A PROJECT REPORT ON

NETWORK INTRUSION DETECTION SYSTEM.

Submitted to the partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE & ENGINEERING

Submitted by

AKSHAY DAWAR [Reg No:RA2011028030017]
RVS PRANAV [Reg No:RA2011028030018]
SOHEN MONDAL [Reg No:RA2011028030010]

Supervised by:

MS. ANJALI MALIK
Assistant Professor

SRM Institute of Science and Technology
Delhi NCR Campus, Modinagar,
Ghaziabad (UP)-201204

MAY 2024

Bonafide Certificate

Certified that this project report titled "Network Intrusion Detection System using ML" is the bonafide work of "RVS PRANAV [Reg No: RA2011028030018]", "AKSHAY DAWAR [Reg No: RA2011028030017]", "SOHEN MONDAL [RA2011028030010]", who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation based on which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Ms. Anjali Malik
GUIDE
Associate Professor
Dept. of Computer Science & Engineering

SIGNATURE

Dr. Avanish Vashisht
HEAD OF THE DEPARTMENT
Dept. of Computer Science & Engineering

Signature of the Internal Examiner

Signature of the External Examiner

ACKNOWLEDGEMENTS

We would like to express my heartfelt appreciation to Ms.Anjali Malik, Assistant Professor and Project Supervisor at SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, for her invaluable insights and expertise in the subject matter, which motivated us to work diligently.

Our profound gratitude goes out to Dr. Jitendra Singh and Mrs.Abhilasha Singh, Project Coordinators at SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, for their enlightening guidance and skillful coordination, which served as a perpetual source of inspiration.

We would also like to extend my sincere thanks to Dr. S. Vishwanathan, Director of SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, for his unwavering support that enabled us to undertake and complete our project work.

Our special thanks go to Dr. D. K. Sharma, Dean (Academics), and Dr. R. P. Mahapatra, Dean (E&T) at SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, for their valuable guidance and unconditional support.

We would like to express my gratitude to Dr. Avanish Vashisht, Head of the Department of Computer Science and Engineering at SRM Institute of Science and Technology, Delhi-NCR Campus, Modinagar, for his suggestions and encouragement in completing this project.

We also owe thanks to all the teaching and non-teaching staff members of our college who provided us with direct or indirect help throughout our studies and project work. Finally, we would like to express our sincere appreciation to our parents, family members, and friends for their unwavering support and encouragement, and to all our well-wishers.

AKSHAY DAWAR [Reg No:RA2011028030017]

RVS PRANAV [Reg No:RA2011028030018]

SOHEN MONDAL [Reg No:RA2011028030010]

DECLARATION

We, AKSHAY DAWAR [Reg No:RA2011028030017] , RVS PRANAV [Reg No:RA2011028030018] , SOHEN MONDAL [Reg No:RA2011028030010] hereby declare that the work which is being presented in the project report "Network Intrusion Detection System using ML" is the record of authentic work carried out by us during the period from January 23 to May 23 and submitted by us in partial fulfillment for the award of the degree "Bachelor of Technology in Computer Science and Engineering" to SRM IST, NCR Campus, Ghaziabad (U.P.). This work has not been submitted to any other University or Institute for the award of any Degree/Diploma.

AKSHAY DAWAR [Reg No:RA2011028030017]

RVS PRANAV [Reg No:RA2011028030018]

SOHEN MONDAL [Reg No:RA2011028030010]

ABSTRACT

Among the most important issues facing modern society is network security. The weaknesses in network security have grown in importance over the last ten years due to the internet's rapid expansion and widespread use. To improve accuracy and efficiency in identifying possible security breaches, this study suggests a Network Intrusion Detection System (NIDS) that makes use of Machine Learning (ML) capabilities. The suggested NIDS seeks to evaluate network traffic patterns and spot unusual behaviors suggestive of cyber threats by utilizing a variety of machine learning approaches, including ensemble methods, supervised learning, and unsupervised learning. Additionally, using carefully labeled datasets, we will train the algorithm to identify patterns linked to both benign and malevolent network activity. This study shows how useful the Knowledge Discovery and Data Mining (KDD) dataset is assessing machine learning techniques. It focuses mostly on the KDD preparation step to provide a credible and fair experimental data set.

Tables of Contents

ACRONYMS AND ABBREVIATION

ACL: Access Control List

DDOS: Distributed Denial of Service

DNS: Domain Name Server

DOS: Denial of Service

HTTP: Hyper Text Transfer Protocol

IP: Internet Protocol

NIDS: Network Intrusion Detection System

TCP: Transmission Control Protocol

UDP: User Datagram Protocol

R2L: Remote to Local User

U2R: User to Root Access

LIST OF FIGURES

INTRODUCTION

- In this moder era protection of individual property has become very important. The new ways through which cyber-attacks happen have become very concerning to the authorities. In times like this Machine Learning and NIDS (Network intrusion detection system has come as handy for solving our various problems.

- The old or traditional methods of tracking intrusion have long been left behind by the new ways how cyber have become prevalent in these times. Therefore, ML-based intrusion detection systems provide a standard way of using algorithms to detect pattern autonomously and detect oddity from traffic network, providing more accurate and dynamic threat detection. As the rule-based systems are vulnerable to huge amount of false positive rates, limited scalability, thereby losing their efficacy in detecting intrusions.

- We would define the underlying rules and principles that govern the many principles of ML driven models in cybersecurity. By deep diving into the details of ML algorithms, feature selection strategies and preprocessing data techniques we provide cybersecurity researchers with tools needed to use ML-based intrusion detection systems.

- Moreover, we can employ a variety of models rather than just a few to determine the best course of action when dealing with data silos issues. We go over every framework available for successfully refining various machine learning algorithms. We also go through methods for securing our data.

Keywords:

1. **Machine Learning:** Machine learning is a disciplinary of computer science that uses statistic and proven data, artificial intelligence to train the computer to take decision depending on the different models made of the data. In traditional programming the computer was already provided with a vast amount of data but here the machine is trained to identify patterns and relations so that they can generalize the whole data and come up with a prediction or value.

   ML algorithms has a wide range of techniques, like linear regression, support vector machines, neural networks, and more. Deep learning, a subset of ML, uses artificial neural networks layers to automatically learn hierarchical representations of data, which in turn help in image recognition, natural language processing, and autonomous driving.

   The applications of ML are vast and diverse, spanning industries such as healthcare, finance, marketing, cybersecurity, and beyond. From personalized recommendations on streaming platforms to medical diagnosis assistance and fraud detection in financial transactions, ML has revolutionized how businesses operate and how we interact with technology.

2. **Security:** Security is one of the most important parts of Network Intrusion Detection Systems (NISD) to protect against cyber threats. NIDS acts as a guard against any potential security breaches. It uses a various technique together like signature-based detection, anomaly detection, and increasingly, machine learning algorithms, it can identify and malware attack, DDOS attack and any user with a suspicious intention.

   Any NIDS would employ a multi-layered straggle to defend, signature base detection with techniques like machine learning and anomaly detection. This allows the NIDS to provide adequate protection against both known and unknown threats, including zero-day attacks. Keeping regular updates is important to keep pace with evolving attacks making sure that the system remain vigilant against attack vectors.

   In summary, security in Network Intrusion Detection Systems encompasses a proactive and adaptive approach to threat detection, leveraging advanced technologies and strategies to safeguard network integrity and protect against cyber threats in an ever-evolving landscape.

3. **Intrusion Detection:** In this new age intrusion detection has become a very important concept. With the rapid inter connection of devices and exponential growth of data it is no longer be useful to use traditional methods so we employ new techniques like machine learning, big data analysis, A.I to bolster our defense.

Today's NIDS are adept to detect a large variety of threats ranging from identify anomalies, detect pattern which indicate malicious behaviour and to provide methods to solve potential risk. Modern NIDS system harness the power of intelligence feeds to detect threats, behavioural of the user, to provide protection against cyber-attacks. They include data from various sources like network logs to provide full-fledged security from the attacks.

In addition to traditional on-premises deployments, modern IDS offerings extend to cloud-based architectures and managed security services, offering scalability, agility, and ease of management. They seamlessly integrate with existing security infrastructure, such as firewalls, intrusion prevention systems (IPS) to orchestrate a unified defense strategy that fortifies overall cybersecurity posture.

4. **Cloud:** Network Intrusion detection have under gone a large transformation in which traditional ways of security measure is no longer adequate to protect against a distributed cyber thread so that the reason why there is wide spread adoption of cloud technologies to use cloud-native solution for scalable and flexible capabilities.

   By using cloud-based IDS technique we can scale and detect and possible security incident that can lead to potential threat and can also analyze a large amount of data.

Traditional Network Intrusion Detection:

Before the introduction of Machine Learning (ML) into Network Intrusion Detection Systems (NIDS), traditional methods were used which consist of rule-based approaches and signature-based detection systems. The several characteristics are:

Signature-Based Detection:

These are those system that have predefined rules and patterns which is used to identify the attackers. This system falls in front of newer ways of threat attack.

