****

**Big Data Analysis for The Detection of Denial of Service & SQL Injections**

**By**

|  |  |
| --- | --- |
| **Student Name** | **Student ID** |
| Hajer Mohammed | 441808330 |
| Rajwah Mohammed | 439815569 |
| Sara Abdulkarim | 442809521 |

**Supervied** **by**

Dr. Suhail

Submitted to King Khalid University in fulfilment of the requirements for the B.Sc. Degree in Information Systems

**Academic Year 2023-24/1444-1445 (Semester I)**

**Department of Information Systems**

**College of Computer Science**

**King Khalid University**

-

Comments by Supervisor(s) / Examiner(s) Name / Signature of Supervisor and examiners

-

|  |  |  |
| --- | --- | --- |
| Name | Comments | Signature |
|  |  |  |
|  |  |  |
|  |  |  |

# **Abstract**

Big data analysis has emerged as a powerful tool in the field of cyber security, enabling organizations to detect and prevent cyber-attacks more effectively. With the exponential growth of data generated by various sources, traditional security measures have become insufficient in identifying and mitigating sophisticated threats. This has led to the adoption of big data analytics techniques, which involve processing and analyzing large volumes of structured and unstructured data to uncover patterns, anomalies, and potential indicators of cyber-attacks. By leveraging advanced algorithms and machine learning models, organizations can proactively identify threats, and enhance their incident response capabilities.

In the proposed project, we propose a web-based system to detect DoS attacks and SQL injection. The proposed system will allow the user through user interfaces to enter the data required for the detection. Relying on artificial intelligence, a model will be trained to be able to detect a denial-of-service attack (DoS) and SQL injection from any data entered to it. Open-source software will be used to develop the model and the proposed website. The proposed project is expected to obtain high accuracy in detecting the above-mentioned attacks.

***Keywords;*** SQL Injection, DoS, Website, HTML, CSS, Python.

# **Declarations**

# **Acknowledgment**

First and foremost, we would like to thank our supervisor, Dr. Sohail, for his invaluable guidance and support throughout the project. We are grateful for his patience, encouragement, and constructive comments. We would also like to thank our colleagues for their help and support. We have learned a lot from them, and we are grateful for their friendship In addition, we would like to thank our teachers for their help and interest in our progress in our project.

# **Table of Contents**

[**Abstract** 3](#_Toc152935874)

[**Declarations** 4](#_Toc152935875)

[**Acknowledgment** 5](#_Toc152935876)

[**Table of Contents** 6](#_Toc152935877)

[**List of Figures** 8](#_Toc152935878)

[**List of Tables** 9](#_Toc152935879)

[**CHAPTER 1: INTRODUCTION AND BACKGRPUND** 10](#_Toc152935880)

[**1.0 Goal** 11](#_Toc152935881)

[**1.1 Motivation** 11](#_Toc152935882)

[**1.2 Signification** 12](#_Toc152935883)

[**1.3 Method** 13](#_Toc152935884)

[**1.4 Report Outline** 14](#_Toc152935885)

[**CHAPTER 2: THEORETICAL BACKGROUND** 16](#_Toc152935886)

[**2.0 What Is Big Data Analysis for The Detection of Cyber Attacks** 16](#_Toc152935887)

[**2.1 Theoretical Concepts Related to Big Data Analysis for Cyber Attack Detection** 16](#_Toc152935888)

[**2.2 SQL Injection Attack** 18](#_Toc152935889)

[**2.3 Denial of Service Attack (DoS)** 19](#_Toc152935890)

[**2.4 The Problem With Big Data in Cyber Attacks** 21](#_Toc152935891)

[**2.5 Similar Work** 22](#_Toc152935892)

[**2.5.1 SQL Injection Detection Using Machine Learning** 22](#_Toc152935893)

[**2.5.2 Development of a Compressive Framework Using Machine Learning Approaches for SQL Injection Attacks** 22](#_Toc152935894)

[**2.5.3 Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning** 23](#_Toc152935895)

[**2.5.4 DDoS Detection Using Machine Learning Technique** 23](#_Toc152935896)

[**CHAPTER 3: PROJECT MANAGEMENT** 25](#_Toc152935897)

[**3.0 Approach** 25](#_Toc152935898)

[**3.1 Risk Management** 26](#_Toc152935899)

[**3.2 Project Plan** 27](#_Toc152935900)

[**3.2.1 Abstract** 27](#_Toc152935901)

[**3.2.2 Work Breakdown** 28](#_Toc152935902)

[**3.2.3 Milestones & Deliverables** 30](#_Toc152935903)

[**3.2.4 The Team: Roles and Responsibilities** 30](#_Toc152935904)

