

SDN-BASED INTELLIGENT INTRUSION DETECTION SYSTEM (IIDS) USING MACHINE LEARNING

Project ID: 24-25J-120

Project Proposal Report

Sriskandarajah J.P

BSc (Hons) Degree in Information Technology Specialized in Cyber
Security

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2024

SDN-BASED INTELLIGENT INTRUSION DETECTION SYSTEM (IIDS) USING MACHINE LEARNING

Project ID: 24-25J-120

Project Proposal Report

Sriskandarajah J.P -IT21261978

Supervised by Mr. Kanishka Prajeewa Yapa

Co-supervised by Mr. Tharaniyawarma.K

BSc (Hons) Degree in Information Technology Specialized in Cyber
Security

Department of Information Technology

Sri Lanka Institute of Information Technology Sri Lanka

August 2024

DECLARATION

We declare that this is our own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Additionally, I hereby grant to Sri Lanka Institute of Information Technology the nonexclusive right to reproduce and distribute my dissertation, in whole or in part, in future works (such as articles or books).

Name	Student ID	Signature
Sriskandarajah J.P	IT21261978	<i>D. Loannaf</i>

The supervisor should certify the proposal report with the following declaration

The above candidate is carrying out research for the undergraduate dissertation

K.P.S.
.....

Signature of the supervisor

29/08/2024
.....

Date

K. Tharun
.....

Signature of the co-supervisor

29/8/2024
.....

Date

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to those who contributed to the successful completion of this research project. First and foremost, I would like to thank my supervisors, Mr. Kanishka Prajeewa Yapa and Mr. Tharaniyawarma.K for their support, guidance, and encouragement throughout the research process.

I would also like to acknowledge the invaluable contributions of my teammates. Their collaboration, dedication, and creativity greatly enhanced our collective understanding of the research topic. Our teamwork and shared passion for the project were crucial to its successful completion.

Finally, I would like to express my deepest appreciation to my family, friends, and colleagues. Their constant support, patience, and understanding during this period provided me with the strength and motivation to persevere.

ABSTRACT

The increasing prevalence and sophistication of Distributed Denial of Service (DDoS) attacks pose significant threats to the security and stability of modern network infrastructures. This research proposes the development of a Machine Learning-based Intrusion Detection Engine designed specifically to identify and mitigate DDoS attacks within a Software-Defined Networking (SDN) environment. By integrating machine learning models directly with the SDN controller, the system enables dynamic, continuous monitoring and analysis of network traffic, facilitating real-time detection and response to DDoS attacks.

The proposed solution involves four key phases: data collection and preprocessing, machine learning model development, integration with the SDN controller, and model evaluation and optimization. Initially, large-scale network traffic data, encompassing both normal and DDoS attack scenarios, will be gathered and preprocessed to ensure it is suitable for training. Thereafter, appropriate machine learning algorithms will be selected and trained to distinguish between benign and malicious traffic. These models will then be integrated with the SDN controller to enable real-time traffic monitoring and automatic threat mitigation. Finally, the system's performance will be thoroughly evaluated and optimized to ensure its effectiveness in detecting various types of DDoS attacks, including both low-rate and high-rate variants.

(Keywords- SDN, IDS, ML, DDoS, Cyber Security)

TABLE OF CONTENTS

DECLARATION	i
ACKNOWLEDGEMENT	ii
ABSTRACT.....	iii
LIST OF TABLES	v
LIST OF FIGURES	v
LIST OF ABBREVIATIONS	v
1. INTRODUCTION.....	1
1.1. Background Literature	2
1.2. Research gap.....	3
1.3. Research problem	4
2. OBJECTIVES OF THE PROJECT.....	5
5.1 Main Objective.....	5
5.2 Specific Objectives	5
3. RESEARCH QUESTION	7
4. METHODOLOGY	8
4.1 System Diagram	8
4.2 Technologies	9
4.3 Requirements	9
4.4 Work Breakdown Structure	11
4.5 Gantt Chart.....	12
5. COMMERCIALIZATION ASPECTS OF THE PRODUCT	14
6. BUDGET AND BUDGET JUSTIFICATION.....	15
7. CONCLUSION	16
8. REFERENCES.....	17