Rule-Based Systems:

In this rule-based system certain rules were either block or flagged of as they were indictive of malicious activity. There rigidity made them less adaptive with growing system.

Limited Adaptability:

Traditional IDS were less adaptive and time consuming to be used as newer signatures were always manually fed.

High False Positive Rates:

Rule-based system was prone to give a large false report which would make the system detective.

Inability to Detect Unknown Threats:

The traditional system was unable to react to newer threat and even struggled against previously detected threat.

Scalability Challenges:

As data consumption and network traffic increased the traditional system were unable to keep up with chancing trends.

Reactive Approach:

The traditional NIDS operated on a reactive basis, responding to known threats or deviations from predefined rules.

Reasons for NIDS:

Proactive Threat Detection: NIDS provide early warnings, enabling swift response and mitigation actions. It serves as a frontal defense which is used to find any anomaly like malware attack, denial-of-service attack, or any suspicious activity. It helps us to come with early warnings so that we can reduce its impact and take actions.

Regulatory Compliance: Compliance mandates the way organization implement system to defend against cyber-attacks. Standards like PCI, DSS, HIPAA and GDPR ensures that these rules are followed around all the organization to protect confidential data.

Data Protection: Data is the new gold and it play a very important role in todays life. NIDS provides protection of data and mitigates action to reduce the risk of data breach.

Incident Response Readiness: The provision of strategies to secure data is contingent upon the presence of incident response readiness. It facilitates teams' quick response to lessen the effects of security breaches.

Continuous Threat Monitoring: NIDS keep an eye out for new and developing threats by utilizing sophisticated detection algorithms and threat information feeds. NIDS improve its detection capabilities by proactively identifying emerging threats and modifying defenses to efficiently manage risks by incorporating threat intelligence into their operations.

To put it briefly, network integrity preservation, data protection, proactive threat detection, regulatory compliance, incident response preparation, and continuous threat monitoring are all included in the diverse function that NIDS play in cybersecurity. Organizations can strengthen their defenses and maintain resilience against the constantly changing threat landscape by implementing NIDS as part of a complete security plan.

The market for intrusion detection has been growing for a long time. Experts accept it to grow to a 7-billion-dollar industry by 2025 with 14 % growth per annum. There is a reason why all the big companies and trying to make their system full proof and are hiring experts who can deal with any problems may come at their way.
It has been a very important topic for academics and security of a nation and therefore has been studied a lot.

We choose Aws which has a lot of service in public, private or hybrid network. We also look for the blackened services that are provided by the country.

Market For NIDS:

The Network Intrusion Detection Systems (NIDS) market is witnessing substantial growth, spurred by escalating cyber threats and stringent regulatory requirements. Key factors contributing to the market's size and expansion include:

Rising Cyber Threats: The huge surge in cyberattacks, like malware, ransomware, and phishing, have made the organization to use robust security solutions like NIDS to fortify their defenses against evolving threats.

Regulatory Compliance: With the introduction of different compliance rules, it necessitates the implementation od NIDS to provide effective detection for propelling growth in the market.

Cloud Adoption: With the growing adoption of cloud-based technology, organization are looking for NIDS that have cloud compatible solution. Now organization are looking for scalable offering to provide comprehensive security.

Technological Advancements: Organizations are looking for creative NIDS solutions that make use of these technologies in order to stay ahead of increasingly sophisticated attacks, as machine learning and behavioral analytics, two recent developments in NIDS technology, improve threat detection capabilities.

Industry Adoption: NIDS finds applications across diverse industry verticals, including finance, healthcare, government, retail, and manufacturing. Each sector's unique security requirements drive demand for tailored NIDS solutions, contributing to market expansion.

Global Market Competition: The NIDS market is pigeon-holed by competition among numerous merchants offering a wide variety of solutions. Established cybersecurity firms and emerging startups vie for market share, fostering innovation and driving market growth.

OBJECTIVE

- The primary objective of this project is to design and implement a network intrusion detection system that harnesses the capabilities of machine learning for enhanced threat detection.

- This report is structured to provide a wide-ranging summary of the project, starting with various aspects of network intrusion detection, including data collection, feature extraction, model selection, training, and evaluation and machine learning.

- Subsequently, the methodology employed in designing and implementing the intrusion detection system is elucidated, followed by detailed results and discussions.

- Finally, conclusions are drawn, and avenues for future research are outlined.

Building a network intrusion detection system is the task at hand to identify abnormalities and network attacks based on: -

Classification using many variables: Normal activity, DOS, PROBE, R2L, or U2R.

LITERATURE SURVEY

"A survey of data mining and machine learning methods for cyber security intrusion detection." IEEE Communications Surveys & Tutorials 18.2 (2016): 1153-1176.

- This survey paper offers a detailed examination of machine learning (ML) and data mining (DM) techniques employed in cyber analytics to bolster invasion detection capabilities.

- The paper includes concise tutorial overviews of each ML/DM method under consideration. Through a combination of assessing citation counts and the relevance of emerging methodologies, the survey identifies and analyzes papers that represent each method. Additionally, it provides insights gleaned from the examination of various cyber datasets crucial for ML/DM applications in cybersecurity.

- Acknowledging the intricacy of ML/DM algorithms, the paper delves into the challenges associated with deploying these techniques in cybersecurity contexts. It also offers recommendations on the suitability of specific methods for particular scenarios, aiming to assist practitioners in making informed decisions regarding method selection.

Roesch, Martin. "Snort—lightweight intrusion detection for networks." USENIX Association, 1999.

- Snort is a useful tool for network security. Because of its cross-platform compatibility and lightweight design, it may be used to monitor a wide range of networks, including smaller TCP/IP configurations. Through the detection of unusual network activity and possible assaults, Snort gives administrators critical information they need to evaluate risks and determine the best course of action. Using a data-driven approach, administrators may make well-informed decisions to reduce risks and safeguard their networks

- Packet payload inspection is the main feature that distinguishes Snort from tcpdump. Snort is a packet decoder that decodes the application layer. It can be configured with rules to gather traffic containing particular data in its application layer. Because of this, Snort can identify a wide range of malicious behavior, such as buffer overflows, CGI scans, and any other data in the packet payload that may be identified by a distinct detection fingerprint.

Network Intrusion Detection Systems: Anomaly Detection via Machine Learning and Dimensionality Reduction

- Using the NSL-KDD dataset as a benchmark, the paper examines the efficacy of several machine learning methods for network intrusion detection. The study uses Principal Component Analysis (PCA) to reduce dimensionality and feature scaling as preprocessing methods. Information is preserved when the original features of the dataset are reduced to primary components.

- The following seven machine learning methods are assessed: Random Forest Classifier, Decision Tree Classifier, Linear Support Vector Classifier, Gaussian Naive Bayes, K-Neighbors Classifier, Random Forest Classifier, and a Random Forest version involving PCA. Evaluation metrics include recall, precision, and accuracy on tests and in training.

- K-Neighbors Classifier outperforms Logistic Regression in terms of training and test accuracy, exhibiting a competitive performance. Additionally performing are Gaussian NB, Linear SVC, Random Forest Classifier, and Decision Tree Classifier.

- According to the study, these machine learning algorithms are suitable for intrusion detection tasks; the most reliable performance is the K-Neighbors Classifier. PCA reduces recall marginally but simplifies computation without sacrificing considerable accuracy, suggesting a trade-off between accuracy and sensitivity to positive cases.

Deep Learning Methods for Cyber Security Survey

Cybersecurity is all about protecting your digital world—your computers, networks, and information—from bad actors and harmful activities online. It's like having locks and security systems for everything you do on the internet.

Attack Prevention: The goal of cybersecurity is to protect against a variety of cyberattacks, including ransomware, phishing schemes, malware infections, and more.

Ensuring Privacy: By limiting who has access to your data and making sure that your online actions are private and safe, it helps you keep your privacy online.

Network Security: To guard against breaches and prevent unwanted access, networks, such as Wi-Fi in homes and businesses, should have security measures in place. This is known as cybersecurity.

Using strong passwords, updating software, avoiding dubious emails or links, and utilizing security tools like firewalls and antivirus software are all part of maintaining excellent cybersecurity practices.

Mukkamala, S., et al. "Intrusion detection using ensemble of soft computing paradigms." Journal of Computer Applications 28.2 (2005):

- Paper explores intrusion detection using ensemble of intelligent paradigms: ANNs, SVMs, and MARS.
- Intrusion detection crucial for safeguarding critical infrastructure reliant on computer networks.
- Comparative study shows ensemble approach surpasses individual methods in classification accuracy.
- Various intelligent computing techniques used for building IDSs, including ANNs, SVMs, and data mining.
- SVMs found superior to ANNs in many aspects of intrusion detection.
- Emphasizes importance of accurate detection for maximal IDS performance.
- Experimental results highlight efficacy of ensemble techniques on DARPA intrusion data.
- Audit data analysis crucial post-attack for damage assessment and attack trace back.
- Intrusion detection methods evolved from rule-based approaches to neural networks and data mining.

