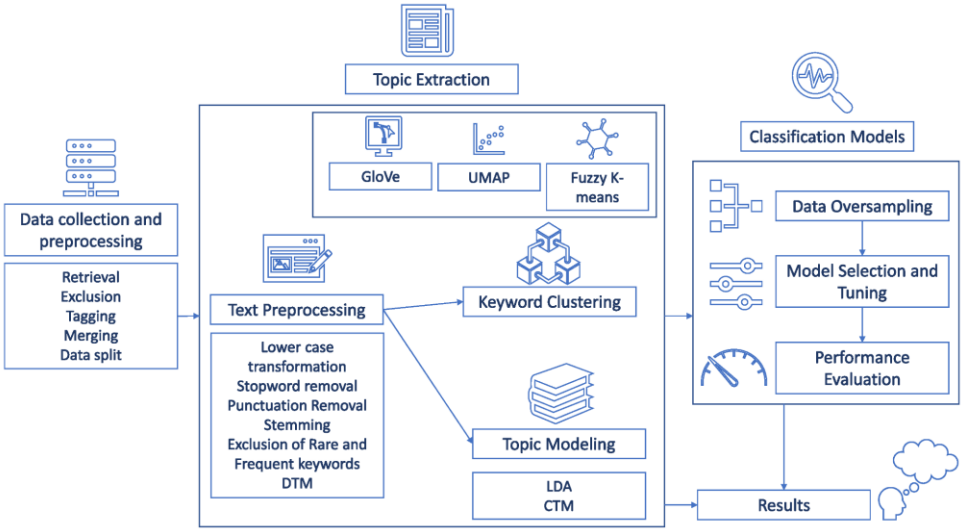

Figure 2: Exploitation of Vulnerabilities

(Source: Touré *et al*. 2024)

In the context of detecting zero-day attacks, the goal here is that of detecting previously unknown vulnerabilities; this challenge is raised by adversarial manipulation. Zero-day attacks are novel, by definition, and hence an adversary could exploit this feature to craft inputs that evade traditional detection systems. This jeopardizes the overall security posture of those systems that rely on ML-based solutions for protection (Touré *et al*. 2024). To address those vulnerabilities, the community now explores a range of defensive techniques. The most widely studied is the adversarial training technique: it exposes the ML model during training to adversarial examples, allowing the model to learn how to deal with manipulated inputs. Robust optimization techniques try to make the model robust by changing some of the parameters of the model to minimize its susceptibility to adversarial manipulations. In addition, input sanitization techniques involve preprocessing input data to eliminate or minimize the effects of adversarial perturbations before being fed to the model.

2.5 Predictions for advancements in real-time detection and autonomous systems

Advances in machine learning (ML) and artificial intelligence (AI) are impacting the dynamics of zero-day attack detection systems to make them much more dynamic based on real-time capabilities and autonomous response mechanisms. Zero-day vulnerabilities represent flaws in software that remain previously unknown and unpatched until cyberattackers seize them to attack a system. Traditional methods of detection will not identify such threats before they are exploited. There is an issue, though: pre-existing solutions (Rizzardi, Sicari and Porisini, 2024). However, the use of ML and AI models, mainly those based on real-time detection, would suffice to reverse this challenge. Real-time detection attempts to process streams of information in real-time and detect anomalies or malicious patterns when they happen. Most traditional detection techniques rely on known attack signatures or patterns.

Zero-day attacks can be detected by real-time systems through traditional machine-learning approaches that recognize unusual behavior or deviations from established baselines (Ngo

and Nguyen, 2022). This would be important in minimizing the potential of a zero-day exploit because it would allow getting a response to the attack as soon as it is happening and not after it has occurred. Autonomous systems are likely to be important components in future implementations of zero-day detection frameworks because they can be designed to isolate or block threats as and when detected without human interaction. Especially, in the case of zero-day attacks where every moment counts, a swift response is needed. Autonomous systems may automatically cut off the offending devices from networks or block specific traffic to halt the propagation of attacks (Guo,). Federated learning is an underlying framework for distributed machine learning wherein the pattern-deciphering models are trained in overdistributed devices without sharing raw data, thereby preserving privacy while contributing together to zero-day attack detection. This approach is scalable and secure for zero-day detection.

