41180 Assessment task 3: Network Intrusion Detection System

Group 16

Jiapeng Yang 24580312

# Contents

[Introduction 3](#_Toc167464514)

[Intrusion Detection System 3](#_Toc167464515)

[Network environment, attack and defense description 4](#_Toc167464516)

[Monitoring and reporting using NIDS on the Defender's computer 4](#_Toc167464517)

[Implementation, description, and results of the code 6](#_Toc167464518)

[Reflection 13](#_Toc167464519)

[Conclusion 14](#_Toc167464520)

[Reference 15](#_Toc167464521)

# Introduction

The rapid development of information technology has led to the internet and digitization becoming an indispensable part of daily life. The advent of smartphones, laptops, smart TVs, and smart appliances has transformed the way we live our lives. However, with the increasing prevalence of the internet, the nature of network threats and the field of network security intervention are also undergoing constant evolution. These issues not only threaten personal privacy but may also have serious implications for corporate and national security. Consequently, the assurance of network security has become a crucial undertaking in the contemporary information society.

Network security is defined as the set of measures, technologies, or practices employed to prevent or mitigate the risk of unauthorized access to networks, devices, and data. This involves the protection of computer systems and networks from threats and attacks, including information leakage, theft, destruction, or service interruption. The primary objective of network security is to guarantee the confidentiality, integrity, and availability of data. Among the numerous network security technologies, intrusion detection systems (IDS) occupy a pivotal position.

# Intrusion Detection System

IDS is a network security tool that is employed to monitor network traffic and devices with the objective of detecting known malicious activities, suspicious activities, or violations of security policies. Upon the detection of security risks and threats, the IDS will issue alerts to the IT and security teams.

Depending on its location within the system and the type of activity being monitored, IDS can be divided into two distinct categories: Network Intrusion Detection Systems (NIDS) and Host Intrusion Detection Systems (HIDS). Network intrusion detection systems (NIDS) are deployed at critical locations within the network, with the objective of monitoring all traffic that passes through. In contrast, host intrusion detection systems (HIDS) are deployed on an independent host, with the aim of monitoring the traffic entering and exiting the host. This report primarily concerns the use of NIDS to defend against attacks.

# Network environment, attack and defense description

The objective of this experiment is to construct a virtual network environment that will simulate scenarios of network attacks and defenses. In this experiment, both the attacker and the defender are involved. The role of the attacker is to generate and send malicious traffic, whereas the defender is responsible for deploying an intrusion detection system (NIDS) and monitoring network traffic to detect any abnormal behavior. The normal and attack traffic in the network will be simulated using the datasets provided in Lab10. The datasets comprise a substantial quantity of both normal and anomalous traffic data, which is well-suited for the training and testing of NIDS models.

The role of attackers

Attackers simulate network attacks and generate malicious traffic to test their defense capabilities.

The role of defenders

Defenders deploy NIDS to monitor network traffic in real-time, identify and detect abnormal behavior, and protect network security. Real time monitoring of data packets passing through the network, capturing and analyzing traffic characteristics. Using the NIDS model to detect abnormal traffic in the network, such as DoS DDoS and port scanning attacks. When an attack is detected, timely report and take corresponding defense measures to ensure network security.

## Monitoring and reporting using NIDS on the Defender's computer

### The basic principles of NIDS

A Network Intrusion Detection System (NIDS) is a security measure employed to monitor network traffic and identify suspicious activities and known threats. Network intrusion detection systems (NIDS) are typically deployed at critical locations within the network, such as behind firewalls or routers, with the objective of monitoring all traffic passing through. Upon the detection of a potential attack, the NIDS will generate alerts and transmit them to the network administrator or security team.

### Steps for monitoring and reporting using NIDS

Firstly, NIDS captures packets from network traffic and extracts useful features, including protocol type, source IP address, destination IP address, port number, and packet size. Subsequently, the captured data packets are analysed, the pertinent features extracted, and these features converted into a format that the NIDS can process. The extracted features from NIDS analysis are subjected to analysis through the application of predefined security rules and algorithms, with the objective of detecting any anomalous behavior. In the future, NIDS will employ a variety of detection methods, including signature detection and anomaly detection, to identify potential attacks and suspicious activities. Upon the detection of a potential intrusion, the NIDS will generate alerts and furnish detailed information to the network administrators or security teams. This information will include the type of attack, the source address, the target address, and the timestamp. Finally, the security team implements the appropriate measures based on the alert information, such as blocking suspicious IP addresses, adjusting firewall rules, isolating infected systems, or conducting further investigation into the incident.

