Identifying the Origin of Cyber Attacks Using Machine Learning and Network Traffic Analysis

Tianqing Du

Teda International School No. 72, 300457 3rd Avenue Teda Tianjin, China

Abstract. In this paper, PCAP refers to Packet Capture, Network Intrusion Detection Systems refers to NIDS, Artificial Intelligence refers to AI, machine learning refers to ML, Computer Vision refers to CV, and Natural Language Processing refers to NLP. While the development of the internet promotes global progress, it also brings various cyber-attacks, such as phishing, junk emails, and keylogging. To ensure a clean internet environment, it is essential to identify the origin of cyber-attacks for effective defense and mitigation. This paper introduces an effective method of internet protection—machine learning. A common technique in the modern world, machine learning offers significant insights into locating the IP address and data origin. The focus of this paper is on how supervised machine learning is used to determine the data origin. The Random Forest Classifier is the key model analyzing network traffic data to predict the origin of cyber-attacks. By converting IP addresses, packet lengths, and protocol types into numerical features from PCAP files, this study applies machine learning techniques to classify attack behaviors. Additionally, an experiment testing the model's effectiveness is designed to prove its efficiency and ensure the model's precision.

1 Introduction

With the rapid development of 5G, the internet, IoT, and cloud computing, network complexity and traffic volume have increased significantly, leading to a rise in sophisticated cyber-attacks. These advancements pose significant challenges to internet environments and cybersecurity. Therefore, effective detection and response are crucial. NIDS, a second line of defense behind firewalls, plays an important role in identifying malicious attacks on the internet and providing real-time monitoring and protection.

Around 1980, James Anderson first introduced the concept of intrusion detection, although it was initially constrained by limited computing resources. However, recent advancements in computing power and AI have enabled the application of machine learning (ML) methods to network security [1, 2]. Various studies have demonstrated the effectiveness of ML in identifying different types of cyber-attacks [3-5]. Despite this progress, the imbalance between normal and malicious traffic data presents a significant challenge since most network traffic is benign, making it difficult for ML models to accurately detect rare attacks [3, 5]. Deep learning, a subfield of ML, has shown remarkable success in fields like

Corresponding author: andrew2902@tedais.net

computer vision (CV) and natural language processing (NLP), and its application to intrusion detection is gaining momentum [2, 4]. Deep learning models can extract high-dimensional data features and convert anomaly detection into a classification problem, improving real-time processing capabilities [1, 2]. However, the imbalance in network traffic data remains a challenge. To address this, the Difficult Set Sampling Technique (DSSTE) is proposed to handle class imbalance in network traffic data. This novel algorithm reduces the number of majority class samples and augments minority class samples, improving classifier performance. Researchers validate the approach using classic and contemporary datasets, such as NSL-KDD and CSE-CIC-IDS2018, and evaluate several ML and deep learning algorithms, including Random Forest, Support Vector Machine, XGBoost, Long Short-Term Memory (LSTM), AlexNet, and Mini-VGGNet [6-10].

2 Methodology

The overall data set was collected from the internet, which is called traffic from the GitHub websites. Network traffic data was collected from PCAP files, which contained raw packet information from a network under observation. These files store comprehensive details about every packet transmitted, including source and destination IP addresses, protocols used, and packet sizes.

2.1 Data Collection

The overall dataset was collected from the internet, specifically traffic from GitHub websites. Network traffic data was collected from PCAP files, which contain raw packet information from a network under observation. These files provide detailed information about the dataset, such as the packet transmission process and the source and destination IP addresses. For this study, the data was specifically focused on packets that include identifiable IPv4 layers. These layers provide valuable insights for model training, especially for identifying the origin of the network traffic.

2.2 Feature Extraction

From the collected PCAP data, the following features were extracted to serve as input for the machine learning model

- Source IP Address contains the IP addresses for the data set.
- Destination IP Address indicates the intended recipient of the packet.
- Protocol Type defines the communication protocol used (e.g., TCP, UDP).
- Packet Length demonstrates the represents the size of the packet in bytes.

These features help to provide essential information about the flow of traffic between each packet and allow the model to detect similar patterns that could indicate abnormal activity.

2.3 Data Preprocessing

Since the dataset traffic was extracted from the internet, it was crucial to transcribe the data into accessible PCAP files. The conversion process included several key feature transformations.

IP addresses, initially string values, were converted into numerical integers using Python's IP address module. This transformation enables the model to process IP addresses as numerical data points.

Protocol type was converted into numerical labels using a LabelEncoder from the sklearn library, allowing the model to handle categorical data representing different communication protocols.

The dataset was divided into training and testing subsets using an 80/20 split, ensuring the model could be trained on one portion of the data and evaluated on unseen data to measure its performance.