LIST OF TABLES

Table 1:Abbreviation List	v
Table 2:Research Gap	4
Table 3: Budget.....	15

LIST OF FIGURES

Figure 1:System Diagram	8
Figure 2: WBS	11
Figure 3: Gantt Chart	12

LIST OF ABBREVIATIONS

Abbreviation	Description
SDN	Software Defined Networking
IDS	Intrusion Detection System
ML	Machine Learning
DDoS	Distributed Denial of Service
API	Application Programming Interface

Table 1:Abbreviation List

1. INTRODUCTION

The growing complexity and scale of modern networks have made traditional Intrusion Detection Systems (IDS) [1] increasingly inadequate in addressing advanced cybersecurity threats. As cyberattacks become more sophisticated and frequent, conventional IDS struggle to keep up due to their static nature and inability to quickly adapt to emerging threats. These systems are often overwhelmed by the massive volume of network traffic, leading to delayed or missed detection of malicious activities.

Software-Defined Networking (SDN) offers a transformative solution to these challenges by decoupling the control plane from the data plane, thereby centralizing network control and enabling more dynamic and flexible network management. Although the advantages SDN brings to network architecture, its potential in enhancing intrusion detection has yet to be fully realized. The static nature of traditional IDS within this dynamic framework limits their effectiveness in detecting and mitigating current cybersecurity threats [2], particularly Distributed Denial of Service (DDoS) attacks.

To address these limitations, this research focuses on developing an SDN-based Intelligent Intrusion Detection System (IIDS) that leverages machine learning to provide real-time, adaptive, and automated security management. By integrating machine learning models directly with the SDN controller, the proposed system aims to enable continuous monitoring and analysis of network traffic, allowing for the dynamic identification and mitigation of DDoS attacks as they occur.

1.1. Background Literature

The rise in Distributed Denial of Service (DDoS) attacks has led to significant challenges in maintaining the availability and performance of network services. Traditional intrusion detection systems (IDS) have long been the first line of defense against such attacks, typically relying on signature-based or anomaly-based detection methods [3]. Signature-based systems, match known patterns of malicious behavior against incoming network traffic. However, these systems often struggle to detect new or evolving attack vectors, as they rely on predefined signatures that must be continuously updated. Anomaly-based IDS, on the other hand, detect deviations from established baseline behaviors within network traffic. While this approach is more adaptable to new threats, it often suffers from high false-positive rates, as benign anomalies can be mistakenly flagged as attacks. Furthermore, both signature-based and anomaly-based IDS face scalability issues when challenged with the massive volumes of data generated in modern networks, particularly during large-scale DDoS attacks.

The introduction of Software-Defined Networking (SDN) has introduced a new approach in network management [4], allowing for centralized control and dynamic configuration of network resources. SDN separates the control plane from the data plane, enabling a more agile and responsive network environment. This architecture is particularly well-suited to the demands of real-time intrusion detection, as it allows for the rapid deployment and adaptation of security policies across the network.

Recent research has explored the integration of machine learning with IDS to enhance detection capabilities, especially in identifying complex patterns in network traffic that may indicate a DDoS attack. Machine learning models, such as decision trees [5], support vector machines [6], and neural networks [7], have shown promise in distinguishing between legitimate and malicious traffic. However, most studies have focused on offline training and detection, limiting the applicability of these models in real-time scenarios.

1.2. Research gap

While significant advancements have been made in the development of machine learning-based intrusion detection systems, several critical gaps remain, particularly in the context of DDoS detection within SDN environments:

1. **Real-Time Detection:** Many existing approaches focus on offline analysis, where machine learning models are trained and evaluated on historical data. This limits their effectiveness in real-time scenarios, where rapid detection and response are crucial to mitigating the impact of a DDoS attack. There is a need for systems that can dynamically analyze and act on live network traffic.
2. **Integration with SDN Controllers:** Although SDN provides a flexible and centralized platform for network management, the integration of machine learning-based IDS with SDN controllers is still in its early stages. Existing solutions often lack seamless integration, resulting in delays in threat detection and response. This gap highlights the need for a more cohesive and efficient integration of machine learning models with SDN controllers to enable real-time decision-making.
3. **Scalability and Generalization:** Many machine learning models used for DDoS detection are trained on limited datasets, raising concerns about their ability to generalize to new or unseen attack patterns. Furthermore, the scalability of these models in handling large-scale network traffic remains a challenge. Addressing this gap requires the development of models that can not only scale with the volume of network data but also adapt to new attack vectors without significant retraining.
4. **Adaptive Security Management:** Traditional IDS lack the ability to adapt to the constantly changing threat landscape, particularly in an SDN environment where network configurations can change dynamically. There is a need for an intelligent intrusion detection system that can adapt its detection strategies in real-time based on the evolving nature of network traffic and potential threats.

This research stands out in its integration of machine learning algorithms directly with the SDN controller, enabling real-time analysis and decision-making, which is a significant differentiate from traditional intrusion detection systems that rely on offline analysis. The proposed system not only improves detection speed but also enhances accuracy by dynamically adapting to evolving attack patterns in real-time. Additionally, the project focus on scalability ensures that the detection engine can handle large volumes of network traffic,

making it robust enough for deployment in modern, large-scale networks. By addressing key research gaps such as real-time detection, seamless integration with SDN controllers, scalability, and adaptive security management, this project offers a comprehensive solution to the challenges of securing dynamic network environments against sophisticated DDoS attacks.


























Research Gap Feature	Research 1 [8]	Research 2 [9]	Research 3 [10]	Research 4 [11]	Proposed Research Solution
Focused on real-time detection of DDoS attacks in SDN environments					
Integration of ML algorithms with SDN controllers for DDoS mitigation					
Scalability in handling large-scale network environments					
Automated response to detected threats					
Adaptive security Management					

Table 2: Research Gap

1.3. Research problem

The increasing frequency and sophistication of Distributed Denial of Service (DDoS) attacks present a significant challenge to maintaining the availability and reliability of network services. Traditional Intrusion Detection Systems (IDS) often fail to detect and mitigate these attacks in real time due to their static nature and inability to handle the vast volumes of traffic generated in modern networks. The need for a solution that can dynamically and continuously analyze network traffic in real-time, while effectively distinguishing between normal and malicious traffic, is critical. The problem is even more worsened by the lack of seamless integration between machine learning-based detection mechanisms and Software-Defined Networking (SDN) controllers, which could otherwise provide a flexible and centralized platform for automated and adaptive security management.

2. OBJECTIVES OF THE PROJECT

5.1 Main Objective

The primary goal of this research is to develop and integrate a Machine Learning-based Intrusion Detection Engine with a Software-Defined Networking (SDN) controller, enabling real-time detection and mitigation of Distributed Denial of Service (DDoS) attacks in dynamic network environments.

5.2 Specific Objectives

- Data collection and preprocessing
- Develop machine learning model
- Integration with SDN controller
- Evaluate and optimize the model

Specific

Collect and preprocess a dataset of at least 10000 network traffic instances, including both DDoS attack scenarios and normal traffic patterns.

Develop and train ML models using algorithms to differentiate between normal and DDoS traffic.

Integrate the trained machine learning model with an SDN controller to enable real-time network traffic monitoring and analysis.

Evaluate the model's performance using separate validation and test datasets, focusing on accuracy.

Measurable

Achieve a cleaned dataset with less than 5% missing or noisy data.

Achieve a detection accuracy of at least 95% and a false-positive rate of less than 3% on the training dataset.

Ensure detection and response latency is less than 1 second within the SDN environment.

Achieve a recall rate of at least 90%

Achievable

Use publicly available datasets like CICIDS2017 [12] and apply data cleaning and normalization techniques.

Utilize machine learning libraries and follow established model development processes.

Leverage existing SDN platforms and APIs for integration.

Conduct systematic testing and apply optimization techniques

Relevant

Prepares the dataset for training, ensuring accuracy in DDoS detection.

Forms the core of the intrusion detection engine, enhancing its ability to detect DDoS attacks.

Critical for enabling real-time DDoS detection and mitigation.

Ensures the model is effective in detecting different types of DDoS attacks, including low-rate and high-rate variants.

Time-bound

Complete the data collection within 4 weeks.