### Using security rules within NIDS to identify and respond to network attacks

**Signature detection**

Signature detection is a method of detection based on known attack features (signatures). Network intrusion detection systems (NIDS) employ a predefined attack signature library to identify features in network traffic. Upon successful matching, the NIDS will recognize the attack and generate an alert.

**Anomaly detection**

Anomaly detection identifies traffic that deviates significantly from normal behavior by learning patterns of normal network traffic. Upon the detection of anomalous behavior, the NIDS will generate alerts.

# Implementation, description, and results of the code

文本

描述已自动生成

Firstly, import the required library and set the drawing parameters.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

Then load the data and print out the first four lines, review their basic information, ensure that the data is correct and understand the structure.

图形用户界面, 文本, 应用程序

描述已自动生成

Train. describe() generates descriptive statistical information for the training dataset. Print the count of each unique value in the num'outbound\_cmds column of the training and testing datasets. Remove the num'outbound\_cmds column from the training and testing datasets. Because the num'outbound\_cmds column is considered redundant and does not provide useful information. Finally, print the count of each attack category in the training dataset.

Through this step, we can understand the basic statistical information of the data, check and process redundant features, and view the distribution of attack types.

图形用户界面, 文本

描述已自动生成

In this step, numerical attributes are standardized to ensure they have zero mean and unit variance. This helps to improve the performance and stability of many machine learning algorithms.

文本

描述已自动生成 In the aforementioned code, the classification attributes are converted into numerical form and merged with numerical attributes for the purpose of feature selection and model training. The output of train\_x.shape is (25192, 40), indicating that the merged training dataset has 25,192 rows and 40 columns. The output of test\_df.shape is (22544, 40), indicating that the merged test dataset has 22544 rows and 40 columns. The outputs demonstrate that the features of the dataset have been successfully standardized and encoded, rendering them suitable for feature selection and model training.

图表

描述已自动生成 The code depicted in the preceding image generates a random forest classifier and applies it to the training set. Finally, the most significant features are extracted and a bar chart is generated to illustrate the relative importance of each feature. The results indicate that src\_bytes and dst\_bytes are the most significant features.

图形用户界面, 文本, 应用程序

描述已自动生成

The code was generated by importing the RFE and itertools modules, creating an RFE model and selecting 15 features, and finally summarizing and displaying the output. The aforementioned steps allow for the identification and selection of the most crucial features, which can then be employed to inform decisions pertaining to data preprocessing, the identification of the most influential features for model performance during training, and the enhancement of the predictive capacity of the model.

图形用户界面, 文本, 应用程序

描述已自动生成

This code uses the train\_test\_split function to divide the dataset into training and testing sets.

The partitioned dataset includes:

* X\_train: The feature data of the training set.
* X\_test: The feature data of the test set.
* Y\_train: The target variable of the training set.
* Y\_test: The target variable of the test set.

In intrusion detection systems (NIDS), a variety of machine learning and deep learning-based methods are employed to identify the specific types of attacks. The following are several methods for implementation. The following machine learning and deep learning-based methods are employed: KNN, LGB, BGB, and Decision Tree . The results will be demonstrated by means of a simulation of attacks and the subsequent use of NIDS for the purpose of defense.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

This code trains multiple different machine learning models, including KNN, LGR, BNB, and Decision Tree will be evaluated to select the model that performs best in detecting network intrusions.

**文本, 应用程序

描述已自动生成**

**手机屏幕截图

描述已自动生成表格

描述已自动生成**

**手机屏幕截图

描述已自动生成表格

描述已自动生成** The preceding results indicate that the Naive Bayes and Logistic Regression models exhibit satisfactory performance, although there are instances of misclassification. The K-Nearest Neighbors classifier model performs exceedingly well, with nearly all samples correctly classified. The Decision Tree model performs optimally on the training data set, but there is a potential for overfitting, which necessitates further validation on the test data set.

图形用户界面, 文本, 应用程序

描述已自动生成

The code is employed to assess the efficacy of the model on test data. Each model generates an accuracy score, a confusion matrix, and a classification report, which are then printed to determine the performance of the models in use.

Naive Bayes：

* Accuracy: 0.91
* The prediction of normal traffic performs well, but there is a certain degree of error in predicting abnormal traffic.

Decision Tree：

* Accuracy: 0.9947
* The performance is very good, with only a very small amount of misclassification.

KNeighborsClassifier：

* Accuracy: 0.9917
* The performance was very good, and almost all samples were correctly classified.