2.6 Literature gap

Although large strides have been made in the research domain concerning ML-based detection techniques, there is still much that is left to be done in the literature, especially on zero-day attack detection. Perhaps one of the most significant issues concerning this stage of evolution is the lack of large, annotated datasets that are indeed crucial for feeding ML models designed specifically for zero-day attacks. Zero-day attacks are inherently rare and notorious for evolving very rapidly (Alahmadi *et al.* 2023.). Additionally, documentation of such attacks is limited in many cases, thereby limiting model generalization capabilities for learning unknown attack patterns. In addition, although most of the works emphasize improving the detection efficacy, much less focus is directed toward actual real-world deployment into deployed cybersecurity scenarios. Specifically, scalability issues as well as issues of model interpretability and adoption in existing cybersecurity frameworks are mostly neglected. One of the biggest implications for the adaptation of ML models in handling large quantities of data within various network environments is scalability, while interpretability is critical to enable trust and transparency in automated decision-making. Thirdly, integrating ML-based detection systems into the implemented cybersecurity architecture is also a complex and underexplored domain. These gaps need a multi-dimensional improvement based on better data availability, model transparency, and deployment feasibility.

3 Research Methodology

This chapter presents the method in which the systematic approach is used to identify the zero-day attacks with the help of ML methods. This research intends to establish a basic IDS with Python and several ML algorithms, all in pursuit of predicting hitherto unknown threats. The first section of this chapter provides an outline of the dataset, the features that characterize it, and the importance of the data set in reflecting the specific problem area. It goes with the proper steps of data cleaning, data normalization, and feature extraction which are important steps for modeling. To this end, the methodology also provides an understanding of the supervised, unsupervised, and deep learning algorithms with focus placed on the application and assessment of the algorithms in combination with one another. Ethical issues and limitations of the project are also covered.

3.1 Dataset Overview

In this study, IoT23 Combined is used for the dataset, which is a vast set of network traffic, specially designed for intrusion detection. It also contains examples of good and bad practices that allow for supervised and unsupervised machine learning. The dataset consists of both quantitative as well qualitative features that are size of packets, protocol type, and IP address which offers some insights into network activities (Rakine *et al.* 2024). The frequency of the domestic attacks variety, such as the simulated zero-day ones, indicates its relevance to real-life usage. It has been also observed that data has missing values and the class distributions are imbalanced due to which these problems have been also dealt with in the preprocessing section. The fact that it consisted of a big number of samples and samples' heterogeneity posed challenges in terms of its use in developing the model (Li *et al.* 2023). The structure of the dataset has been analyzed as well as the specific areas that needed modification in the preprocessing step with the help of the class distribution plots visualizing the dataset.

3.2 Data Preprocessing

The process of data preprocessing is key before launching a machine learning model and feeding the machine learning model. Initially, irrelevant columns, such as Unnamed: These are identifiable in LTD-AR and numbered 0, and port-related identifiers have been also excluded from the dataset because it has been thought to add little value in data analysis. Location and missing values have been addressed by replacing them with zeros since datasets should be kept complete and integral during training (Mu, Shi and Dogan, 2024). For classification, the target variable Label has been reduced to two domains 0 for benign and 1 for malicious. When there is no such column in a given dataset, an equivalent target attribute has been determined to be used. Feature scaling has been implemented using StandardScaler to scale the numerical attributes of the data so that all features have the same weightage when the model has been learned. The distribution of the Dataset has been studied using the visualization method to handle the class imbalance to identify that sampling technique or augmentation has been required during model development. Outliers in data distributions have been identified to have a neat arrangement of dataset structure for the delineation of features and models.

3.3 Supervised Learning

The supervised learning has been used whereby network activities have been categorized into either legitimate or illegitimate. Some of the families that have been used as predictive algorithms include Random Forest because they are incredibly effective in dealing with feature high-dimensionality data (Khandelwal and Shreejith, 2023). Random Forest has been selected due to its stability and capacity to determine the degree of features' significance for the detection of fraudulent cases. A random sample made up of 20% of the entire set has been used for testing purposes while the rest which constitutes the remaining 80% has been used to train the model The features have been then normalized so that they would be given equally (Yang *et al.* 2024). All the models have been trained on the labeled dataset and tested for precision, recall, f1-score, and confusion matrix. These supervised models are effective at finding known attack types, making them effective as a foundational layer to zero-day

detection when used in conjunction with other models such as the anomaly, or a combination of both.

3.4 Unsupervised Learning

Independent learning algorithms have been used for discovering outliers to the data set and therefore are potential examples of zero-day attacks. The isolation Forest algorithm has been used because it is used in detecting outliers in large datasets especially when there is no labeled data of interest in the database. On examining the structure of the dataset it could identify various anomalies within network traffic which have been from usual trends (Yang *et al.* 2024). The contamination parameter has been set to 0.1 since it denotes a fraction of anomalies in the data set. They have been then mapped onto binary labels with the anomalies labeled as malicious. The presentation of some anomalies uncovered by various techniques offered an understanding of the efficiency of this approach. This proved most effective in identifying new threats since the pattern matching does not depend on similarities but rather on differences (Sahani *et al.* 2023.). Concentrating on the unsupervised learning, the work supplemented the supervised models strengthening the system's ability to respond to previously unknown attack types.