Data Mining Approaches for Intrusion Detection

- To find recurring and practical patterns in system attributes for characterizing user and program behavior, apply data mining techniques.
- Summary of the frequent episodes' algorithm and association rules, two widely used data mining methods.
- Algorithms calculate record patterns inside and between audits, which are crucial for characterizing user or program behavior.
- Found patterns aid in feature selection and direct the audit data collection process.

- For effective learning and real-time detection, suggest an agent-based architecture for intrusion detection systems.
- Models that describe "normal" system behavior are defined, and real actions are compared for anomalies, in order to detect intrusions.
- Get rid of ad hoc and manual components from creating intrusion detection systems.
- Misuse detection matches and encodes incursion patterns, whereas anomaly detection locates typical usage patterns.
- Determine whether audit data is normal or abnormal using classification techniques.
- To facilitate feature selection, link analysis establishes relationships between database fields.
- A crucial tool for behavior profiling is sequence analysis, which simulates sequential patterns.
- The methodology is adaptable to many computing environments and is mechanical and universal.

Amin, Syed Muhammad, et al. "Deep learning for network intrusion detection: A survey." IEEE Communications Surveys 23.1 (2021): 202-231.

- Recent development of Intrusion Detection Systems (IDS) due to widespread security threats in computer networks.
- Focus on deep learning techniques for automatic intrusion detection and abnormal behavior identification.
- Review analyzes deep learning-based intrusion detection methods and various IDS.
- Evaluation of deep learning techniques using routine metrics like accuracy, precision, recall, detection rate, false alarm rate.
- Discussion on contests and solutions in network security and privacy.
- IoT network architecture involves device communication across network layers, utilizing sensors for real-time data collection.
- Challenge of facing variations in malicious software leading to network breaches.

- Evolution of complex cyber-attacks necessitates novel prevention and detection techniques.
- Importance of IDS in addressing diverse security threats in networks.
- Critical review of high-tech deep learning-based techniques for IDS.
- Conclusion addresses future challenges and potential solutions in network security.
- Related work explores existing techniques and results based on deep learning, including Auto-IF and WFEU-FFDNN for anomaly detection.
- Performance evaluation of proposed IDS systems on datasets like UNSW-NB15 and AWID, suitable for wired and wireless network applications.
- Proposal of Automated IDS using Recurrent Neural Networks (RNN) for fog networks.

An Examination of Artificial Intelligence Methods for Network Malicious Activity Recognition in Developing Technologies

- Increased cyberattacks target emerging technologies like Cloud, Fog, Edge computing, and IoT, threatening network security and economic information.
- Network anomaly detection systems (NADSs) crucial for monitoring and preventing potential threats and abnormal user behavior.
- Malicious incidents, including DoS and DDoS attacks, disrupt regular network traffic, causing significant consequences for users and organizations.
- Zero-day attack detection challenging due to lack of signature specifications.
- Conventional architecture of machine learning-based NADS comprises four main modules:
- Packet decoder: Takes in unprocessed network traffic packets and forwards pertinent data to the pre-processing module.
- Pre-processing: Creates the normalized feature vectors required for learning-based systems.
- Classifier system: Builds a model to discriminate malicious instances against normal ones.
- Detection and recognition: Identifies malicious instances and various types of abnormality, transmitting alerts for system administrator reaction.

Cyberattack detection with an IoT-based smart city's machine learning-based intrusion detection system

- Artificial intelligence (AI) is transforming the world, leading to the development of smart products.
- Smart cities incorporate AI and Internet of Things (IoTs) innovations for enhanced functionality.
- Despite the convenience of smart cities, security concerns hinder their progress.
- Intrusion Detection Systems (IDS) monitor network traffic and alert users to anomalies.
- Machine Learning-based IDS intelligently detects threats, assesses data packet legitimacy, and notifies users.
- Researchers apply various ML techniques to IDS to enhance detection accuracy.
- Relative examination of ML algorithms trained on UNSW-NB15 dataset includes ADA Boost, LSVM, Auto Encoder Classifier, QSVM, and Multi-Layer Perceptron.
- ADA Boost algorithm achieves outstanding accuracy of 98.3% in the results.

A Comprehensive Review of Artificial Immune System-Based Intrusion Detection Systems

- Intrusion Detection Systems (IDS) identify abnormalities and attacks compromising system or network confidentiality, integrity, and availability.
- IDS shares parallels with Human Immune Systems (HIS), which detect harmful pathogens in humans.
- Mechanisms inspired by HIS, such as Artificial Immune Systems (AIS), are utilized in IDS to detect malicious packets.
- AIS mimics HIS processes to identify and prevent harmful pathogens in networks.
- Focus is on distributed agent-based systems in AIS-based IDS.
- Commonly used algorithms in AIS-based IDS are discussed.

- Limitations of existing work and future directions in AIS-based IDS are explored.

A framework for satellite communications that uses machine learning and deep learning techniques to identify heterogeneous Internet traffic

- The Internet network system facilitates communication, transactions, and entertainment, comprising terrestrial and Satellite components.
- By providing Quality of Service (QoS) and making the most use of the resources at their disposal, satellite communication providers hope to increase customer satisfaction.
- By minimizing errors such as packet delays and information loss in satellite communications, based on error conditions and Internet traffic kinds (such as VoIP, streaming, and browsing), QoS can be improved.
- This work uses machine learning (ML) and deep learning (DL) techniques to find new methods for categorizing Internet traffic in order to improve quality of service (QoS).
- ML and DL solutions will integrate with a known Satellite Communication and QoS management architecture, assessing all necessary components.
- A rivalled Satellite Communication platform will generate a comprehensive set of Internet traffic for system development and testing.
- The classification system will handle various types of Internet communications, processing incoming traffic hierarchically for high classification performance.
- Cloud-emulated platform experiments validate the proposal and provide Satellite architecture deployment instruction

.

A Comparative Analysis of Network Intrusion Detection Systems' Anomaly Detection Schemes

- Intrusion detection involves techniques for identifying attacks on computers and network infrastructures.

- Irregularity detection is pivotal in intrusion detection, identifying deviations from normal behavior that may indicate attacks or faults.

- A thorough comparison of many anomaly detection techniques is carried out in this research.

- Evaluation of supervised and unsupervised anomaly detection techniques and their variants is conducted using actual network data and the DARPA 1998 data set including network connections.

- Evaluation is performed using standard techniques and specific metrics suited for detecting attacks involving numerous connections.

- The findings of the experiments indicate that some anomaly detection techniques perform well on spotting new intrusions in both actual network data and DARPA'98 data.

Existing Challenges and Proposed Solutions in Network Intrusion Detection Systems

Existing Challenges:

- Conventional intrusion detection systems face issues such as high false positives, limited detection of novel attacks, scalability constraints, configuration complexity, and performance overhead.

Proposed Solutions:

- To improve accuracy and adaptability, embrace AI and machine learning approaches, particularly deep learning.
- Use anomaly detection to minimize false positives and identify anomalous behaviors.
- For efficiency, use dimensionality reduction and feature selection.
- Use behavioral analysis to identify insider threats.
- To increase accuracy and scalability, use distributed and collaborative detection techniques.
- Make feedback and ongoing monitoring possible so that detection models may be updated.
- For proactive defense, integrate with threat intelligence streams.
- To expedite incident response procedures, automate response steps.

The purpose of these technologies is to expand the resilience, adaptability, and efficacy of intrusion detection systems against changing cyberthreats.

Tools used:

1 Python Language:

- Python is a general-purpose, high-level and it is quite popular these days.
- Extensive Data analysis and machine learning works can be done through it easily as compared to their languages such as C/C++, java, JavaScript etc.
- Python language is currently used by most technical organizations.

2 Jupyter Notebook:

- Jupyter Notebook is a latest interactive web-based ide for python programming and testing.
- It improves workflow and efficiency.

3 Libraries Used:

3.1 Matplotlib version 3.0.3:

- Matplotlib is a 2D plotting based library which is often used in data science projects.
- It can easily generate histograms, plots, bar charts, error charts and scatterplots etc.
- It also low code in nature as compared with other libraries.

3.2 Seaborn version 0.9.0:

- Seaborn is a library that is built in conjunction to the matplotlib library.
- It aims to make graphs and plots a central part of the data science project.

3.3 Sklearn version 0.20.3:

- Scikit-learn (Sklearn) is a comprehensive library used for machine learning processes.
- In itself it is a compilation of several machine learning modules which can be easily imported and applied to a project.

3.4 NumPy version 1.24.3:

- NumPy is also known as Numerical Python and it was created in the year 2005.
- It delivers operation on linear algebra, arrays, transform series and matrices.
- It is fast and efficient(runtime) as it is built on C++ platform.

3.5 Pandas version 2.0.1:

- Pandas is used for working with datasets and data frames.
- It was created in the year 2008 and since then reading cleaning data using pandas has become much popular.

Models Used:

1.Gaussian Naive Bayes

Gaussian Naïve Bayes (GNB) is a technique which uses Bayes theorem where we calculate the probability from the evidence overserved. It is different from the Bayes algorithm which follows feature independence, GNB uses features that follow a Gaussian (normal) distribution. The above line means all the data points that forms a class cluster around the mean according to the bell curve shape.