[**CHAPTER 4: REQUIREMENTS AND ANALYSIS** 30](#_Toc152935905)

[**4.0 Requirement Elicitation** 31](#_Toc152935906)

[**4.1 System Requirements** 32](#_Toc152935907)

[**4.1.1 Functional Requirements** 33](#_Toc152935908)

[**4.1.2 Non-Functional Requirements** 34](#_Toc152935909)

[**4.2 Personas** 35](#_Toc152935910)

[**4.3 System Models** 35](#_Toc152935911)

[**4.3.1 Use Case Diagram** 35](#_Toc152935912)

[**4.3.2 Sequence Diagram** 36](#_Toc152935913)

[**4.3.3 Activity Diagram** 38](#_Toc152935914)

[**4.3.4 Class Diagram** 41](#_Toc152935915)

[**CHAPTER 5: SYSTEM DESIGN** 42](#_Toc152935916)

[**5.0 Product Features** 42](#_Toc152935917)

[**5.1 User Interface** 42](#_Toc152935918)

[**5.2 Data Storage** 46](#_Toc152935919)

[**5.3 High Level Design** 46](#_Toc152935920)

[**CHAPTER 6: CONCLUSIONS AND FUTURE WORK** 47](#_Toc152935921)

[**5.0 Future Work** 47](#_Toc152935922)

[**5.1 Conclusion** 48](#_Toc152935923)

[**REFERENCES** 49](#_Toc152935924)

# **List of Figures**

[Figure 3. 1: Milestones & Deliverables 26](#_Toc152504320)

[Figure 4. 1: Use case Diagram 33](#_Toc152504329)

[Figure 4. 2: Sequence Diagram for "Registration" Process 34](#_Toc152504330)

[Figure 4. 3: Sequence Diagram for "Login" Process 35](#_Toc152504331)

[Figure 4. 4: Sequence Diagram for "Attack Detection" Process 35](#_Toc152504332)

[Figure 4. 5: Admin Activity Diagram 36](#_Toc152504333)

[Figure 4. 6: User Activity Diagram 37](#_Toc152504334)

[Figure 4. 7: Class Diagram 38](#_Toc152504335)

[Figure 5. 1: Registration Interface 40](#_Toc152504321)

[Figure 5. 2: Login Interface 40](#_Toc152504322)

[Figure 5. 3: SQL Injection Detector Interface 41](#_Toc152504323)

[Figure 5. 4: DoS Detector Interface 41](#_Toc152504324)

[Figure 5. 5: Information Interface 42](#_Toc152504325)

[Figure 5. 6: Manage Users Interface 42](#_Toc152504326)

[Figure 5. 7: ER Diagram 43](#_Toc152504327)

[Figure 5. 8: System Design 43](#_Toc152504328)

# **List of Tables**

[Table 2. 1: Comparison Table between the proposed system and other systems 24](#_Toc152935925)

[Table 3. 1: Roles and Responsibilities 26](#_Toc152504374)

[Table 4. 1: Interview Questions and Answers 28](#_Toc152504379)

# **CHAPTER 1: INTRODUCTION AND BACKGRPUND**

Big data analysis is a critical tool for detecting cyber-attacks, using its ability to process and analyze large amounts of data in real time. By leveraging big data analysis, organizations can detect and respond to cyber-attacks more quickly, reducing potential damage. Machine learning algorithms can enhance accuracy and efficiency in detecting cyber-attacks, learning from new data and adapting to evolving attack techniques. Big data analysis allows network traffic to be monitored and analyzed in real time to identify suspicious behavior that may indicate a cyber-attack. Insights gained from big data analysis can be used to improve cyber security strategies and strengthen defenses against future attacks. As cyber threats become more sophisticated, big data analysis becomes critical for effective detection and response [1].

In today’s digital age, where data is the new currency, ensuring the security of information systems has become paramount. As technology advances, so do the techniques used by malicious actors to exploit vulnerabilities in these systems. Two common types of attacks that pose significant threats to the security of databases and web applications are SQL injection and denial of service (DoS) attacks. In this project, we will delve deeper into SQL injection and DoS attacks. We will explore their underlying mechanisms, discuss real-world examples illustrating their consequences, analyze preventive measures that can be implemented at different levels, and evaluate existing detection and mitigation techniques. By understanding these attack vectors and learning how to defend against them effectively, we can contribute towards building more secure information systems

This chapter contains a full introduction about the problem definition, project objectives, motivation and the signification of the proposed project.

## **1.0 Goal**

The project's main objective "big data analysis for detection of cyber-attacks" is to develop a robust and efficient system that can analyze large volumes of data in real-time to identify and detect potential cyber-attacks. The project aims to detect two types of cyber-attacks, which are a denial of service attack and an SQL injection attack [2].