2.4 Model Selection: Random Forest Classifier

The Random Forest classifier offers several significant advantages when selecting models, such as robustness, the ability to handle high-dimensional data, and stability in the presence of outliers. It improves classification precision by constructing multiple decision trees; this ensemble learning technique reduces overfitting and enhances prediction accuracy.

A grid search was performed to optimize the model's hyperparameters. The fine-tuned parameters are as follows.

The number of trees in the forest is (100, 200, 300), and Max depth measures each tree's maximum depth (10, 20, 30).

The GridSearchCV method from sklearn was used to evaluate the model with different parameter combinations through cross-validation, ultimately selecting the best-performing model based on balanced accuracy.

2.5 Model Training and Evaluation

The Random Forest model was trained on processed data, and its performance was evaluated through several metrics, including accuracy, precision, recall, and F1-score. A confusion matrix was constructed to visualize the model's performance and classification results.

3 Experimental Results

3.1 Best Model and Parameters

After performing a grid search, the best-performing Random Forest model had the following hyperparameters in Table 1.

Table 1. Best performing table

N estimators	Max depth
200	20

This model achieved the highest balanced accuracy on the validation set, demonstrating its ability to classify network traffic based on extracted features.

3.2 Confusion Matrix

The model's ability to predict various protocol types is demonstrated through the confusion matrix. Each row represents the actual protocol class, while each column represents the predicted class. Fig. 1 shows the confusion matrix, where correct classifications are indicated by diagonal values, and incorrect classifications are represented by off-diagonal values.

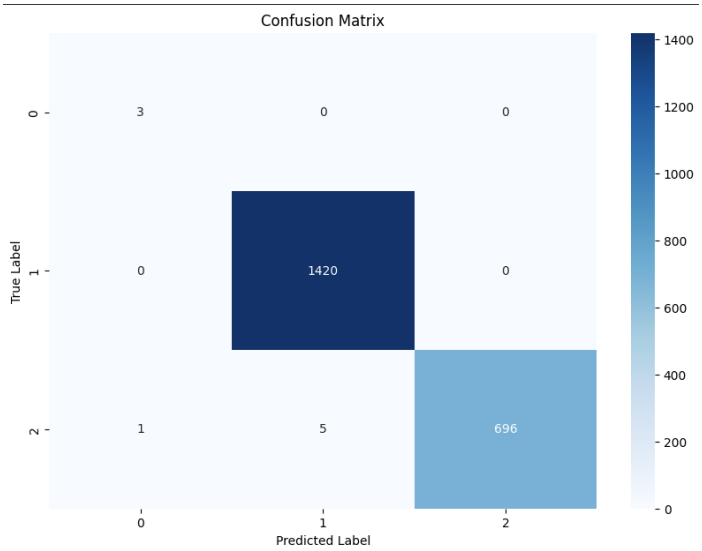

Fig. 1. Confusion matrix(Picture credit : Original)

3.3 Feature Importance

The Random Forest model also provides insight into the importance of each feature in the classification process. One notable feature was the Source IP Address, followed by the Destination IP Address and Packet Length, indicating that the origin and target of network packets are key indicators for identifying potential malicious activity. All three elements are shown in Fig. 2.

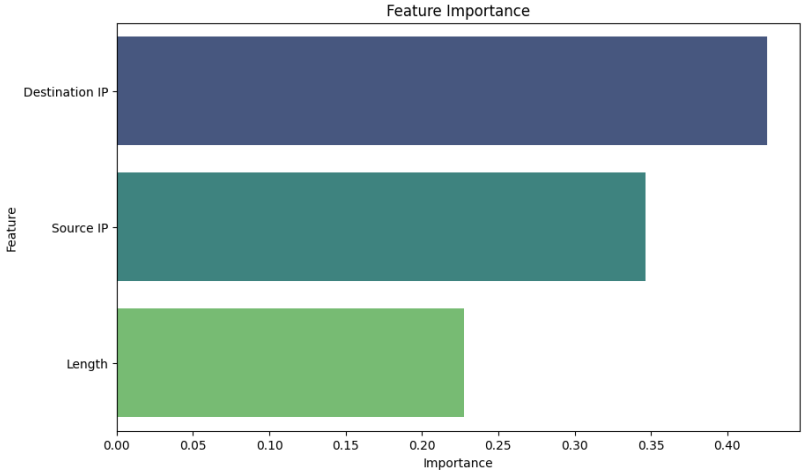

Fig. 2. Feature importance(Picture credit : Original)

3.4 Discussion

The results indicate that network traffic patterns, particularly the IP addresses and packet sizes, play a crucial role in determining the source of cyber-attacks. While this approach is effective in classifying protocols, additional research is needed to refine the model for

identifying the country of origin based on network traffic. Incorporating external data sources such as geographical IP mapping or real-time threat intelligence feeds could further enhance the accuracy of this method.