The working of GNB include:

Bayesian Inference: GNB uses the Bayer theorem to find the probability all the classes which matches the input feature. It combines the probability of finding the same class with correspondence of finding the same feature resulting in coming to subsequent probability for each class.

Assumption of Independence: GNB assumes that the features are independent of all class label and that we cannot find the value of one feature through the feature of other.

Gaussian Distribution: It assumes that all class follows Gaussian distribution if you have continuous feature.

Classification: GNB uses the Bayer theorem to calculate the posterior probability with observed feature of classes. It then selects the highest posterior probability from the above sample.

It often performs well in practice, particularly with small to medium-sized datasets. It may not be suitable for datasets where the independence assumption doesn't hold or where features don't follow a Gaussian distribution. It finds its application in medical diagnosis.

2. Logistic regression:

It is a statistical method used for solving binary classification problems. It is a technique that is used to predict the probability of the event by putting the data against logistic curve.

It includes:

Binary Classification: It is ideal for binary classification problem where the variable have only two outcomes like 1/0, yes/no etc.

Sigmoid Function: Instead of giving a straight answer like linear regression, logistic regression uses a special S-shaped curve called the sigmoid function. It is different from the linear regression wherein it uses a s-shaped curve i.e. sigmoid function to take input and divide it in range of 0 and 1. It switches like a flip-lop between two outcomes.

Linear Decision Boundary: It is about finding a plane in higher dimensions that best separates the two classes in your data. This line acts as a boundary for decision making.

Training: It is a mode that learns from the data to predict the minimum variation between the predicted probability and actual outcome. It uses the process of maximum likelihood estimation.

Decision Making: If you are using a threshold which is usually 0.5 then it decides which class it would assign the data point based on whether it predicted data above or below it.

It works best when the association between the input features and the outcome is roughly linear, and there are no significant interactions between features.

3. Neural Network (MLP):

A neural network mimics a human brain and forma crucial concept of AI and machine learning which can be used to find different patterns in the data.

Key components include:

Neurons: Basic units that process input data to produce an output with internal parameters.

Layers: In neural network, neurons are grouped into layers: Input receives data, hidden processes it, and output produces results.

Connections: Neurons are linked to connection with adjustable weights.

Activation Function: Each neuron applies a function to introduce complexity, like sigmoid or ReLU.

Feedforward Propagation: Input data in neural network moves through the network of layers.

Backpropagation: Error is traced back through the network to adjust weights and minimize prediction errors.

Training: Networks learn from labeled data, adjusting parameters iteratively.

4. **Linear Support Vector Machine (SVM)**

It is a classification algorithm that finds the best straight line or hyperplane in higher dimensions to separate different classes of different data points.

Here is a simpler explanation:

Binary Classification: It is used to find a line that divides the two data.
Margin: It is used to find the widest possible gap between different classes by finding the line with the nearest point.

Straight Line Decision Boundary: Othe4 methods use curve to separate classes but SVM uses a straight line.

Training: The SVM looks for the data points where it can be classified to get the maximum margin.

Kernel Trick: When data is not in the form that can be separated by a line then we can use Kernal trick to transform the data to higher dimension.

Regularization: SVM also has restriction for miscalculation to reduce the balance between making less mistakes and wider margin.

5**. Support Vector Classifier**

It is primary used to separate data in different groups by finding the best boundary line between them.

Here's how it works:

Sorting Data: SVDC sees the data and looks for ways to sort them in groups.

Important Points: they pay very close attention to line as it helps to place them the line at the right side.

Handling Tough Cases: Sometimes the data can be very complex and it would be complicated to separate them in groups therefore svc uses a kernel method to provide higher dimension to data.

Getting It Right: SVC tries to strike the right balance between fitting the data well and making good guesses for new data it has not seen before.

After training and fitting the dataset on the various machine learning algorithms stated above, we will implement a voting ensemble algorithm to maximize the accuracy of the model and find a final accuracy score for the model.

When multiple algorithms are combined to enhance the accuracy, it is called ensemble algorithm. Majority votes, bagging and boosting algorithm are all examples of ensemble algorithm. We will be using a voting ensemble algorithm. Voting is one of the simplest ways of combining the predictions from multiple machine learning algorithms.

Using your training dataset, it initially builds two or more independent models. Next, when your models are asked to forecast new data, you may use a Voting Classifier to wrap them around and average the predictions of the sub-models.

METHODOLOGY

1. Cleaning and preparing the dataset

One of the few publicly accessible data sets for network-based anomaly detection systems is the KDDCUP'99 data set, which is commonly utilized in these systems.

List of Features for the dataset: -

["duration","protocol_type","service","flag","src_bytes","dst_bytes","land", "wrong_fragment","urgent","hot","num_failed_logins","logged_in", "num_compromised","root_shell","su_attempted","num_root","num_file_creations", "num_shells","num_access_files","num_outbound_cmds","is_host_login", "is_guest_login","count","srv_count","serror_rate", "srv_serror_rate", "rerror_rate","srv_rerror_rate","same_srv_rate", "diff_srv_rate", "srv_diff_host_rate","dst_host_count","dst_host_srv_count","dst_host_same_srv_rate", "dst_host_diff_srv_rate","dst_host_same_src_port_rate", "dst_host_srv_diff_host_rate","dst_host_serror_rate","dst_host_srv_serror_rate", "dst_host_rerror_rate","dst_host_srv_rerror_rate","attack", "last_flag"]

Basic Features of each network packet: -

- **Duration:** How long the connection has been in place for
- **Protocol type:** The connection's protocol
- **Service:** Used destination network service
- **Flag:** Connection status: Normal or Error
- **Src_bytes:** The number of bytes of data sent from the source to the destination across a single connection
- **Dst_bytes:** The total bytes of data sent from the source to the destination via a single connection
- **Land:** If the source and destination IP addresses and port numbers match, this variable takes on the value 1, and if not, it takes on the value 0..
- **Wrong fragment:** The total amount of incorrect fragments in this relationship
- **Urgent:** The total number of urgent packets for this connection. When a packet has the urgent bit set, it is deemed urgent.

Attack Class: Attack Type

1. DoS: Back, Land, Neptune, Pod, Smurf, Teardrop, Apache2, Udpstorm, Processtable, Worm (10)

2. Probe: Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint (6)

3. R2L: Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named (16)

4. U2R: Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps (7)

2. Basic Exploratory Analysis

We will distribute the dataset on the following basis: -

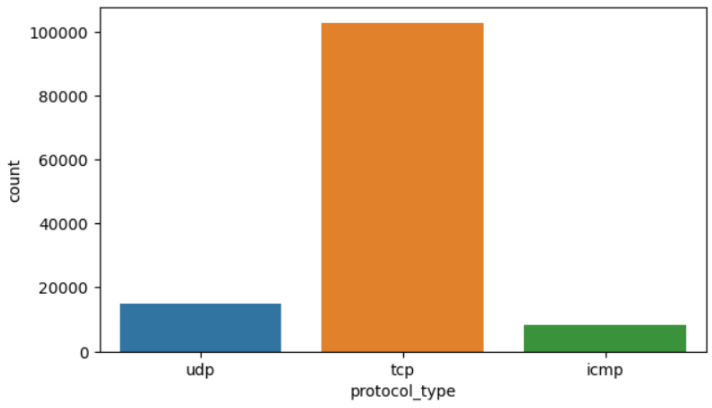
- Protocol distribution
- Service distribution
- Flag distribution
- Attack distribution
- Attack class distribution

Protocol Distribution-

Basic Exploratory Analysis

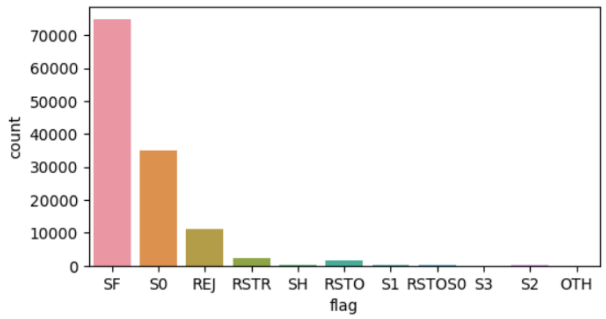
protocol distribution service distribution flag distribution attack distribution attack class distribution

```
In [12]: plt.figure(figsize=(7,4))
         sns.countplot(x="protocol_type",data = train)
         plt.show()
```



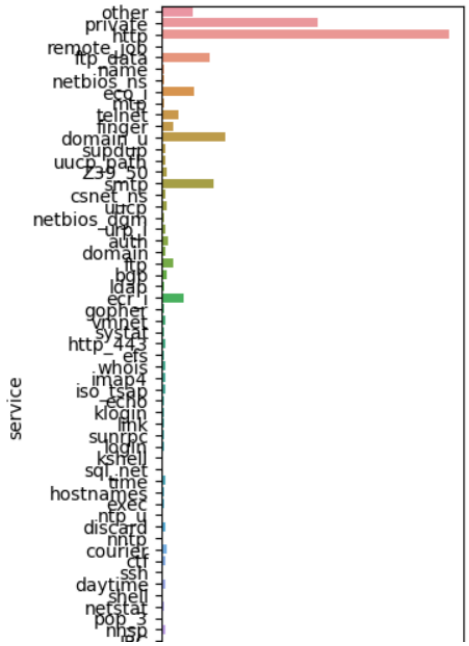
Flag Distribution-

```
In [14]: plt.figure(figsize=(6,3))
         sns.countplot(x="flag", data=train)
         plt.show()
```