Logistic Regression：

* Accuracy: 0.9546
* Performed well, but there were some misclassifications.

The aforementioned methodologies were employed to simulate the attack and demonstrate the efficacy of the defensive measures. The Decision Tree and KNighborsClassifier models demonstrated the greatest performance on both training and testing data, and thus have the potential to significantly enhance the accuracy and reliability of intrusion detection in practical applications.

# Reflection

The objective of this experiment was to investigate the implementation and application of Network Intrusion Detection Systems (NIDS) in depth. To this end, a virtual network environment was constructed and simulated. This not only enhances our understanding of network security and intrusion detection technology, but also cultivates our ability to apply machine learning methods in practical scenarios.

Learning outcomes

1. Data preprocessing and feature selection:

* We have learned how to extract useful information from raw data and preprocess the data through methods such as standardization and encoding.
* Through Recursive Feature Elimination (RFE), we can select the most important features for the model, thereby improving its performance and efficiency.

1. Application of decision tree model:

* The use of the decision tree model is not only simple and intuitive, but also performed well in this experiment. We have learned how to use decision tree models for classification and gained a deeper understanding of the decision basis of the model through the visualization of important features.

1. Model evaluation and comparison:

* By comparing different models, such as decision trees Through the evaluation and comparison of KNighborsClassifier and Logistic Regression, we can clearly see the performance differences of each model in terms of accuracy, precision, recall, and F1 score.
* This comparative analysis helped us select the most suitable model for intrusion detection and clarified the advantages and disadvantages of each model.

1. Simulate attacks and defense strategies:

* We successfully simulated network attacks and used NIDS for defense. This process not only validated the effectiveness of our model, but also helped us understand the importance and working principle of NIDS in practical applications.
* By monitoring network traffic in real-time and identifying abnormal behavior, we can effectively detect and defend against various network attacks, such as DoS DDoS and port scanning, etc.

Challenges and Gains

During the experiment, we encountered several challenges, including the complexity of data preprocessing, the accuracy of feature selection, and the risk of model overfitting. Nonetheless, these challenges have also served as invaluable learning opportunities. Through a comprehensive literature review, team discussions, and repeated experiments, we were able to not only resolve the aforementioned issues but also gain invaluable experience.

# Conclusion

This project successfully implemented a machine learning-based network intrusion detection system (NIDS) and demonstrated its effectiveness by simulating a network environment. The primary methodology employed for the identification and defense against a range of network attacks was the use of decision tree classifiers. The project team achieved the following main accomplishments throughout the entire project process:

1. Data preprocessing and feature selection:

* We ensure data quality by handling redundant features and standardizing numerical attributes.
* The recursive feature elimination (RFE) method was used to select the most important features and optimize model performance.

1. Implementation and evaluation of machine learning models:

* The decision tree classifier performed well in our experiment, with high accuracy, and was able to effectively classify normal and abnormal traffic.
* We also evaluated other machine learning models, such as KNighborsClassifier and Logistic Regression, and found that the decision tree model performed the best.

1. Simulated attacks and NIDS defense:

* Successfully simulated various network attacks and detected and responded to these attacks in real-time through NIDS.
* The results indicate that machine learning based NIDS can effectively monitor network traffic, identify and defend against potential intrusion behaviors.

This project has enabled me to not only master the fundamental principles and technologies of network intrusion detection, but also to enhance our capacity to apply machine learning methods in practice. This has established a robust foundation for my future research and practical applications. In the future, I intend to further investigate and apply more advanced deep learning methods to enhance the performance and resilience of intrusion detection systems.

# Reference

1. Ahmad, I., Basheri, M., Iqbal, M. J., & Rahim, A. (2018). Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection. *IEEE Access*, 6, 33789-33795.
2. Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys & Tutorials*, 18(2), 1153-1176.
3. Chandrasekhar, P., & Raghuveer, K. (2013). Intrusion detection technique by using K-means, Fuzzy Neural Network and SVM classifiers. *International Conference on Computer Communication and Informatics (ICCCI)*, 1-7.
4. Javaid, A., Niyaz, Q., Sun, W., & Alam, M. (2016). A deep learning approach for network intrusion detection system. *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*, 21-26.
5. Khan, L. U., Yaqoob, I., Tran, N. H., Kazmi, S. M. A., Dang, T. N., & Hong, C. S. (2020). Edge-computing-enabled smart cities: A comprehensive survey. *IEEE Internet of Things Journal*, 7(10), 10200-10232.