The limitation of this study lies in the use of protocol classification as a proxy for attack behavior. While protocol classification is informative, a more comprehensive analysis incorporating payload inspection and behavioral patterns of known attack signatures could improve attribution accuracy. Additionally, the randomization and obfuscation techniques used by sophisticated attackers may still challenge machine learning-based attribution methods.

3.5 Experiment

In the model, to test whether it can successfully identify IP addresses, the dataset was divided into two separate sets: a testing set and a training set. In the testing set, the data were divided into three classes to calculate the model's accuracy, as shown in Table 2.

Table 2. Experiment Data Analysis

class	precision	recall	F1-score	Support
1	0.75	1	0.86	3
2	1	1	1	1420
3	1	0.99	1	702

For Class 0, the precision is 0.75 and the recall is 1, indicating that the model correctly identified 75% of the data belonging to Class 0, achieving a recall rate of 100%. However, due to the small sample size, this result has limited statistical significance. The F1 score, which is the harmonic mean of precision and recall, is 0.86, showing good performance for Class 0.

For Class 1, both the precision and recall are 1, highlighting that the model's predictions for Class 1 are strong. All data predicted as Class 1 are correct, and all samples that actually belong to Class 1 are correctly identified. Additionally, the F1 score is perfect, indicating excellent performance for this class. The support is 1,420, meaning that with a large number of samples, the classification results for this class are statistically strong.

For Class 2, the precision is also 1, and the recall is 0.99, indicating nearly perfect performance for this class. The model makes almost no errors in predicting Class 2, with only a few actual Class 2 samples being misclassified. The F1 score is 1, remaining high and indicating very good classification performance for Class 2.

The model performs exceptionally well on Class 1 and Class 2, with a large number of samples in these two classes, leading to near-perfect classification results.

Class 0 has a very small number of samples. Although the recall is 100%, the precision is lower (0.75), indicating some potential misclassifications. Due to the extremely small sample size in Class 0 (only 3 samples), this result could be significantly influenced by the sample size.

Overall, the model performs excellently, particularly demonstrating stability with a large number of samples in Class 1 and Class 2. Despite the issue of imbalanced data, the model still provides strong results, especially in terms of precision and recall.

4 Conclusion

The study highlights the potential of using machine learning models, especially Random Forests, to analyze network traffic and identify the source of cyber-attacks. By extracting key features from network packets, such as IP addresses and packet lengths, the model successfully predicts network protocols and provides insights into network behaviors. Future

work will involve extending this approach to identify not only protocols but also the countries of origin of cyber-attacks, using more advanced network analysis techniques.

5 References

1. Y. Zhang, X. Wang, Q. Li, Y. Chen, An efficient cybersecurity attack detection model using machine learning techniques. Cluster Comput. **2024**, s10586-024-04662-6 (2024)
2. J. Li, H. Zhang, T. Wong, A novel deep learning-based approach for network intrusion detection. In: P. Zhou, Q. Lin (Eds.), Advances in Data Science and Cybersecurity, (2024), 85-97
3. Y. Zhou, L. Huang, X. Zhang, S. Liu, A review of network security monitoring based on machine learning techniques. J. Inf. Secur. Appl. **60**, 102864 (2021)
4. D. George, S. Liu, Enhancing cybersecurity threat detection using machine learning. J. Cybersecurity Res. **14**, 112-125 (2021)
5. T. Jones, P. Smith, Advanced threat detection using AI-driven techniques. Network Secur. **2018**, 10-14 (2018).
6. P. Zhou, Q. Lin, Enhancing cybersecurity with machine learning models. Advances in Data Science and Cybersecurity, (2022), 85-97
7. S. G. M. Rahman, M. S. Hossain, K. A. M. Khair, A hybrid machine learning approach for intrusion detection systems. J. Netw. Comput. Appl. **123**, 90-99 (2019)
8. R. B. K. Prabhu, A. N. Jadhav, M. R. Patil, A comprehensive review on machine learning-based intrusion detection systems. Int. J. Inf. Secur. **20**, 1-24 (2021)
9. A. A. M. Khedher, A. M. Khlifi, M. A. Abid, Anomaly-based intrusion detection system using deep learning techniques. Future Gener. Comput. Syst. **106**, 263-275 (2020)
10. T. Alharbi, M. M. Khan, R. S. Alzahrani, Performance evaluation of machine learning algorithms for intrusion detection systems. J. King Saud Univ. Comput. Inf. Sci. **34**, 571-580 (2022)