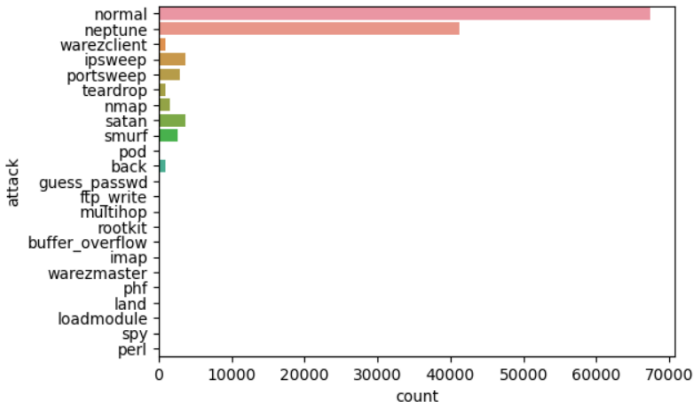
Service Distribution-

```
In [93]: plt.figure(figsize=(3,8))
         sns.countplot(y="service", data=train)
         plt.show()
```



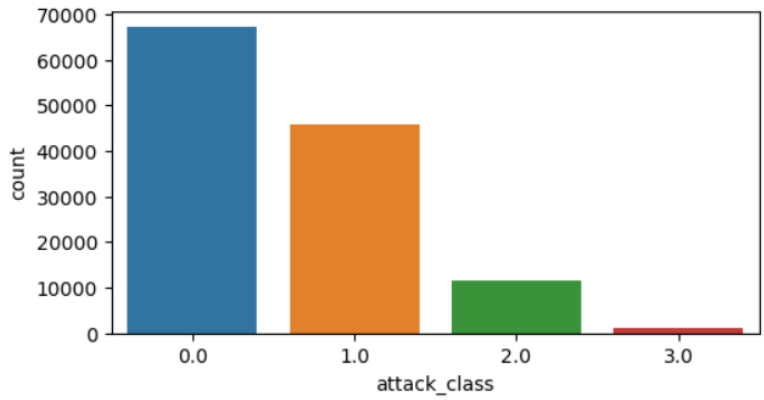
Attack distribution-

```
In [15]: plt.figure(figsize=(6,4))
         sns.countplot(y="attack", data=train)
         plt.show()
```



Attack class distribution-

```
In [16]: plt.figure(figsize=(6,3))
         sns.countplot(x="attack_class", data=train)
         plt.show()
```



3. Variable Reduction using Select K-Best technique

Variable reduction, also known as feature selection or dimensionality reduction, is an essential stage in data analysis and machine learning preparation that aims to enhancing model performance, interpretability, and computational efficiency.

Variable Reduction is possible using different techniques based on

-low variance
-high missing values
-high correlations

Variable reduction involves selecting a subset of relevant features from the original set of variables. This process is crucial for mitigating the "curse of dimensionality," which states that having too many features might result in overfitting and higher processing complexity, and reduced model generalization.

Cases where the number of features is prohibitively high, dimensionality reduction techniques are employed to transform the dataset into a lower-dimensional space while preserving most of the essential information.

We will be using Select K-Best Technique for our model. The Select K-Best technique is a feature selection method commonly used in machine learning to select the top k most relevant features from a dataset. Select K-Best works by assigning a score to each feature in the dataset based on a predefined scoring function. The scoring function evaluates the statistical relationship between each feature and the target variable.
Common scoring functions include chi-squared for categorical targets and ANOVA F-value for numerical targets.
After computing scores for each feature, Select K-Best selects the top k features with the highest scores. The value of k is determined by the user and depends on factors such as the desired model complexity, computational resources, and the nature of the dataset.

4. Training the model

- We will create a new feature “attack_class” to add to the dataset. Attack class feature will be the target. it consists of 5 categories which will be predicted using multinomial classification. 0 means normal 1 means DOS 2 means Probe 3 means R2L 4 means U2R.
- Then we will train the model on different machine learning algorithms: -
    1. Logistic Regression (LR)
    2. Linear SVC (SVM)
    3. Gaussian Naïve Bayes (GNB)
    4. SVC
    5. Neural Network: Multilayer Perceptron (MLP

RESULT

The accuracy scores of the different machine learning algorithms when applied on the dataset are as following: -

- Logistic Regression (LR)

```
In [65]: from sklearn.linear_model import LogisticRegression
         lr_clf = LogisticRegression(random_state=0, solver='lbfgs',multi_class='multinomial', max_iter = 10000).fit(X_train, Y_train)

In [66]: y_pred=lr_clf.predict(X_test)
         y_pred
Out[66]: array([1., 0., 2., ..., 1., 0., 2.])

In [67]: lr_score = lr_clf.score(X_test,Y_test)

In [68]: scores_list["LR"] = lr_score

In [69]: accuracy_score(Y_test, y_pred)
Out[69]: 0.8392849221487824
```

Accuracy score -83.92

- Linear SVC (SVM)

```
In [76]: from sklearn.svm import LinearSVC
         svm_clf = LinearSVC(random_state=0, tol=1e-5, dual= False)
         svm_clf.fit(X_train, Y_train)
Out[76]: LinearSVC
         LinearSVC(dual=False, random_state=0, tol=1e-05)

In [77]: y_pred=svm_clf.predict(X_test)
         y_pred
Out[77]: array([1., 0., 2., ..., 1., 0., 2.])

In [78]: from sklearn.metrics import accuracy_score
         svm_score = svm_clf.score(X_test,Y_test)

In [79]: scores_list["SVM"] = svm_score

In [80]: accuracy_score(Y_test, y_pred)
Out[80]: 0.8400833961761967
```

Accuracy score –84.00

- Gaussian Naïve Bayes (GNB)

```
In [86]: from sklearn.naive_bayes import GaussianNB

In [87]: gnb_clf = GaussianNB()
         gnb_clf.fit(X_train,Y_train)

Out[87]: GaussianNB()

In [88]: y_pred=gnb_clf.predict(X_test)
         y_pred

Out[88]: array([1., 0., 2., ..., 1., 0., 1.])

In [89]: accuracy_score(Y_test, y_pred)

Out[89]: 0.783968415916249
```

Accuracy score –78.39

- SVC

```
In [70]: from sklearn.svm import SVC
         from sklearn.pipeline import make_pipeline

         model = SVC(kernel='rbf', class_weight='balanced',gamma='scale')

In [71]: model.fit(X_train, Y_train)

Out[71]: SVC(class_weight='balanced')

In [72]: y_pred=model.predict(X_test)
         y_pred

Out[72]: array([1., 0., 2., ..., 2., 0., 2.])

In [73]: svc_score = model.score(X_test,Y_test)

In [74]: scores_list["SVC"] = svc_score

In [75]: accuracy_score(Y_test, y_pred)

Out[75]: 0.713525262831034
```

Accuracy score –71.35

- MLP

```
In [81]: from sklearn.preprocessing import StandardScaler
         from sklearn.neural_network import MLPClassifier
         scaler = StandardScaler()
         scaler.fit(X_train)

Out[81]: StandardScaler()

In [82]: train_X = scaler.transform(X_train)
         test_X = scaler.transform(X_test)

In [83]: mlp = MLPClassifier(hidden_layer_sizes=(30,30,30))
         mlp.fit(train_X,Y_train)

Out[83]: MLPClassifier(hidden_layer_sizes=(30, 30, 30))

In [84]: mlp_score = mlp.score(test_X,Y_test)
         scores_list["MLP"] = mlp_score

In [85]: accuracy_score(Y_test, y_pred)

Out[85]: 0.8400833961761967
```

Accuracy score –84.00

Now, using the voting ensemble algorithm we can also find a final accuracy score: -

```
Out[58]: VotingClassifier(estimators=[('svm', LinearSVC()), ('svc', SVC()),
                                      ('logist', LogisticRegression()),
                                      ('mlp', MLPClassifier()),
                                      ('gnb', GaussianNB())])

In [59]: y_pred=ensemble.predict(X_test)
         y_pred

Out[59]: array([1., 0., 2., ..., 1., 0., 2.])

In [60]: accuracy_score( Y_test, y_pred )

Out[60]: 0.8379984917712816
```

Accuracy score –83.79

FUTURE SCOPE

- A growing field of research in artificial intelligence and machine learning examines the application of various classifier techniques to intrusion detection systems. It has drawn the interest of scholars for a considerable amount of time.

- This work demonstrates the great utility of the KDD dataset in testing various classifiers. In order to generate just experiments and totally randomized independent test data, the study focuses on the KDD preprocess step.