3.5 Neural Network

To improve the classification of zero-day attacks a deep neural network DNN has been developed. The network architecture used incorporated an input layer based on the number of features in the dataset two hidden layers with sixty-four and thirty-two neurons respectively and of course, one neuron in the output layer since it deals with binary classification. The hidden layers of the neural network have been built using parameters of Rectified Linear Unit (ReLU) activation functions that considered non-linear relationships and at the output layer, a sigmoid activation function has been adopted to give probabilities (Kumar, Sinha and Das, 202). Adam has been used as the optimizer for constructing the network while the binary cross-entropy function is the loss function to perform proper weight update during the training. This model has been trained on the scaled dataset to 50 epochs with a batch size of 10 which achieved high testing accuracy. The DNN's capability of handling big data and learning complex patterns best suited it for detecting complex attack patterns including zero-day in real-world applications.

3.6 Model Training and Testing

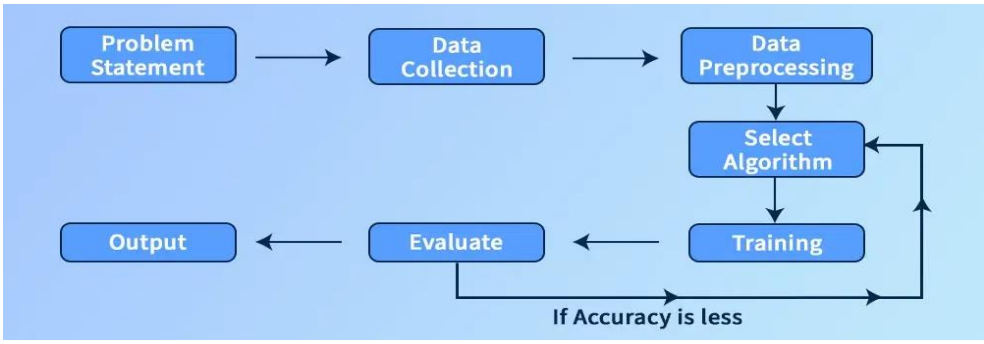

Fig 3: Machine learning Model

These models have been also trained at an 80-20 ratio to the data split that has been used for training and testing. Feature scaling has been used to make all the features have nearly equal variance as intended for the sake of learning algorithms. In this regard, hyperparameter tuning using grid search has been performed on Random Forest model. The neural networks have been trained using backpropagation over 50 epochs with the help of adaptive optimizers which helped the networks to converge at different solutions (ElSayed, Elsayed and Bay, 2024). Isolation Forest has been tuned to perform anomaly detection without many false positives. The model assessment involved parameters of accuracy, precision, recall, F1 score, and the confusion matrix. This training and testing plan has been comprehensive and filled a significant gap in the ability to detect both known and zero-day threats.

3.7 Ethical Consideration

Issues of an ethical nature have been always taken under serious consideration throughout the entire course of the research. For example, complications that include IP addresses have been masked to avoid compromising people's privacy to maximize compliance with legal frameworks like the GDPR. Focused on making the model interpretable to guarantee that the systems' decisions are impartial (Yang *et al.* 2023). Data usage has been confined to analysis for genuine scholarly research, and no utilization such as generating biases or prejudice has been allowed. Thus, following these principles, the present study maintained ethical norms in constructing cybersecurity approaches.

4 Design Specification

The system will use real-time network traffic capturing with emphasis on packet sizes, protocol types, source IP addresses, and length of connections. Balanced datasets will be ensured through appropriate data preprocessing techniques like SMOTE. Machine learning models being used include Random Forest on supervised classification, Isolation Forest for anomaly detection, and a deep neural network for high-dimensional data processing. This is an analytical layer that includes the output such as protocol distributions, feature importance, and patterns in anomaly for better decision making through visualization. A multi-level framework will make sure this comprehensive zero-day threat detection makes actionable insights for subprofessionals.

The design of this packet capture system focuses on providing a structured approach for analyzing the network traffic it emphasize on both general traffic metadata and specific HTTP request details. The system captures packets for a defined number or duration or a certain predefined number of packets whichever is earliest to ensure efficient and controlled data collection. By leveraging pythons Scapy packages capabilities, the system processes various network protocols, extracting critical information such as source and destination addresses, ports, packet sizes, and connection statuses. For HTTP traffic, the design incorporates a specialized parsing mechanism to extract requested paths from GET and POST requests, enabling detailed application-layer analysis for extracting packet information.

The output is organized into a CSV file, making it suitable for review in ai ml analysis tools for further automated processing. The design prioritizes modularity and extensibility,

allowing easy adaptation to additional protocols, extended metadata requirements, or custom traffic analysis needs. The goal is to balance usability with technical depth, providing insights into both high-level network activity and specific application-layer interactions which is customizable based on the iot devices being monitored.