The project aims to Leverage big data and advanced analytics techniques to enhance organizations' security posture by proactively identifying and mitigating cyber threats. Overall, the project aims to provide organizations with an effective tool to detect denial of service (DoS) attacks and SQL injection in real-time using big data analysis techniques. By leveraging advanced analytics algorithms on large-scale data sets, it seeks to improve cyber security defenses by identifying threats early and enabling proactive mitigation measures.

## **1.1 Motivation**

There are several motivations for using big data analysis for cyber-attack detection especially DoS attack and SQL injection, including:

1. Protecting assets and data

Detecting cyber-attacks is an important means of protecting assets and important data in organizations and companies, which, if lost or stolen, can have significant negative consequences.

1. Proper data analysis

Big data analysis helps to extract important information from very large and complex data, making it easier to detect attack patterns and schemes.

1. Efficiency and continuous improvement

Big data analysis is used to achieve efficiency and improve the performance of cyber security strategies, which in turn helps to improve business results and reduce error rates.

1. Flexibility and adaptability

Big data analysis allows for quick adaptation to changing conditions, as it can be used to detect constantly changing attack patterns.

1. Gross Domestic Product (GDP)

Cyber-attacks are characterized by their high cost of losses. Big data analysis helps to avoid these major costs, increasing the overall GDP of companies and countries.

## **1.2 Signification**

The project "Big Data Analysis for Detection of SQL injection and DoS attacks" is significant for several reasons:

1. Dos and SQL injection are a growing threat

With the increasing reliance on technology and the internet, Dos and SQL injection have become more frequent and sophisticated. This project aims to address this issue by using big data analysis techniques to detect and prevent such attacks.

1. Big data analysis can handle large volumes of data

Traditional methods of detecting Dos and SQL injection may not be able to handle the massive amounts of data generated in real time. Big data analysis techniques, on the other hand, are designed to process and analyze large volumes of data quickly and efficiently.

1. Early detection and prevention

By analyzing big data, patterns, and anomalies can be identified that may indicate potential Dos and SQL injection attacks. This allows for early detection and prevention measures to be implemented before significant damage occurs.

1. Improved accuracy

Big data analysis techniques can provide more accurate results compared to traditional methods. By analyzing vast amounts of data from various sources, it becomes easier to identify patterns and trends that may indicate a Dos and SQL injection.

1. Real-time monitoring

The project aims to develop real-time monitoring systems that continuously analyze incoming data streams for potential Dos and SQL injection attacks. This proactive approach allows for immediate action to be taken in response to detected Dos and SQL injection.

## **1.3 Method**

There are several methods for using big data analysis to detect and prevent Dos and SQL injection attacks, such as:

1. Data mining

Using data mining techniques is an effective way to identify and understand patterns in large and complex data sets. This makes it easier to detect unusual activities that may indicate SQL injection and DoS attacks.

1. Machine learning

Machine learning algorithms can be trained to recognize patterns of activity that may be associated with cyber-attacks, and can also be used to automate responses to those attacks.

1. Text and sentiment analysis

Text and sentiment analysis can be used to monitor social media, e-mails, and other communications for indications of potential SQL injection and DoS attacks

1. Rule-based detection systems

These systems can be set up to automatically alert security personnel when specific rules or conditions are met that may indicate a SQL injection and DoS attack.

1. Behavioral analysis

Behavioral analysis can be used to identify unusual user or device activity that may indicate a SQL injection and DoS attack. This information can be used to develop more effective cyber security strategies. It's important to note that different organizations may use different combinations of the above methods, depending on their specific needs and resources.

## **1.4 Report Outline**

I. Introduction

A. Background on cyber-attacks and their impact

B. Importance of big data analysis in detecting cyber attacks

II. Overview of Big Data Analysis

A. Definition and characteristics of big data

B. Role of big data analysis in cyber security

III. Data Collection for Cyber Attack Detection

A. Sources of data for analysis (e.g., network logs, system logs, user behavior)

B. Challenges in collecting and storing large volumes of data

IV. Preprocessing and Data Cleaning

A. Techniques for cleaning and preparing raw data for analysis

B. Dealing with missing or incomplete data

V. Feature Extraction and Selection

A. Identification of relevant features for cyber-attack detection

B. Techniques for reducing dimensionality and selecting important features

VI. Machine Learning Algorithms for Cyber Attack Detection

A. Overview of popular machine learning algorithms used in big data analysis for cyber-attack detection (e.g., anomaly detection, classification)