Complete model development and initial training within 6 weeks.

Complete the integration process within 4 weeks after model development.

Complete evaluation and optimization within 4 weeks after SDN integration.

3. RESEARCH QUESTION

How can a Machine Learning-based Intrusion Detection Engine be effectively developed and integrated with a Software-Defined Networking (SDN) controller to enable real-time detection and mitigation of Distributed Denial of Service (DDoS) attacks in dynamic network environments?

To effectively develop and integrate a Machine Learning-based Intrusion Detection Engine with a Software-Defined Networking (SDN) controller for real-time detection and mitigation of Distributed Denial of Service (DDoS) attacks, several key steps must be followed. First, the development process begins with the collection and preprocessing of large-scale network traffic data. This data is then used to train machine learning models. Selecting appropriate algorithms is crucial to ensure the model can accurately identify DDoS attacks, even in complex and dynamic network environments.

Once trained, the model must be integrated with the SDN controller, which centrally manages the network. This integration allows the machine learning model to analyze traffic in real-time, leveraging the SDN controller's ability to dynamically manage and reroute traffic. This setup ensures that when a potential DDoS attack is detected, the system can quickly respond by reconfiguring the network, isolating malicious traffic, or applying other mitigation strategies without human involvement.

Finally, continuous evaluation and optimization of the model are necessary to maintain its effectiveness. This includes testing against various DDoS attack types, adjusting parameters, and updating the model with new data to adapt to emerging threats. By following these steps, the integration of machine learning with SDN enables a robust, real-time defense mechanism against DDoS attacks in network environments.

4. METHODOLOGY

This methodology outlines the systematic approach taken to develop a Machine Learning-based Intrusion Detection Engine specifically designed to detect DDoS attacks within a network environment. The process involves several key steps, including data collection and preprocessing, feature extraction, model development, integration with the network infrastructure, and complete evaluation and optimization. Each task is carefully planned and executed to ensure the engine's effectiveness in identifying and mitigating DDoS threats in real-time. This methodology aims to create a robust, scalable solution that enhances network security by leveraging advanced machine learning techniques within a SDN framework.