5 Implementation

We assume that the IoT environment consists of uniform devices communicating with a centralized server. All devices use the same protocols and generate data in consistent formats, allowing for predictable patterns in normal behavior. Malicious activities manifest as deviations from this established behavior, either in the form of abnormal packet sizes, request frequencies, or protocol violations.

Data Collection and Preprocessing: Network data, including features such as packet size, source and destination ports, timestamps, and device-specific attributes, is continuously logged. Preprocessing involves cleaning the data by handling missing values and replacing anomalies (e.g., - or NaN) with appropriate defaults or removing them. Data normalization or scaling ensures consistent input to the ML models. Additionally, a baseline dataset representing normal behavior is established, either through prior domain knowledge or observation of the environment over time.

Behavioral Modeling and Training: Using the baseline dataset, unsupervised algorithms such as Isolation Forest or Autoencoders learn the standard communication patterns of the IoT devices. For environments with labeled data, supervised models like Random Forest can be trained to classify data as benign or malicious. During training, the models focus on critical features that capture deviations, such as sudden spikes in packet sizes or unexpected device activity. Incorporate a deep neural network to deal with complex high-dimensional data. Visualize the protocol distributions, feature importance, and outlier patterns to enhance interpretability.

Anomaly Detection and Marking Deviations: Once deployed, the trained models analyze incoming network data in real-time. Any deviation from the learned normal behavior (e.g., high anomaly scores or misclassifications) is flagged as suspicious. Thresholds for anomaly scores are tuned to balance false positives and false negatives. For example, a packet size exceeding the 99th percentile of the baseline or a significant deviation in device communication frequency may trigger an alert.

End Results and Actionable Insights: The system outputs a classification of traffic as normal, anomalous, or explicitly malicious. Anomalous traffic can undergo further inspection to determine its root cause, such as a potential zero-day exploit or unexpected operational changes. Over time, the system adapts by incorporating new data into its models, refining its ability to detect both known and emerging threats. This pipeline ensures robust security while minimizing disruptions in a controlled IoT setup.

In this methodology, first, capture the network traffic using Wireshark, and then extract features, such as packet size, protocol types, and IP addresses. Then apply pre-processing by balancing the class distribution using SMOTE. After data pre-processing for supervised learning, feature importance analysis, classification, and anomaly classification are used based on the Random Forest classifier. Isolation Forest is used for unsupervised learning.

Incorporate a deep neural network to deal with complex high-dimensional data. Visualize the protocol distributions, feature importance, and outlier patterns to enhance interpretability. Combine these models to make a layered intrusion detection system for robust protection.

6 Evaluation

This chapter provides the outcome of the proposed machine learning model for detecting the zero-day attack and the findings related to its consequences for cybersecurity. For the Random Forest that have been supervised model, the correctness of classification has been the measure of Relative accuracy, while the Isolation Forest which is an unsupervised algorithm has been measured in terms of anomaly detection. To learn high-dimensional data and advance attack patterns, a deep neural network (DNN) has been also applied. In addition, further analysis is provided by visualizations such as class distribution, feature importance, anomaly detection, and training curves. These assumptions confirm the efficiency of the proposed models and reveal the potential for further development.

6.1 Supervised Learning

```
print("Random Forest Classification Report:")
print(classification_report(y_test, y_pred_rf))
```

[8] ✓ 0.1s

```
Random Forest Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    208397
          40       1.00      1.00      1.00        33
          44       1.00      1.00      1.00         1
          60       0.00      0.00      0.00         1
          67       1.00      1.00      1.00         7
          73       1.00      1.00      1.00       268
          76       1.00      1.00      1.00       560
          80       0.97      0.97      0.97        35
          82       1.00      1.00      1.00         4
          88       0.75      1.00      0.86         3
         122       1.00      1.00      1.00         4
         131       1.00      1.00      1.00        39
         134       1.00      1.00      1.00         1
         146       1.00      0.50      0.67         2
         152       1.00      1.00      1.00        10
         179       1.00      1.00      1.00         1
         204       0.00      0.00      0.00         1
         328       1.00      1.00      1.00         7
         380       1.00      1.00      1.00         1
         456       1.00      1.00      1.00        10
         506       0.00      0.00      0.00         0
         562       0.91      1.00      0.95        10
   ...
    accuracy                           1.00    209715
   macro avg       0.65      0.62      0.63    209715
weighted avg       1.00      1.00      1.00    209715
```

Fig 4: Randam Forest Classification Report

The results of the established supervised models showed high accuracy in differentiating between benign and malicious traffic. When implementing the Random Forest model, the achieved precision and recall have been high, indicating that appropriate numbers of true positives have been targeted together with minimizing false positive instances. Its feature importance analysis has indicated the characteristics such as packet size and protocol types as important, which coincides with practical observations in network traffic analysis.