B. Evaluation metrics for assessing the performance of machine learning models

VII. Scalability and Performance Considerations

A. Challenges in processing large-scale datasets efficiently

B. Techniques for distributed computing and parallel processing

VIII. Real-time Monitoring and Alerting Systems

A. Designing real-time monitoring systems using big data technologies (e.g., Apache Kafka, Apache Spark)

B. Generating timely alerts to respond to ongoing cyber attacks

IX. Case Studies and Success Stories

1. Examples of successful implementation of big data analytics in detecting cyber attacks
2. Impact on cyber security posture and incident response time

X Conclusion

A. Summary of key findings

B. Future directions in big data analysis for cyber-attack detection

# **CHAPTER 2: THEORETICAL BACKGROUND**

This chapter includes the theoretical background on which the proposed project is based. Through this chapter, the types of attacks that the proposed system addresses will be introduced, and the basic problem of the research will be described.

## **2.0 What Is Big Data Analysis for The Detection of Cyber Attacks**

Big data analysis for the detection of cyber-attacks involves the use of advanced analytics techniques to process and analyze large volumes of data to identify patterns, anomalies, and potential threats. This approach leverages the massive amounts of data generated by various sources within an organization's network infrastructure, including logs, network traffic, user behavior, and system events [3].

## **2.1 Theoretical Concepts Related to Big Data Analysis for Cyber Attack Detection**

1. Data Collection

Big data analysis relies on collecting diverse and comprehensive datasets from various sources within an organization's network infrastructure. This includes logs from servers, firewalls, intrusion detection systems (IDS), antivirus software, network traffic data, and more.

1. Data Integration

The collected data needs to be integrated into a centralized repository or a data lake for efficient processing and analysis. This involves transforming and normalizing the data to ensure consistency and compatibility across different sources.

1. Data Preprocessing

Before analyzing the data, preprocessing steps are performed to clean and filter out irrelevant or noisy information. This may involve removing duplicate records, handling missing values, normalizing variables, and applying other techniques to improve data quality [4].

1. Anomaly Detection

Big data analytics techniques can be used to identify abnormal patterns or behaviors that deviate from expected norms. Anomaly detection algorithms can help detect potential cyber-attacks by flagging unusual activities such as unauthorized access attempts or abnormal network traffic patterns.

1. Machine Learning Algorithms

Machine learning algorithms play a crucial role in big data analysis for cyber-attack detection. Supervised learning algorithms can be trained on labeled datasets to classify different types of attacks based on historical examples. Unsupervised learning algorithms can help identify unknown attack patterns by clustering similar instances together.

1. Real-time Monitoring

Big data analytics enables real-time monitoring of network activities by continuously analyzing incoming streams of data in near real-time. This allows for timely detection and response to potential cyber threats as they occur.

1. Predictive Analytics

By analyzing historical attack patterns and correlating them with current network conditions, big data analytics can help predict future cyber-attacks or vulnerabilities. Predictive models can assist in proactive threat mitigation strategies.

1. Visualization Techniques

Visualizations play a crucial role in big data analysis for cyber-attack detection as they provide intuitive representations of complex information. Interactive dashboards and visualizations allow security analysts to explore large datasets effectively and identify suspicious activities more efficiently. Overall, big data analysis provides organizations with enhanced capabilities for detecting cyber-attacks by leveraging large-scale datasets and advanced analytical techniques. It enables proactive threat identification, faster response times, improved situational awareness, and better overall cyber security posture.

## **2.2 SQL Injection Attack**

SQL injection is a common security vulnerability in web applications that allows an attacker to manipulate SQL queries by injecting malicious SQL code. One important use case for SQL injection detection is in a web application's login form [5].

The user provides their username and password in a standard login form. An SQL query uses these details to obtain the relevant user record from the database. By changing the input fields to introduce malicious SQL code, an attacker can take advantage of SQL injection vulnerabilities, thereby avoiding authentication and getting unauthorized access to the system. These steps can be taken in order to identify and stop SQL injection:

1. Parameterized Queries

Use prepared statements or parameterized queries rather than dynamically concatenating user inputs into the SQL query string. In order to prevent SQL injection attacks, this method automatically manages correct escaping and encoding and separates the SQL code from the user inputs.

1. Input Validation and Sanitization

Before utilizing user input in SQL queries, make sure it is validated and cleaned. In addition to eliminating or escaping any unusual characters that might be exploited for SQL injection, this entails verifying that the expected data type, length, and format are present.

1. Whitelist Input Filtering

To validate user inputs, employ a whitelist technique that only accepts the characters or patterns that are necessary for each input field. Reject any input that does not meet the requirements specified in the whitelist to aid in preventing malicious SQL code from being executed.