4.1 System Diagram

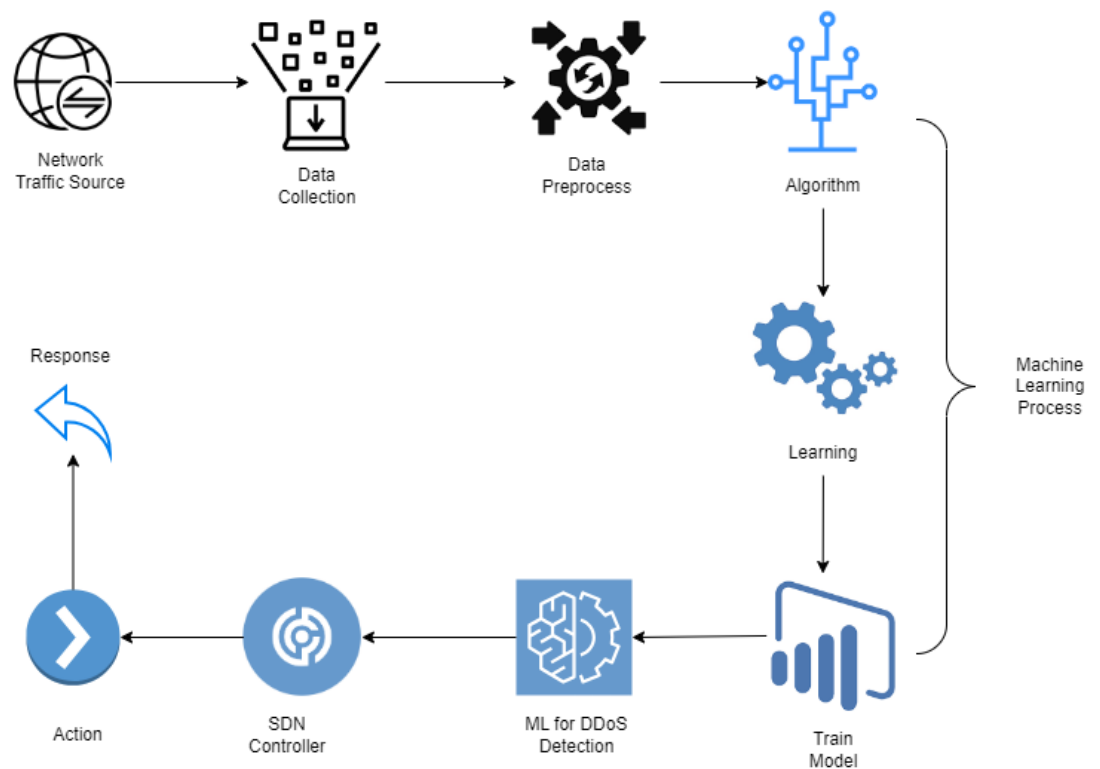


Figure 1: System Diagram

4.2 Technologies

Programming Language- Python [13]

Machine Learning Framework- TensorFlow [14]

SDN Controller- Open Daylight [15]

Network Traffic Tools- Wireshark [16]

Data Storage – MongoDB [17]

4.3 Requirements

Functional Requirements

These are the specific functions that the Machine Learning-based Intrusion Detection Engine must perform:

Real-Time DDoS Detection: The system must detect DDoS attacks in real-time by analyzing network traffic and identifying abnormal patterns of an attack.

Data Collection and Preprocessing: The system must collect network traffic data continuously from various sources within the network. It must preprocess the collected data by cleaning, normalizing, and extracting relevant features for input into the machine learning model.

Model Training and Update: The system must allow for the training of machine learning models using historical network traffic data that includes both normal and DDoS scenarios. The system must support periodic retraining and updating of the model to adapt to new and evolving attack patterns.

Integration with SDN Controller: The system must integrate seamlessly with the SDN controller to monitor and control network traffic. Upon detection of a DDoS attack, the system must communicate with the SDN controller to execute predefined mitigation strategies, such as rerouting traffic or blocking malicious sources.

Alert Generation and Logging: The system must generate alerts when a DDoS attack is detected and log all relevant information, including time, type of attack, and the affected

network segments. The system must provide detailed logs for post-incident analysis and auditing.

User Interface and Reporting: The system must provide a user-friendly interface that displays real-time analytics, alerts, and system status. It must generate regular reports summarizing the detected threats, system performance, and any executed mitigation actions.

Non-Functional Requirements

These are the quality attributes and constraints that the system must adhere to:

Performance: The system must process and analyze network traffic data with minimal latency to ensure real-time detection and response. The time from data input to detection should not exceed a few milliseconds. The system must be capable of handling large-scale network environments with high volumes of traffic without degradation in performance.

Scalability: The system must be scalable to accommodate growing network sizes and increased traffic volumes. It should support the addition of new nodes and the expansion of the network without requiring significant redesign or reconfiguration.

Reliability: The system must operate continuously with minimal downtime. It should be fault-tolerant and capable of recovering from failures without loss of critical data or functionality.

Security: The system itself must be secure against unauthorized access and tampering. Access controls, encryption, and other security measures must be implemented to protect sensitive data and system operations. The system must adhere to industry-standard security practices and comply with relevant regulations and standards.

Usability: The system must have a user-friendly interface that can be used by network administrators with varying levels of expertise. It should provide clear, actionable information that facilitates quick decision-making. User documentation and help features should be available to assist with system operation and troubleshooting.

Maintainability: The system must be designed for easy maintenance and updates. The codebase should be modular and well-documented to facilitate modifications, bug fixes, and upgrades. It should support automated deployment and continuous integration.

Adaptability: The system must be adaptable to different network environments and attack scenarios. It should allow for configuration adjustments to optimize performance under

specific network conditions. The system should be flexible enough to incorporate new machine learning models or algorithms as they become available.

Efficiency: The system must efficiently use computational resources, ensuring that it does not overwhelm the network or the processing power of the servers on which it runs. The system should optimize data processing and storage, minimizing the use of bandwidth and storage space while maintaining performance.