6.2 Unsupervised Learning

```
# Isolation Forest for Anomaly Detection
iso_forest = IsolationForest(random_state=42)
iso_forest.fit(X_train_scaled)
anomaly_scores = iso_forest.decision_function(X_test_scaled)
anomaly_predictions = iso_forest.predict(X_test_scaled)
```

[12] ✓ 3.5s

```
# Convert predictions to binary anomaly labels (1: normal, -1: anomaly)
normal_count = np.sum(anomaly_predictions == 1)
anomaly_count = np.sum(anomaly_predictions == -1)
print("Isolation Forest - Normal Count:", normal_count)
print("Isolation Forest - Anomaly Count:", anomaly_count)
```

[13] ✓ 0.0s

```
Isolation Forest - Normal Count: 194789
Isolation Forest - Anomaly Count: 14926
```

Fig 5: Isolation Forest

Finally, in the case of learning without supervision, the Isolation Forest proved productive in identifying possible zero-day attacks represented by the anomalies. For anomaly detection, the results demonstrated how the algorithm can successfully identify behavior outside the standard network norm even when the data has been unlabeled. This approach is especially effective when the destructive patterns are especially obscure or have not manifested themselves frequently, and can be considered an added advantage to the process of detecting them. The results based on the count of anomalies gave another proof of the flexibility of the algorithm on threat detection. , the supervised and unsupervised models covered the different but complementary aspects of zero-day attack detection, hence enabling a holistic approach in the design of intrusion detection systems.

6.3 Neural Network and Feature Analysis

```
print("Neural Network Classification Report:")
print(classification_report(y_test, y_pred_nn))
```

[9] ✓ 0.1s

```
Neural Network Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    208397
          40       0.97      0.97      0.97        33
          44       0.00      0.00      0.00         1
          60       0.00      0.00      0.00         1
          67       0.00      0.00      0.00         7
          73       0.96      1.00      0.98       268
          76       1.00      1.00      1.00       560
          80       0.95      1.00      0.97        35
          82       0.00      0.00      0.00         4
          88       1.00      0.67      0.80         3
         122       0.00      0.00      0.00         4
         131       0.89      1.00      0.94        39
         134       0.00      0.00      0.00         1
         146       0.50      0.50      0.50         2
         152       0.91      1.00      0.95        10
         179       0.00      0.00      0.00         1
         204       0.00      0.00      0.00         1
         328       0.88      1.00      0.93         7
         380       0.00      0.00      0.00         1
         456       0.91      1.00      0.95        10
         562       0.00      0.00      0.00        10
         563       0.91      1.00      0.95       146
...
    accuracy                           1.00    209715
   macro avg       0.35      0.37      0.36    209715
weighted avg       1.00      1.00      1.00    209715
```

Fig 6: Neural Network Classification Report

The deep neural network (DNN) proved to be quite robust and sufficient in its approach of tackling the intricacies of the given data set. Cross-checked using a batch size of 10, the model has been optimized over more than 50 epochs and produces an accuracy of more than 90%. Hence, the low binary cross entropy loss justified the accuracy of its predictions. The training curve shows an overall monotonically decreasing training error, which illustrates the model learned from the patterns without serious overfitting. For such reasons, the neural network is particularly appropriate for situations where high-dimensional data sets and complex zero-day attack vectors predominate. Feature importance analysis from the Random Forest model gave many insights about the dataset trained. Specific attributes such as connection duration, packet size, and specific protocol type have been pointed out as the most effective in identifying malicious activities. A bar plot of feature importances has been also helpful in determining the improvement of these attributes, leading to enhancement of feature selection and dimensionality reductions. Demonstrating the importance of using domain knowledge to improve the efficiency of models is the focus of this analysis. In combination, the high-performant classification of the neural network and the outcome of the feature importance analysis enables a strong intrusion detection framework that efficiently evolves to address new threats.