1. Least Privilege Principle

Make sure the application's database user account has the minimal rights required to access the required data. Limiting the privileges lessens the possible damage that could result from an effective SQL injection attack.

1. Web Application Firewall (WAF)

Make use of a WAF with SQL injection prevention and detection features. A WAF can use predetermined patterns or heuristics to evaluate incoming requests and prevent malicious SQL injection attempts.

1. Regular Security Audits

Perform regular security audits and penetration testing on your web application to identify and address any potential SQL injection vulnerabilities. This helps ensure that your application remains secure against emerging threats.

## **2.3 Denial of Service Attack (DoS)**

Denial of Service (DoS) attacks aim to disrupt the normal functioning of a system or network by overwhelming it with a high volume of malicious traffic or resource-intensive requests. Detecting and mitigating DoS attacks is crucial to maintaining the availability and performance of your system. Here's a use case for DoS detection [6]:

1. Traffic Monitoring

To continuously monitor incoming traffic patterns, implement traffic monitoring solutions such as network intrusion detection systems (NIDS) or network traffic analyzers. These programs scan network packets for anomalies or patterns that point to a denial-of-service assault, including an abrupt spike in traffic or an abnormally large number of requests coming from one source.

1. Rate Limiting

To limit how many requests are accepted from a particular IP address or client in a given amount of time, implement rate limitation techniques. You can stop a single source from flooding your system with too many requests by imposing limitations and setting thresholds.

1. Anomaly Detection

To find unusual activity in system measurements or network traffic, apply anomaly detection algorithms. Analyzing traffic patterns, resource usage, or request rates may be part of this. Algorithms for machine learning can be trained to identify typical patterns, issue alarms, or take preventative action when deviations happen.

1. Traffic Filtering and Blacklisting

Use traffic filtering techniques to restrict or stop access from IP addresses that seem dubious or malicious. Keep track of known attackers or dubious sources on a blacklist, and set up your firewalls or network devices to refuse traffic coming from those addresses.

1. Load Balancing and Redundancy

Distribute incoming traffic across multiple servers using load balancers. This helps distribute the load and mitigate the impact of a DoS attack by allowing traffic to be handled by multiple resources. Additionally, redundancy in critical components helps maintain service availability even if a particular resource becomes overwhelmed.

1. DDoS Mitigation Services

Consider utilizing Distributed Denial of Service (DDoS) mitigation services provided by specialized vendors. These services employ advanced traffic analysis techniques, rate limiting, and traffic filtering to detect and mitigate large-scale DDoS attacks before they reach your network.

1. Incident Response and Monitoring

Create a monitoring system and incident response plan to promptly detect and address possible denial-of-service attacks. Set up tools for tracking and analyzing system and network events through logging and monitoring to enable quick identification and investigation of suspicious activity.

## **2.4 The Problem With Big Data in Cyber Attacks**

SQL injection and denial of service (DoS) attacks are both serious cyber security threats, each with distinct characteristics and potential consequences, the main problem in SQL injection is attackers exploit vulnerabilities in an application's code to insert malicious SQL statements into input fields, URLs, or cookies. And the main problem in Denial of services It is that the attack aims to shut down a device or network, making it inaccessible, which affects the data and may lead to significant data loss or its duplication [7].

The problem with big data about cyber-attacks lies in the vulnerabilities and risks associated with handling and storing large volumes of data. Big data refers to the massive amounts of information that organizations collect, analyze, and utilize for various purposes. However, this abundance of data also presents attractive targets for cybercriminals. Here are some key issues related to big data and cyber-attacks [8]:

1. Data Breaches: Big data repositories often contain sensitive information such as personal details, financial records, or intellectual property. If these databases are compromised through cyber-attacks like hacking or unauthorized access, it can lead to significant privacy breaches and financial losses.
2. Data Integrity: Cyber-attacks can also target the integrity of big data by altering or corrupting the stored information. Manipulating critical datasets can have severe consequences, especially in sectors like finance, healthcare, or critical infrastructure where decisions are made based on accurate and reliable information.
3. Lack of Security Measures: Managing and securing big data environments can be complex due to their scale and diversity. Organizations may struggle to implement adequate security measures across all components involved in storing, processing, and analyzing big data sets. This lack of comprehensive security measures increases the risk of successful cyber-attacks.

To mitigate these risks, organizations must prioritize cyber security measures such as implementing strong access controls, encryption techniques, regular security audits, employee training on best practices, and employing advanced threat detection systems to safeguard their big data infrastructure from potential cyber threats.