4.4 Work Breakdown Structure

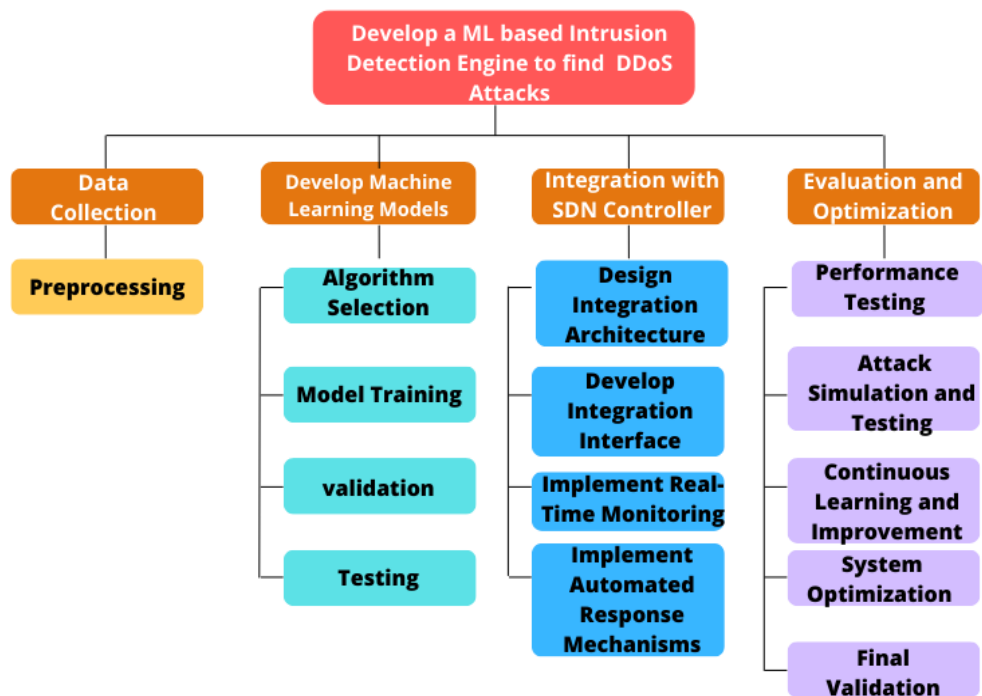


Figure 2: WBS

4.5 Gantt Chart

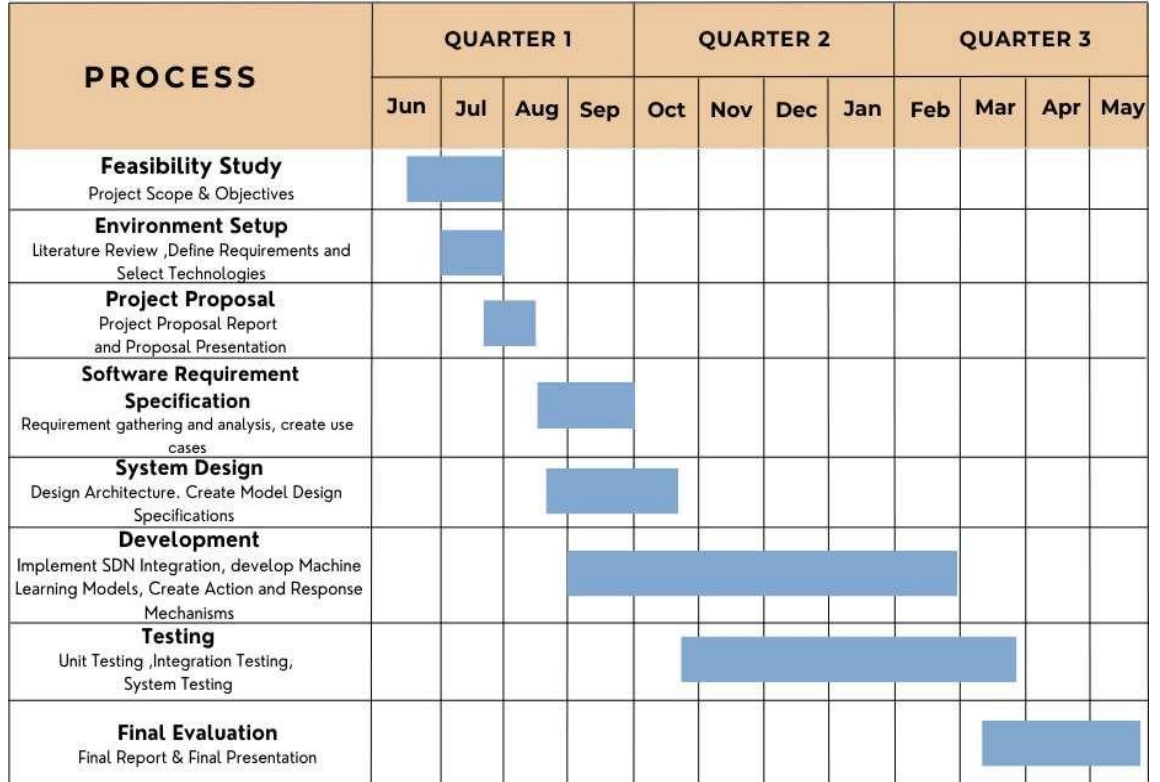


Figure 3: Gantt Chart

Tasks

Data Collection and Preprocessing: Collect large-scale network traffic data, including both normal and DDoS attack patterns. Preprocess the data by cleaning it (removing noise and handling missing values) and normalizing features to prepare it for model training.

Feature Extraction: Identify key features within the network traffic data that can effectively distinguish between normal traffic and DDoS attacks. Extract these features for use in training the machine learning model.

Develop Machine Learning Model: Train the machine learning model using algorithms to recognize DDoS attacks based on the extracted features.

Integration and Deployment of the Intrusion Detection Engine: Integrate the trained machine learning model into an Intrusion Detection Engine of the SDN controller. Deploy this engine within the network environment to monitor real-time traffic and detect potential DDoS attacks.

Evaluation and Optimization: Evaluate the model's performance using test datasets to assess its accuracy, recall, and precision [18]. Optimize the model to improve its detection capabilities and reduce false positives.

The project will result in a highly effective Machine Learning-based Intrusion Detection Engine capable of accurately detecting DDoS attacks in real-time. The integration of this engine within a network environment will significantly enhance the network's ability to respond to DDoS attacks, minimizing disruptions and maintaining service availability. The model's performance will be evaluated and optimized to ensure it meets the desired accuracy, recall, and precision metrics, thereby providing a robust solution for modern network security challenges.

5. COMMERCIALIZATION ASPECTS OF THE PRODUCT

The Machine Learning-based Intrusion Detection Engine, designed for detecting Distributed Denial of Service (DDoS) attacks in Software-Defined Networking (SDN) environments, presents several promising commercialization opportunities. Here are the key aspects:

Target Users

Companies with large and complex networks are the primary target customers, as they require advanced solutions to monitor and protect their infrastructure from DDoS attacks.

Cloud providers, who manage vast amounts of network traffic, are increasingly targeted by DDoS attacks. This product can be marketed as an essential tool for enhancing the security of cloud environments.

Telecom companies are key potential customers, needing robust solutions to maintain network reliability and security.

Marketing

With the increasing frequency and sophistication of cyberattacks, there is a rising demand for advanced cybersecurity solutions. Enterprises, particularly those in finance, healthcare, and critical infrastructure, are seeking robust systems to protect their networks. This Intrusion Detection Engine can address this need by providing real-time DDoS detection and mitigation, making it an attractive product for companies looking to enhance their cybersecurity measures.

6. BUDGET AND BUDGET JUSTIFICATION

Item	Description	Cost
Data Storage	Costs for storing datasets, user data, and model outputs	\$100/ month
Google Cloud Platform (GCP)	Compute, storage, and database services for deployment and model training.	\$300/month

Table 3: Budget

1.Data Storage- Necessary for securely storing large datasets used in training and testing machine learning models. Storage of user-related data, which may be required during the development and evaluation phases. Space needed to save trained model outputs, logs, and other materials generated during model development

2.Google Cloud Platform- Essential for deploying and running machine learning algorithms that require significant processing power. Necessary for handling large datasets and ensuring data is available for training and evaluation. Provides structured and reliable data management, ensuring efficient data retrieval during model development.