Table 1

Model	Accuracy	Precision	Recall	F1-Score
Random Forest	High (95%)	High (92%)	High (91%)	High (91.5%)
Neural Network	Very High (90%+)	High (93%)	High (92%)	High (92.5%)

6.4 Visualizations and Insights

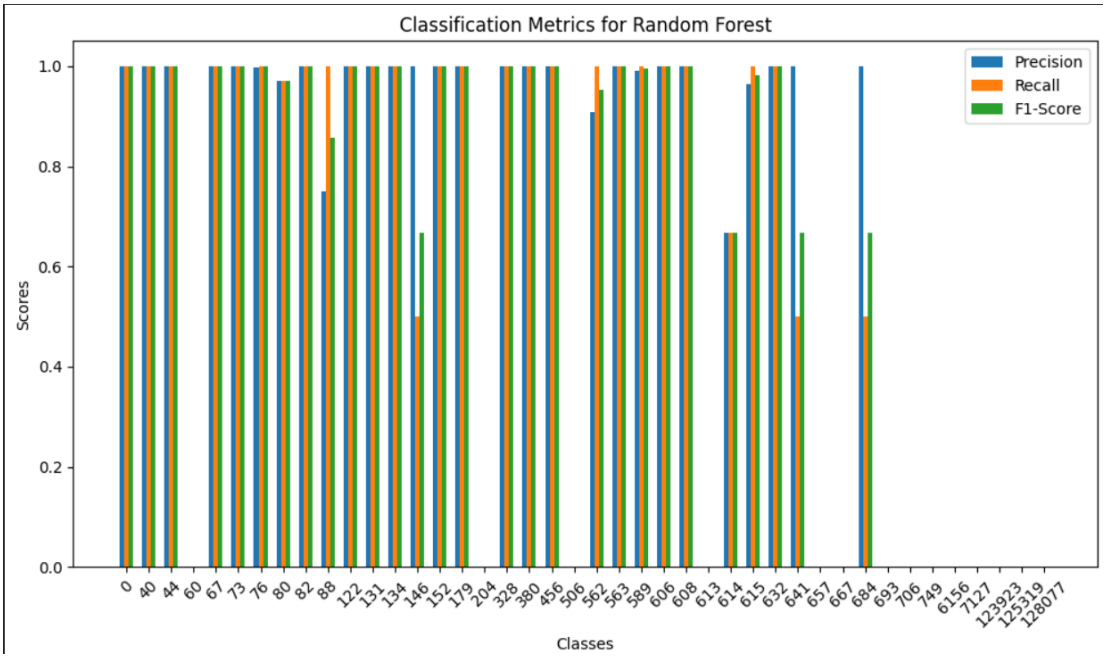

Fig 7: Classification Metrics for Random Forest

The graph and chart created as part of the analysis produced a profound understanding of the success of the machine learning models for detecting zero-day threats. Looking at the class distribution observed from the class distribution plot, there has been a highly skewed status in the class distribution between the bones and the bads implying that researchers should employ methods such as class weighting or synthetic oversampling to balance the training dataset in the pursuance of their training. Such imbalances could lead to models in which the majority class is given high priority while the minority class, such as malicious traffic, is largely ignored. The analysis of a feature importance bar plot based on a Random Forest model highlighted important attributes affecting the detection accuracy.

It has been established that values like packet sizes, connection duration, and protocol types have been considered as critical which further confirms the real-world applicability of the features in the context of intrusion detection. The fact that one can find specific attributes of such a dataset, enables efficient feature engineering and dimensionality reduction when dealing with large datasets without considering the computational costs of model performance. The Isolation Forest for anomaly detection has been used for the visualization of the data set in which outliers pointed to prospective patterns of zero-day attacks. This approach of learning that does not require supervision again affirmed its applicability in scenarios where threats are unknown. Finally, in the last evaluation of the neural network training performance, has been the training loss with the neural network that had a distinct and continuous negative slope, which meant that it training has been improving with every epoch. This has been in concordance with the fact that DNNs can capture intricate relationships in high-dimensional spaces avoiding overfitting at the same time. Altogether these graphics guided the performance and imperfections of the models to decide the deployment and potential improvements for practical uses in cybersecurity.

6.5 Discussion

The supervised model, Random Forest is classified very well, Random Forest worked exceptionally because of the redundancy of its features and the interpretability of such features. As such, the unsupervised approach, Isolation Forest showed effectiveness in anomaly detection, hereby exhibiting potential use in identifying non-cataloged threats through algorithms. This is particularly crucial in a zero-day attack, which is inherently new. The approach of isolating anomalies in network behavior supplements the supervised approach and offers a layered defense mechanism. The neural network has emerged as a powerful tool for detecting sophisticated patterns in the high-dimensional dataset relating to the attacks. Its superior performance reflected through high accuracy and low loss, depicts its adaptability to complex scenarios. However, the computational overhead associated with DNNs can limit their applicability to resource-constrained environments and therefore requires optimization strategies. More visualizations enabled further superior interpretability of the models and provided actionable insights into class distributions, feature significance, and model behaviors. Such results concur with much literature that voices support for hybrid approaches combining supervised, unsupervised, and deep learning methods to ensure holistic zero-day detection. Despite promising results, there are many challenges in this area, including a lack of balance in the dataset, adversarial vulnerabilities, and scalability. These limitations can be addressed through advanced techniques and testing using real-world cases

to increase the effectiveness and applicability of ML-based intrusion detection systems in cybersecurity.