## **2.5 Similar Work**

### **2.5.1 SQL Injection Detection Using Machine Learning**

The authors proposed a methodology that uses machine learning algorithms to analyze and identify patterns that indicate SQL injection attacks. In the data collection phase, the authors collect a dataset consisting of legitimate and malicious SQL queries. They pre-process the data by removing noise and irrelevant information, and then extract relevant features from SQL queries, such as specific keywords or grammatical structures that may indicate an injection attack [9].

To train machine learning models, the authors experimented with different algorithms, including Naive Bayse, Logistic Regression, SVM, CNN, Passive Aggressive Classifier. The models are trained on a labeled dataset, where legitimate queries are classified as normal and malicious queries as attacks.

After experimenting with the five algorithms, building each model, training it, testing it, and measuring its accuracy, it was concluded that the CNN algorithm was the best at detecting an SQL injection attack, as the accuracy of the model was 97%.

### **2.5.2 Development of a Compressive Framework Using Machine Learning Approaches for SQL Injection Attacks**

The paper discusses the creation of a framework that utilizes machine learning techniques to detect and prevent SQL injection attacks. The framework proposed in the paper aims to enhance the existing security measures by applying machine learning algorithms to analyze and identify patterns in SQL queries that could potentially indicate an injection attack. By training the model on a large dataset of both legitimate and malicious queries, it learns to distinguish between them and provide accurate predictions. The authors describe the various steps involved in developing the framework, including data collection, preprocessing, feature extraction, and model training. They employ different machine learning algorithms, such as Decision Trees, Support Vector Machines, and Neural Networks, to build and evaluate the effectiveness of the framework [10].

The results of the experiments conducted by the authors demonstrate that the proposed framework achieves high accuracy in detecting SQL injection attacks while maintaining a low false positive rate. The accuracy results were 99% for the hybrid approach and 98% for ANN .The framework shows promise in providing an additional layer of defense against SQL injection attacks, thereby enhancing the security posture of web applications and databases.

### **2.5.3 Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning**

In this research, a unique machine learning approach for real-time detection of Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks is presented. The suggested method focuses on online detection, which entails real-time network traffic analysis to spot indications of an ongoing attack. The system learns patterns and traits of typical network behavior by utilizing machine learning techniques, and it is therefore able to identify anomalies that might point to an attack. The authors use a dataset that includes both normal and attack traffic to train the machine learning models. They use a variety of methods, like As Random Forests, Decision Trees, and Support Vector Machines, to create models that reliably identify malicious or benign network traffic [11].

Using a sample rate (SR) of 20% of network traffic, the results provide an online detection rate (DR) of attacks above 96%, with high precision (PREC) and low false alarm rate (FAR).

### **2.5.4 DDoS Detection Using Machine Learning Technique**

The paper focuses on machine learning techniques for Distributed Denial of Service (DDoS) attack detection. The writers clean and filter the network traffic data during the data preprocessing step to get rid of noise and unnecessary information. From the preprocessed data, they then extract pertinent information including source IP addresses, packet sizes, and packet rates. The machine learning models use these features as input [12].

To train and assess the models, the authors test a range of machine learning algorithms, such as K-Nearest Neighbors, Decision Trees, and Support Vector Machines. The models are trained and their performance evaluated using a labeled dataset comprising both regular and DDoS attack traffic. The outcomes demonstrate that the machine learning models successfully identify DDoS assaults by achieving high accuracy in differentiating between attack and legitimate traffic. The random forest algorithm's accuracy was 99.76%.

Table 2. 1: Comparison Table between the proposed system and other systems

|  |  |  |  |
| --- | --- | --- | --- |
|  | **ML Algorithms** | **Type of Attacks** | **Accuracy** |
| **SQL Injection Detection Using Machine Learning** | Naive Bayse, Logistic Regression, SVM, CNN, Passive Aggressive Classifier. | SQL Injection | CNN 97% |
| **Development of a Compressive Framework Using Machine Learning Approaches for SQL Injection Attacks** | Decision Trees, Support Vector Machines, and Neural Networks | SQL Injection | 99% for the hybrid approach and 98% for ANN |
| **Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning** | Random Forests, Decision Trees, and Support Vector Machines | DoS Attack | 96% |
| **DDoS Detection Using Machine Learning Technique** | K-Nearest Neighbors, Decision Trees, and Support Vector Machines | DDoS Attack | 99.76% |
| **Big Data Analysis for The Detection of Denial of Service & SQL Injections** | CNN Algorithm | SQL Injection & DoS Attack | - |

# **CHAPTER 3: PROJECT MANAGEMENT**

This chapter contains the model used for project development, risk management, and project plans.