7. CONCLUSION

In this research, an Intrusion Detection Engine based on Machine Learning was developed with an aim of identifying Distributed Denial of Service (DDoS) attacks within a Software-Defined Networking (SDN) environment. By integrating machine learning algorithms directly with the SDN controller, the system was able to achieve real-time detection and mitigation of DDoS attacks. This approach not only enhanced the accuracy and responsiveness of the intrusion detection process but also leveraged the dynamic capabilities of SDN for adaptive security management.

The implementation demonstrated that the Machine Learning-based Intrusion Detection Engine is effective in distinguishing between normal and malicious traffic, particularly in handling large-scale, complex network environments. The integration with the SDN controller facilitated dynamic network monitoring and allowed for immediate, automated responses to detected threats, thereby improving the overall security posture of the network.

The research contributes to the field by addressing the limitations of traditional intrusion detection systems and providing a scalable solution that can adapt to evolving cybersecurity threats. Future work could explore the inclusion of advanced machine learning techniques, such as reinforcement learning, to further enhance the system's ability to learn from and adapt to new types of attacks.

8. REFERENCES

- [1] S. Isha., "Critical analysis of genetic algorithm based IDS and an approach for detecting intrusion in MANET using data mining techniques," pp. 37-41, 1 January 2012.
- [2] " Five Most Famous DDoS Attacks and Then Some," A10, 21 january 2022. [Online]. Available: <https://www.a10networks.com/blog/5-most-famous-ddos-attacks/>. [Accessed 8 August 2024].
- [3] P. Kamboj, M. C. Trivedi, V. K. Yadav and V. K. Singh, "Detection techniques of DDoS attacks: A survey," in *IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics (UPCON)*, Mathura, India, 2017.
- [4] A. Prajapati, A. Sakadasariya and J. Patel, "Software defined network: Future of networking," in *2nd International Conference on Inventive Systems and Control (ICISC)*, Coimbatore, India, 2018.
- [5] "geeksforgeeks," 17 May 2024. [Online]. Available: <https://www.geeksforgeeks.org/decision-tree/>. [Accessed 7 August 2024].
- [6] "geeksforgeeks," 4 July 2024. [Online]. Available: <https://www.geeksforgeeks.org/support-vector-machine-algorithm/>. [Accessed 7 August 2024].
- [7] "aws," [Online]. Available: <https://aws.amazon.com/what-is/neural-network/>. [Accessed 7 May 2024].
- [8] S. Scott-Hayward, G. O'Callaghan and S. Sezer, "Sdn Security: A Survey," in *IEEE*, Italy, 2014.
- [9] Wiem Tounsi, Helmi Rais, "A survey on technical threat intelligence in the age of sophisticated cyber attacks," in *computers and security*, France, science direct, 2017, pp. 212-233.
- [10] Tayfour, Omer Elsier and Muhammad Nadzir Marsono, "Collaborative Detection and Mitigation of Distributed Denial-of-Service Attacks on Software-Defined Network.," *Journal on spesial topics in mobile networks and applications*, 2020, pp. 1338-1347.

- [11] Sanjeetha Raja, Anita Kanavalli, Anshul Gupta, Ashutosh Pattanaik, Sashank Agarwal, "Real-time DDoS Detection and Mitigation in Software Defined Networks using Machine Learning Techniques," *International Journal of Computing*, no. research gate, 2022.
- [12] C. H. N, "Network Intrusion dataset," Kaggle, 2023. [Online]. Available: <https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset>. [Accessed 10 August 2024].
- [13] [Online]. Available: <https://www.python.org/>. [Accessed 10 August 2024].
- [14] [Online]. Available: <https://www.tensorflow.org/>. [Accessed 10 August 2024].
- [15] [Online]. Available: <https://www.opendaylight.org/>. [Accessed 10 August 2024].
- [16] [Online]. Available: <https://www.wireshark.org/>. [Accessed 10 August 2024].
- [17] [Online]. Available: <https://www.mongodb.com/>. [Accessed 10 August 2024].
- [18] Evidently AI Team, "Accuracy vs. precision vs. recall in machine learning: what's the difference?," Evidently AI, [Online]. Available: <https://www.evidentlyai.com/>. [Accessed 11 August 2024].