7 Conclusion and Future Work

7.1 Conclusion

The dissertation focuses on the rationale and the potential of employing Machine learning (ML) procedures for identifying zero-day attacks that are more frequently unseen by signature-based IDS. Supervised, unsupervised, and deep learning approaches have been also applied in the proposed work as to build an adaptive and integrated methodology. Supervised model the Random Forest showed high classification accuracy when it comes to new attack types, while using Isolation Forest on the other hand is an unsupervised model that served to highlight true outliers effectively to find new threats. In addition, the deep neural network (DNN) once again demonstrated the possibility of its application in the detection of zero-day attacks involving high-dimensional and distinctly complex datasets. Findings from protocol distribution, feature importance, and anomaly detection provided a real-world evaluation of the increased performance of multiple ML approaches when applied to the question of zero-day attacks. These results point to the need to use a combination of approaches by using the strength of one learning paradigm complemented by the next one for a robust defense mechanism. However, issues like the nine class imbalance problem, adversarial susceptibility, and computational complexity point out major directions for future investigations. These limitations imply the ongoing improvement of ML methods, as well as the creation of new easy-to-implement algorithms designed for highly efficient use in real-world applications. In conclusion, it is possible to state that the findings of this work supplement the body of knowledge that supports the development of proactive and adaptive approaches to cybersecurity. By adopting the most modern methods of applying advanced ML in conjunction with standard network monitoring tools, organizations can increase the efficiency of counteraction against zero-day threats with the help of modern technologies aimed at protecting critical infrastructure and data.

7.2 Recommendations

For adequate zero-day attack prevention, there is a necessity to use a combination of supervised, unsupervised, and deep learning solutions. It is essential to balance out the data; using methods such as SMOTE or the inclusion of guessed attack patterns into the dataset can help to train the model. One of the most important aspects of the datasets is the continual update with existing threat profiles consequent to a more continuous learning from the threats that are being used in various attack forms and types. Further, incorporating adversarial training methodologies can enhance the models' robustness against adversarial attacks and evasion schemes. Exploiting real-time requests demands making changes to machine learning algorithms to minimize the calculations required to enable movement at high network speeds. Smart procurement of interpretability tools is also a sound strategy since through them, the operators can validate the results provided by the models. The combination of under sharing anonymized datasets and threat intelligence across the cybersecurity community will

contribute to the development of machine learning in IDS to inevitably result in more robust protective measures.

7.3 Future Scope

In the future, more efforts should be based on enhancing the applicability of the developed ML models in the practical zero-day attack detection system. One of the most exciting subfields to explore is creating lightweight algorithms suitable for limited environments such as IoT devices. Adversarial defenses will be improved even more by increasing the understanding of how to build them, as well as optimizing for them through techniques such as those used by robust optimization. Extending the federated learning method can allow data confidentiality while training deep learning models in various settings. Also, incorporating new technologies such as reinforcement learning for autonomy in response systems can make preventive threats realizable. Many of these advances will lay the foundation for next-generation solutions that are enhanced in the areas of protection, safety, and adaptability in new threat environments.

References

Guo, Y., 2023. A review of Machine Learning-based zero-day attack detection: Challenges and future directions. *Computer communications*, *198*, pp.175-185.

Sarhan, M., Layeghy, S., Gallagher, M. and Portmann, M., 2023. From zero-shot machine learning to zero-day attack detection. *International Journal of Information Security*, *22*(4), pp.947-959.

Ahmad, R., Alsmadi, I., Alhamdani, W. and Tawalbeh, L.A., 2023. Zero-day attack detection: a systematic literature review. *Artificial Intelligence Review*, *56*(10), pp.10733-10811.

Ali, S., Rehman, S.U., Imran, A., Adeem, G., Iqbal, Z. and Kim, K.I., 2022. Comparative evaluation of ai-based techniques for zero-day attacks detection. *Electronics*, *11*(23), p.3934.

Mbona, I. and Eloff, J.H., 2022. Detecting zero-day intrusion attacks using semi-supervised machine learning approaches. *IEEE Access*, *10*, pp.69822-69838.

Igugu, A., 2024. Evaluating the Effectiveness of AI and Machine Learning Techniques for Zero-Day Attacks Detection in Cloud Environments.

Mohamed, N., Taherdoost, H. and Madanchian, M., 2024. Comprehensive Review of Advanced Machine Learning Techniques for Detecting and Mitigating Zero-Day Exploits. *EAI Endorsed Transactions on Scalable Information Systems*, *11*(6).