- Among the classification methods (LR, SVM, SVC, MLP, and Gaussian Bayes Network), the linear SVM classifier has the greatest accuracy rate for recognizing and categorizing all KDD dataset attack types (DOS, R2L, U2R, and PROBE).

- The 41 attributes in the KDD dataset have all been recorded; however, additional classifiers and feature selection will be evaluated in further work to determine which features are most crucial.

- A crucial component of system performance during the training phase is the removal of features that are unnecessary and redundant. In future research, feature selection will be considered in a significant way when developing classification algorithms.

- There are numerous algorithms available for feature selection. The optimal feature selection algorithm should be used, since this will help with classification techniques and give the feature selection stage in intrusion detection more weight.

- Hybrid or ensemble classifiers can be used in performance measurement in place of baseline or single classifiers.

In conclusion, this research aims to use machine learning for intrusion detection in order to further the field of network security. By means of analytical and empirical evaluation, we prove the effectiveness and promise of ML-based methods in enhancing the robustness of contemporary network defenses.

CONCLUSION

To sum up, the incorporation of machine learning (ML) and artificial intelligence (AI) methodologies into Network Intrusion Detection Systems (NIDS) signifies a noteworthy progression in the field of cybersecurity. NIDS can now more effectively and precisely analyze large volumes of network data thanks to AI and ML, which improves threat detection capabilities.

Enhanced detection accuracy, less false positives, adaptability to changing threats, and higher operational efficiency through automation are the main advantages of utilizing AI and ML in NIDS. Using AI, NIDS can spot irregularities and intricate patterns that conventional rule-based systems could miss but that point to possible security breaches.

However, careful consideration of several criteria, such as data quality, model training, scalability, and continuing monitoring for model performance and effectiveness, is necessary for the successful implementation of AI-driven NIDS. In order to preserve confidence and support human decision-making in reaction to risks that have been discovered, organizations also need to address issues pertaining to the interpretability and transparency of AI-driven detections.

Future work in AI and ML research and development will improve NIDS capabilities even more, allowing for proactive threat detection and response to protect networks and digital assets from ever-evolving cyberthreats. AI-powered NIDS will be essential for bolstering overall cybersecurity posture and reducing risks in today's interconnected digital ecosystem as cyberattacks become more complex.

REFERENCES

- Buczak, Anna L., and Erhan Guven. "A survey of data mining and machine learning methods for cyber security intrusion detection." IEEE Communications Surveys & Tutorials 18.2 (2016): 1153-1176.

- Axelsson, Stefan. "Intrusion detection systems: A survey and taxonomy." Technical report, Chalmers University of Technology, 2000.

- Roesch, Martin. "Snort—lightweight intrusion detection for networks." USENIX Association, 1999.

- Mukkamala, S., et al. "Intrusion detection using ensemble of soft computing paradigms." Journal of Network and Computer Applications 28.2 (2005): 167-182.

- Lee, Wenke, and Salvatore J. Stolfo. "Data mining approaches for intrusion detection." USENIX Security Symposium 1998.

- Amin, Syed Muhammad, et al. "Deep learning for network intrusion detection: A survey." IEEE Communications Surveys & Tutorials 23.1 (2021): 202-231.

- Mohaisen, Aziz, and Omar Alrawi. "A survey on network anomaly detection using machine learning." Journal of Network and Computer Applications 145 (2020): 102447.

- Garcia-Teodoro, Pedro, et al. "Anomaly-based network intrusion detection: Techniques, systems and challenges." Computers & Security 28.1-2 (2009): 18-28.

- Patcha, Animesh, and Jung-Min Park. "An overview of anomaly detection techniques: Existing solutions and latest technological trends." Computer Networks 51.12 (2007): 3448-3470.

- Alazab, Mamoun, et al. "An efficient model for intrusion detection based on artificial immune system and fuzzy clustering." Computers & Security 30.5 (2011): 331-341.

- Zanero, Stefano, and Matteo Bocchi. "Internet traffic classification using machine learning." ACM SIGCOMM Computer Communication Review 37.4 (2007): 11-22.

- Lazarevic, Aleksandar, et al. "A comparative study of anomaly detection schemes in network intrusion detection." Proceedings of the Third SIAM International Conference on Data Mining. 2003.

- Elhag, Samir Mohamed, and Zeyar Aung. "Hybrid intrusion detection with ensemble of feature selection techniques and machine learning algorithms." IEEE Access 7 (2019): 153931-153945.

- Zhao, Jing, et al. "Survey of deep learning-based intrusion detection approaches." IEEE

Access 8 (2020): 17110-17125.

- Das, Dipankar, et al. "A survey of deep learning in cyber security." Information 10.10 (2019): 307.

- KD99 CUP DATASET (The UCI KDD Archive Information and Computer Science University of California, Irvine CA 92697-3425 October 28, 1999)

Report final main content.Plag Report

ORIGINALITY REPORT

8%	6%	4%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Georgia State University Student Paper	1%
2	ia801805.us.archive.org Internet Source	1%
3	jostmed.futminna.edu.ng Internet Source	1%
4	Submitted to Anna University Student Paper	<1%
5	Amjad Rehman Khan, Muhammad Kashif, Rutvij H. Jhaveri, Roshani Raut, Tanzila Saba, Saeed Ali Bahaj. "Deep Learning for Intrusion Detection and Security of Internet of Things (IoT): Current Analysis, Challenges, and Possible Solutions", Security and Communication Networks, 2022 Publication	<1%
6	vdoc.pub Internet Source	<1%
7	Mahdi Rabbani, Yongli Wang, Reza Khoshkangini, Hamed Jelodar, Ruxin Zhao,	<1%

	Sajjad Bagheri Baba Ahmadi, Seyedvalyallah Ayobi. "A Review on Machine Learning Approaches for Network Malicious Behavior Detection in Emerging Technologies", Entropy, 2021 Publication	
8	github.com Internet Source	<1 %
9	link.springer.com Internet Source	<1 %
10	turcomat.org Internet Source	<1 %
11	Submitted to New York Institute of Technology Student Paper	<1 %
12	eprints.qut.edu.au Internet Source	<1 %
13	kuey.net Internet Source	<1 %
14	publications.eai.eu Internet Source	<1 %
15	www.slideshare.net Internet Source	<1 %
16	Submitted to Manipal University Student Paper	<1 %

WWW.IJRPR.COM

International Journal of Research Publication and Reviews

(Open Access, Peer Reviewed, International Journal)

(A+ Grade, Impact Factor 5.536)

ISSN 2582-7421 ***Sr. No: IJRPR*** 116421-1

Certificate of Acceptance & Publication

This certificate is awarded to "Akshay Dawar", and certifies the acceptance for publication of research paper entitled "NETWORK INTRUSION DETECTION SYSTEM USING ML" in “International Journal of Research Publication and Reviews”, Volume 5, Issue 5 .

Signed Aashish Agarwal ***Date*** 13-05-2024

Editor-in-Chief
International Journal of Research Publication and Reviews

WWW.IJRPR.COM

International Journal of Research Publication and Reviews

(Open Access, Peer Reviewed, International Journal)

(A+ Grade, Impact Factor 5.536)

ISSN 2582-7421 ***Sr. No: IJRPR*** 116421-2

Certificate of Acceptance & Publication

This certificate is awarded to "RVS PRANAV ", and certifies the acceptance for publication of research paper entitled "NETWORK INTRUSION DETECTION SYSTEM USING ML" in “International Journal of Research Publication and Reviews”, Volume 5, Issue 5 .

Signed Aashish Agarwal ***Date*** 13-05-2024

Editor-in-Chief
International Journal of Research Publication and Reviews

WWW.IJRPR.COM

International Journal of Research Publication and Reviews

(Open Access, Peer Reviewed, International Journal)

(A+ Grade, Impact Factor 5.536)

ISSN 2582-7421 ***Sr. No: IJRPR*** 116421-3

Certificate of Acceptance & Publication

This certificate is awarded to "Sohen Anil Mondal", and certifies the acceptance for publication of research paper entitled "NETWORK INTRUSION DETECTION SYSTEM USING ML" in “International Journal of Research Publication and Reviews”, Volume 5, Issue 5 .