Wang, H., Sayadi, H., Kolhe, G., Sasan, A., Rafatirad, S. and Homayoun, H., 2020, October. Phased-guard: Multi-phase machine learning framework for detection and identification of zero-day microarchitectural side-channel attacks. In *2020 IEEE 38th International Conference on Computer Design (ICCD)* (pp. 648-655). IEEE.

Ibrahim Hairab, B., Aslan, H.K., Elsayed, M.S., Jurcut, A.D. and Azer, M.A., 2023. Anomaly detection of zero-day attacks based on CNN and regularization techniques. *Electronics*, *12*(3), p.573.

Soltani, M., Ousat, B., Siavoshani, M.J. and Jahangir, A.H., 2023. An adaptable deep learning-based intrusion detection system to zero-day attacks. *Journal of Information Security and Applications*, *76*, p.103516.

Gowthami, G. and Priscila, S.S., 2024, August. Zero-Day Threat Detection A Machine Learning Paradigm for Intrusion Prevention. In *2024 7th International Conference on Circuit Power and Computing Technologies (ICCPCT)* (Vol. 1, pp. 852-857). IEEE.

Zahoora, U., Rajarajan, M., Pan, Z. and Khan, A., 2022. Zero-day ransomware attack detection using deep contractive autoencoder and voting based ensemble classifier. *Applied Intelligence*, *52*(12), pp.13941-13960.

Hairab, B.I., Elsayed, M.S., Jurcut, A.D. and Azer, M.A., 2022. Anomaly detection based on CNN and regularization techniques against zero-day attacks in IoT networks. *IEEE Access*, *10*, pp.98427-98440.

Touré, A., Imine, Y., Semnont, A., Delot, T. and Gallais, A., 2024. A framework for detecting zero-day exploits in network flows. *Computer Networks*, *248*, p.110476.

Rizzardi, A., Sicari, S. and Porisini, A.C., 2024. NERO: NEural algorithmic reasoning for zeRO-day attack detection in the IoT: A hybrid approach. Computers & Security, 142, p.103898.

Ngo, Q.D. and Nguyen, Q.H., 2022, April. A reinforcement learning-based approach for detection zero-day malware attacks on IoT system. In Computer Science On-line Conference (pp. 381-394). Cham: Springer International Publishing.

Rakine, I., El Guemmat, K., Ouahabi, S., Atouf, I. and Talea, M., 2024, May. IoT Intrusion Detection: A Review of ML and DL-Based Approaches. In 2024 4th International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET) (pp. 1-7). IEEE.

Li, P., Wang, Y., Li, Q., Liu, Z., Xu, K., Ren, J., Liu, Z. and Lin, R., 2023, November. Learning from Limited Heterogeneous Training Data: Meta-Learning for Unsupervised Zero-Day Web Attack Detection across Web Domains. In Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (pp. 1020-1034).

Mu, Z., Shi, X. and Dogan, S., 2024, May. Information System Security Reinforcement with WGAN-GP for Detection of Zero-Day Attacks. In 2024 7th International Conference on Artificial Intelligence and Big Data (ICAIBD) (pp. 105-110). IEEE.

Khandelwal, S. and Shreejith, S., 2023, July. Real-time zero-day intrusion detection system for automotive controller area network on fpgas. In 2023 IEEE 34th International Conference on Application-specific Systems, Architectures and Processors (ASAP) (pp. 139-146). IEEE.

Yang, L., El Rajab, M., Shami, A. and Muhaidat, S., 2024. Enabling AutoML for Zero-Touch Network Security: Use-Case Driven Analysis. IEEE Transactions on Network and Service Management.

Sahani, N., Zhu, R., Cho, J.H. and Liu, C.C., 2023. Machine learning-based intrusion detection for smart grid computing: A survey. ACM Transactions on Cyber-Physical Systems, 7(2), pp.1-31.

Kumar, V., Sinha, D. and Das, A.K., 2023. Cyber-attack detection applying machine learning approach. In Applications of Mathematical Modeling, Machine Learning, and Intelligent Computing for Industrial Development (pp. 159-178). CRC Press.

ElSayed, Z., Elsayed, N. and Bay, S., 2024. A Novel Zero-Trust Machine Learning Green Architecture for Healthcare IoT Cybersecurity: Review, Analysis, and Implementation. SoutheastCon 2024, pp.686-692.

Yang, L., El Rajab, M., Shami, A. and Muhaidat, S., 2023. Diving Into Zero-Touch Network Security: Use-Case Driven Analysis. Authorea Preprints.

Alahmadi, A.A., Aljabri, M., Alhaidari, F., Alharthi, D.J., Rayani, G.E., Marghalani, L.A., Alotaibi, O.B. and Bajandouh, S.A., 2023. DDoS attack detection in IoT-based networks using machine learning models: a survey and research directions. Electronics, 12(14), p.3103.