Signed Aashish Agarwal ***Date*** 13-05-2024

Editor-in-Chief
International Journal of Research Publication and Reviews

WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

NETWORK INTRUSION DETECTION SYSTEM USING ML

***Akshay Dawar*[*1],RVS PRANAV*[*2], Sohen Anil Mondal*[*3]**

[*1] Department of Computer Science Engineering, SRM Institute of Science and Technology, Modinagar, Uttar Pradesh, India
[*2] Department of Computer Science Engineering, SRM Institute of Science and Technology, Modinagar, Uttar Pradesh, India
[*3] Department of Computer Science Engineering, SRM Institute of Science and Technology, Modinagar, Uttar Pradesh, India

ABSTRACT:

Among the most important issues facing modern society is network security. The weaknesses in network security have grown in importance over the last ten years due to the internet's rapid expansion and widespread use. To improve accuracy and efficiency in identifying possible security breaches, this study suggests a Network Intrusion Detection System (NIDS) that makes use of Machine Learning (ML) capabilities. The suggested NIDS seeks to evaluate network traffic patterns and spot unusual behaviors suggestive of cyber threats by utilizing a variety of machine learning approaches, including ensemble methods, supervised learning, and unsupervised learning. Additionally, using carefully labeled datasets, we will train the algorithm to identify patterns linked to both benign and malevolent network activity. This study shows how useful the Knowledge Discovery and Data Mining (KDD) dataset is for testing and evaluating different machine learning techniques. It focuses mostly on the KDD preparation step to provide a credible and fair experimental data set

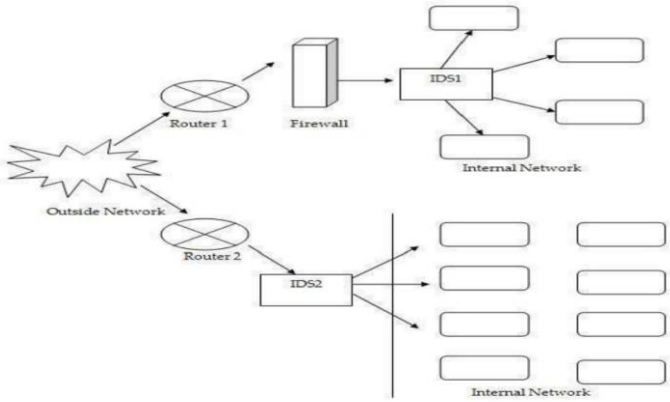
Keywords: Machine Learning, cyber threats, supervised learning, ensemble methods.

INTRODUCTION

In this modern era protection of individual property has become very important. The new ways through which cyber-attacks happen have become very concerning to the authorities. In times like this Machine Learning and NIDS (Network intrusion detection system has come as handy for solving our various problems. The old or traditional methods of tracking intrusion have long been left behind by the new ways how cyber have become prevalent in these times. Therefore, ML-based intrusion detection systems provide a standard way of using algorithms to detect patterns autonomously and detect oddity from traffic networks, providing more accurate and dynamic threat detection. As the rule-based systems are vulnerable to huge amounts of false positive rates, limited scalability, thereby losing their efficacy in detecting intrusions. We would define the underlying rules and principles that govern the many principles of ML driven models in cybersecurity. By deep diving into the details of ML algorithms, feature selection strategies and preprocessing data techniques we provide cybersecurity researchers with tools needed to use ML-based intrusion detection systems. Moreover, we can employ a variety of models rather than just a few to determine the best course of action when dealing with data silos issues. We go over every framework available for successfully refining various machine learning algorithms. We also go through methods for securing our data.

Signature-based detection, Anomaly-based detection, Protocol analysis, Attack categorization, correlations,
False positive,False negative, Packet header analysis, Machine learning algorithms, variance, bias.

1.1. ***Network Structure***



PROCEDURES AND METHODOLOGY

Cleaning and preparing the dataset

This data is KDDCUP'99 dataset, which is widely used as one of the few publicly available datasets for network-based anomaly detection systems.

Basic Exploratory Analysis

We distribute the dataset and explore it based on different parameters: -

1. Protocol Distribution
2. Service Distribution
3. Flag Distribution
4. Attack Distribution
5. Attack Class Distribution

Attack Class is a new parameter created by us, and not previously given in the dataset. It will help in grouping different types of attacks into a class. Attack class features will be the target. It consists of 5 categories which will be predicted using multinomial classification. 0 means normal 1 means DOS 2 means Probe 3 means R2L 4 means U2R.

Variable Reduction

Variable reduction, also known as feature selection or dimensionality reduction, is a critical preprocessing step in machine learning and data analysis aimed at enhancing model performance, interpretability, and computational efficiency.

Variable Reduction is possible using different techniques based on
-low variance
-high missing values
-high correlations

Variable reduction involves selecting a subset of relevant features from the original set of variables. This process is crucial for mitigating the curse of dimensionality, where an excessive number of features can lead to overfitting, increased computational complexity, and reduced model generalization.

Cases where the number of features is prohibitively high, dimensionality reduction techniques are employed to transform the dataset into a lower-dimensional space while preserving most of the essential information.

We will be using Select K-Best Technique for our model. The Select K-Best technique is a feature selection method commonly used in machine learning to select the top k most relevant features from a dataset. Select K-Best works by assigning a score to each feature in the dataset based on a predefined scoring function. The scoring function evaluates the statistical relationship between each feature and the target variable.

Common scoring functions include chi-squared for categorical targets and ANOVA F-value for numerical targets. After computing scores for each feature, Select K-Best selects the top k features with the highest scores. The value of k is determined by the user and depends on factors such as the

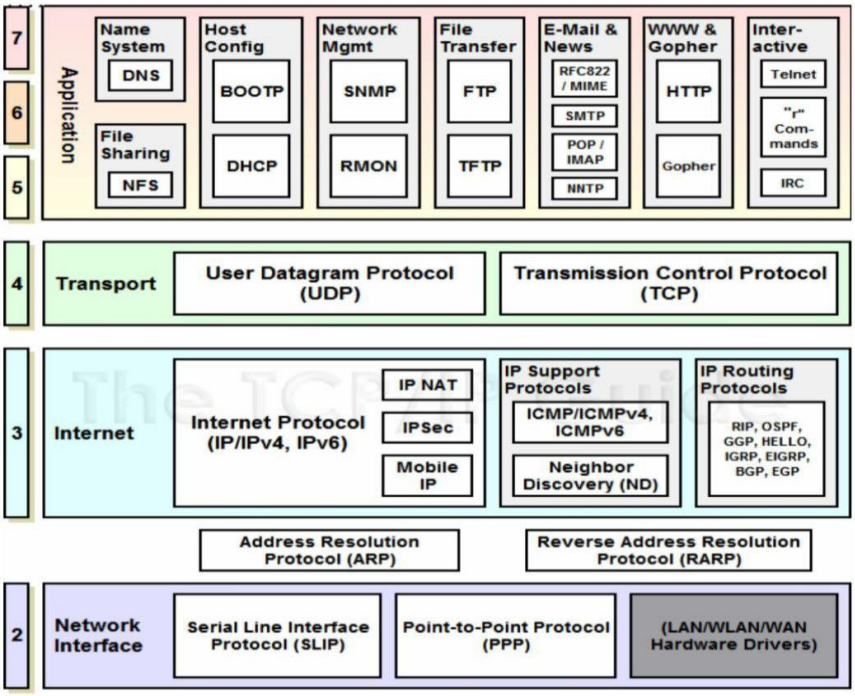
desired model complexity, computational resources, and the nature of the dataset.

Training and testing the model

We will train the model on 5 different machine learning algorithms: -

1. Logistic Regression
2. Gaussian Naïve Bayes
3. Linear SVC
4. Neural Network (Multilayer Perceptron)
5. Support Vector Classifier

TCP/IP HEADER STACK



ANALYSIS AND IMPLEMENTATION

In this section, we will be describing the modeling techniques used in our research.We will be using all supervised algorithms in this research as we are working with clearly labeled data.

Supervised machine learning uses labeled data to generate a function that maps an input to an output. The function is constructed from labeled training data.One of the main advantages of supervised learning is to use previous experiences to produce outputs. In addition, previous results can be used to improve the algorithm by optimizing the performance criteria to reach a precise model.

Supervised learning is used to solve many computational problems. However, the model needs precise and good input during the training phase to produce good outputs. In addition, this training requires a lot of computation time.

We used 5 different machine learning algorithms: -

1. **Logistic Regression**

It is a statistical method used for solving binary classification problems. It is a technique that is used to predict the probability of the event by putting the data against a logistic curve. It includes:

Binary Classification: It is ideal for binary classification problems where the variable has only two outcomes like 1/0, yes/no etc.

Sigmoid Function: Instead of giving a straight answer like linear regression, logistic regression uses a special S-shaped curve called the sigmoid function. It is different from the linear regression wherein it uses a s-shaped curve i.e. sigmoid function to take input and divide it in range of 0 and 1. It switches like a flip-flop between two outcomes.

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Linear Decision Boundary: It is about finding a plane in higher dimensions that best separates the two classes in your data. This line acts as a boundary for decision making.

Training: It is a mode that learns from the data to predict the minimum variation between the predicted probability and actual outcome. It uses the process of maximum likelihood estimation.

Decision Making: If you are using a threshold which is usually 0.5 then it decides which class it would assign the data point based on whether it predicted data above or below it.

It works best when the relationship between the input features and the outcome is roughly linear, and there are no significant interactions between features.

2. Gaussian Naïve Bayes

Gaussian Naïve Bayes (GNB) is a technique which uses Bayes theorem where we calculate the probability from the evidence overserved.
It is different from the Bayes algorithm which follows feature independence, GNB uses features that follow a Gaussian (normal) distribution. The above line means all the data points that forms a class cluster around the mean according to the bell curve shape:

$P(H|U)=(P(U|H)*P(H))/P(U)$

GNB assumes that the features are independent of all class labels and that we cannot find the value of one feature through the feature of another.

3. Support Vector Classifier

The Support Vector Classifier (SVC) is a powerful supervised learning algorithm widely used for classification tasks in machine learning. At its core, SVC aims to find the optimal hyperplane that best separates different classes within a dataset by maximizing the margin between the hyperplane and the nearest data points, known as support vectors. This margin maximization contributes to the algorithm's ability to generalize well on unseen data.It is primarily used to separate data in different groups by finding the best boundary line between them. SVM achieves this by maximizing the margin, or the distance between the boundary line and the nearest data points from each group. SVC can handle both linear and non-linear classification tasks through the use of kernel functions, which map the input data into a higher-dimensional space where a linear separation can be achieved.

The model is trained by solving a constrained optimization problem, typically using quadratic programming techniques, to identify the hyperplane that minimizes classification errors while maximizing the margin. SVC offers several advantages, including robustness against overfitting when the regularization parameter C is appropriately tuned, effectiveness in high-dimensional spaces, and capability to handle non-linear decision boundaries. These characteristics make SVC a popular choice for various applications, ranging from text categorization and image classification to bioinformatics and beyond.

4. Neural Network

A neural network mimics a human brain and forma crucial concept of AI and machine learning which can be used to find different patterns in the data. Neurons are basic units that process input data to produce an output with internal parameters.

A multilayer perceptron is a type of feedforward neural network consisting of fully connected neurons with a nonlinear kind of activation function. It is widely used to distinguish data that is not linearly separable.MLPs have been widely used in various fields, including image recognition, natural language processing, and speech recognition, among others. These layers can be: -
Input layer: - The input layer consists of nodes or neurons that receive the initial input data. Each neuron represents a feature or dimension of the input data. The number of neurons in the input layer is determined by the dimensionality of the input data.
Hidden layer: - Between the input and output layers, there can be one or more layers of neurons. Each neuron in a hidden layer receives inputs from all neurons in the previous layer (either the input layer or another hidden layer) and produces an output that is passed to the next layer.
Outer layer: - This layer consists of neurons that produce the final output of the network. The number of neurons in the output layer depends on the

nature of the task. In binary classification, there may be either one or two neurons depending on the activation function and representing the probability of belonging to one class; while in multi-class classification tasks, there can be multiple neurons in the output layer.

5. Linear Support Vector Machine (SVM)

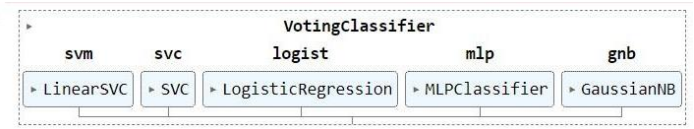
The Linear Support Vector Classifier (Linear SVC) is a variant of the Support Vector Machine (SVM) algorithm designed for linearly separable datasets. It operates by identifying the optimal hyperplane that best divides the classes in the input data space, aiming to maximize the margin between this hyperplane and the nearest data points. Unlike traditional SVC, Linear SVC specifically targets linearly separable data and does not rely on kernel functions for mapping to higher-dimensional spaces. Instead, it optimizes a linear decision boundary directly on the original feature space, making it computationally efficient and well-suited for large-scale datasets.

Linear SVC is trained by solving a convex optimization problem, typically using techniques like coordinate descent or stochastic gradient descent. This classifier is particularly effective when dealing with high-dimensional data or when the number of features exceeds the number of samples. Linear SVC offers simplicity, interpretability, and scalability, making it a preferred choice for applications requiring efficient binary classification with linear decision boundaries.

RESULTS AND DISCUSSIONS

This study underscores the substantial impact of machine learning on enhancing the efficacy of Intrusion Detection Systems (IDSs), emphasizing the pivotal role of dataset quality in determining IDS efficiency. Employing well-curated datasets is imperative, with many reviewed research papers leveraging labeled data to enhance model training. However, the burgeoning size of datasets poses a challenge, as conventional machine learning models may struggle to scale effectively. Consequently, researchers are increasingly turning to deep learning techniques, particularly Convolutional Neural Networks (CNNs), to pioneer innovative solutions. These approaches excel at extracting salient features from raw datasets, bolstering Network Intrusion Detection Systems (NIDS) against zero-day attacks. Moreover, NIDS must be regularly trained with real-time network data, although adopting these advanced methods comes at a cost, demanding more robust computing resources and prolonged processing times to train high-performing models.

Using the voting ensemble algorithm we can also find a final accuracy score: -




```
y_pred=ensemble.predict(X_test)
y_pred
```

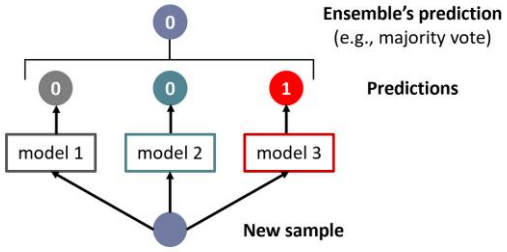
```
array([1., 0., 2., ..., 1., 0., 2.])
```

```
accuracy_score( Y_test, y_pred )
```

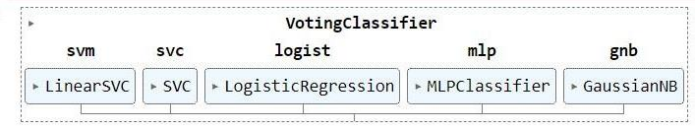
```
0.8379984917712816
```

Ensemble Model Accuracy score –83.79

Voting ensemble methodology:



Voting ensemble methods are often used to enhance classification accuracy and robustness. By combining the collective wisdom of various diverse classifying algorithms, they help in mitigating individual biases and variance.




```
y_pred=ensemble.predict(X_test)
y_pred
```

```
array([1., 0., 2., ..., 1., 0., 2.])
```

```
accuracy_score( Y_test, y_pred )
```

```
0.8379984917712816
```

CONCLUSIONS

Network Intrusion Detection Systems (NIDS) signifies a noteworthy progression in the field of cybersecurity. NIDS can now more effectively and precisely analyze large volumes of network data thanks to AI and ML, which improves threat detection capabilities.

Enhanced detection accuracy, less false positives, adaptability to changing threats, and higher operational efficiency through automation are the main advantages of utilizing AI and ML in NIDS. Using AI, NIDS can spot irregularities and intricate patterns that conventional rule-based systems could miss but that point to possible security breaches.

However, careful consideration of several criteria, such as data quality, model training, scalability, and continuing monitoring for model performance and effectiveness, is necessary for the successful implementation of AI-driven NIDS. In order to preserve confidence and support human decision-making in reaction to risks that have been discovered, organizations also need to address issues pertaining to the interpretability and transparency of AI-driven detections.

Future work in AI and ML research and development will improve NIDS capabilities even more, allowing for proactive threat detection and response to protect networks and digital assets from ever-evolving cyber threats. AI-powered NIDS will be essential for bolstering overall cybersecurity posture and reducing risks in today's digital ecosystem as cyberattacks become more complex.

REFERENCES

1. Buczak, Anna L., and Erhan Guven. "A survey of data mining and machine learning methods for cyber security intrusion detection." IEEE Communications Surveys & Tutorials 18.2 (2016): 1153-1176.
2. Axelsson, Stefan. "Intrusion detection systems: A survey and taxonomy." Technical report, Chalmers University of Technology, 2000.
3. Roesch, Martin. "Snort—lightweight intrusion detection for networks." USENIX Association, 1999.
4. Mukkamala, S., et al. "Intrusion detection using ensemble of soft computing paradigms." Journal of Network and Computer Applications 28.2 (2005): 167-182.
5. Lee, Wenke, and Salvatore J. Stolfo. "Data mining approaches for intrusion detection." USENIX Security Symposium 1998.
6. Amin, Syed Muhammad, et al. "Deep learning for network intrusion detection: A survey." IEEE Communications Surveys & Tutorials 23.1 (2021): 202-231.
7. Mohaisen, Aziz, and Omar Alrawi. "A survey on network anomaly detection using machine learning." Journal of Network and Computer Applications 145 (2020): 102447.
8. Garcia-Teodoro, Pedro, et al. "Anomaly-based network intrusion detection: Techniques, systems and challenges." Computers & Security 28.1-2 (2009): 18-28.
9. Patcha, Animesh, and Jung-Min Park. "An overview of anomaly detection techniques: Existing solutions and latest technological trends." Computer Networks 51.12 (2007): 3448-3470.
10. Alazab, Mamoun, et al. "An efficient model for intrusion detection based on artificial immune system and fuzzy clustering." Computers & Security 30.5 (2011): 331-341.
11. Zanero, Stefano, and Matteo Bocchi. "Internet traffic classification using machine learning." ACM SIGCOMM Computer Communication Review 37.4 (2007): 11-22.
12. Lazarevic, Aleksandar, et al. "A comparative study of anomaly detection schemes in network intrusion detection." Proceedings of the Third SIAM International Conference on Data Mining. 2003.
13. Zhao, Jing, et al. "Survey of deep learning-based intrusion detection approaches." IEEEAccess 8 (2020): 17110-17125.