## **3.0 Approach**

* We define project objectives to develop a machine learning model that can detect and prevent cyber-attacks using big data analysis.
* Defining objectives helps provide clarity and focus, ensuring that all efforts are aligned toward achieving specific outcomes.
* We create a project plan and break down the project into smaller tasks or milestones with defined timelines, responsibilities, and dependencies.
* A well-structured project plan provides a roadmap for executing tasks efficiently, ensures accountability among team members, and helps manage resources effectively.
* Identify the required resources and determine the necessary resources, such as hardware, software, data sources, and human expertise, needed for the project. Allocate these resources accordingly.
* To ensure that we have everything necessary to carry out our project smoothly without any delays or bottlenecks
* Risk management plan identifies potential risks that may impact our project's success (e.g., data breaches, technical failures). Assess their likelihood and impact on the project's timeline and deliverables. Develop strategies to mitigate or respond to these risks.
* Finally, we communicate effectively with stakeholders. Maintain open communication channels with stakeholders, such as team members and our supervisor, throughout the project. Provide regular updates on progress and address any concerns promptly.

## **3.1 Risk Management**

1. Lack of expertise: Big data analytics and cyber security require specialized knowledge and skills, we decided to consider collaborating with experts or seeking guidance from professors or professionals who have experience in the field.
2. Technical challenges: Dealing with large datasets can present technical challenges such as storage limitations, processing power requirements, and scalability issues. To manage these challenges, we will plan our infrastructure requirements carefully, consider cloud-based solutions for storage and processing, and optimize our algorithms for efficient computation.
3. Time constraints: University projects often have strict deadlines, which can put pressure on completing the project within a limited timeframe. To manage this risk, we create a realistic project timeline with clear milestones and allocate sufficient time for each task. Regularly monitor progress to ensure that we stay on track.
4. Ethical considerations: When working with big data related to cyber-attacks, it is important to consider ethical implications such as potential biases in the data or unintended consequences of our analysis. Be mindful of these considerations throughout our project and seek guidance from ethics committees or advisors if needed.
5. 5. Limited resources: As students, we may have limited access to resources such as computing power or specialized software tools. To manage this risk, explore open-source tools and libraries that can help us achieve our objectives without significant financial constraints.
6. Collaboration challenges: working as a team on the project, there may be challenges related to coordination, communication, and division of tasks. To mitigate these risks, we will establish clear roles and responsibilities within the team, maintain regular communication channels (e.g., meetings or online collaboration platforms), and address any conflicts or issues promptly.

By identifying these risks early on and implementing appropriate strategies to manage them effectively, we can increase the chances of success in our graduation project on big data for the detection of cyber-attacks.

## **3.2 Project Plan**

### **3.2.1 Abstract**

Initially, cyber security did not have much importance in society and there was a lack of awareness and understanding of vulnerabilities and attacks affecting the security of devices and information. But now cyber security has become one of the most important matters, awareness of information security and its sensitivity has increased, and prevention methods used to protect sensitive data, whether it belongs to the state or an individual in society, have increased, which has led to an increase in attack methods and a diversity of penetration methods, such as penetrating servers or main devices of institutions and penetrating personal devices. Manipulating the information they contain, intercepting data during its transmission over networks, as well as defacing websites through changing their content or denial of service attacks. This increases the amount of data that must be protected. . . Therefore, the idea of ​​analyzing big data to detect cyber threats came to reduce the problems of security vulnerabilities and attacks that may cause data to be lost, modified, deleted, or used for blackmail. To solve these problems, we need to periodically analyze data, protect it, and detect vulnerabilities, as well as use machine learning and deep learning techniques to detect cyber threats, improve response, and ensure recovery. The goal of these operations is to protect data, reduce vulnerabilities, eliminate cyber threats, contribute to protecting sensitive and important data, reduce blackmailing of users through important data, and maintain privacy and information security, in addition to state security

### **3.2.2 Work Breakdown**

The project "Big Data Analysis to Detect Cyber Threats" involves using advanced techniques and tools to analyze large volumes of data to identify and prevent potential cyber threats. It focuses on utilizing big data analytics to detect patterns, anomalies, and trends that may indicate malicious activities or vulnerabilities in computer systems, networks, or digital platforms. The goal is to enhance cyber security measures by proactively identifying and mitigating cyber threats before they can cause harm. Here are the steps for our project:

1. Data Preprocessing: Cleaning and transforming the collected data to ensure its quality and compatibility for analysis. This may involve removing duplicates, handling missing values, standardizing formats, etc.
2. Data Integration: Combining data from various sources into a unified format or schema for effective analysis. This could involve merging different log formats or integrating threat intelligence feeds with internal data.
3. Feature Extraction: Identifying relevant features or attributes from the collected data that can be used to detect cyber threats. This may include extracting information such as IP addresses, timestamps, user behavior patterns, network traffic characteristics, etc.
4. Machine Learning Model Development: Designing and implementing machine learning algorithms or models suitable for analyzing big data to detect cyber threats. This could involve techniques like anomaly detection, classification algorithms (e.g., decision trees, random forests), clustering algorithms (e.g., k-means), etc.
5. Model Training and Evaluation: Training the developed models using labeled or annotated data to learn patterns of cyber threats. Evaluating the performance of these models using appropriate metrics such as accuracy, precision, recall, F1-score, etc.
6. Real-time Monitoring and Alerting: Implementing a system that continuously monitors incoming data streams in real-time and generates alerts or notifications when potential cyber threats are detected based on the trained models.
7. Visualization and Reporting: Creating visualizations and reports to present the analyzed results understandably for stakeholders
8. Iterative Improvement: Continuously refining and improving the models based on feedback from real-world detections and incorporating new threat intelligence information to enhance the accuracy and effectiveness of the system.
9. Security and Privacy Considerations: Ensuring that appropriate security measures are in place to protect the collected data, comply with privacy regulations, and prevent unauthorized access or misuse.
10. Documentation: Document all processes, methodologies, algorithms used, findings, and recommendations for future reference or replication of the project.

### **3.2.3 Milestones & Deliverables**

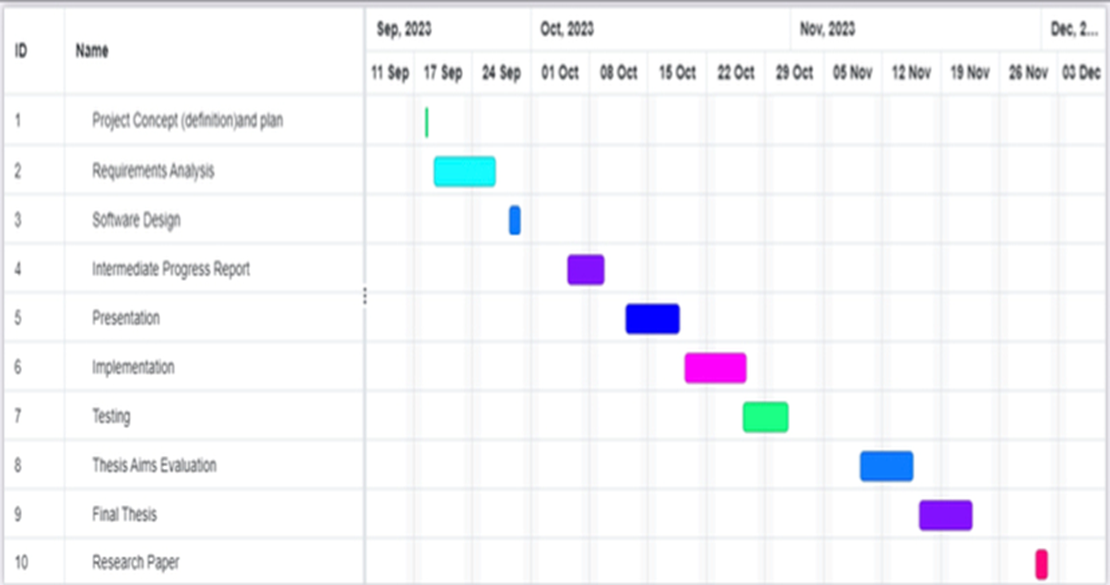


Figure 3. 1: Milestones & Deliverables

### **3.2.4 The Team: Roles and Responsibilities**

Table 3. 1: Roles and Responsibilities

|  |  |
| --- | --- |
| **Responsibilities** | **Student Name** |
| **Project concept (definition) and plan, thesis aim evaluation & final thesis.** | **Sara abdulkarim asiri** |
| **Requirements analysis, Software design & presentation** | **Hajer Mohammed almost** |
| **Intermediate progress report, implementation, research paper & testing.** | **Rajwah mohammed qazwani** |

# **CHAPTER 4: REQUIREMENTS AND ANALYSIS**

System analysis is an important process that involves gathering and analyzing data in order to pinpoint issues and requirements and divide the system into its constituent parts. System analysis is done to precisely identify its goals. It is a technological advancement that makes the system better and guarantees that every system part functions effectively to fulfill its intended role and resolve the examined issue.

The Requirement Elicitation method will be described in this chapter. Subsequently, the technological requirements utilized in the system development process will be discussed, along with the functional and non-functional requirements of the suggested system. In the last section of this chapter, UML diagrams will be drawn containing the Use Diagram, Sequence Diagram, Activity Diagram, and Class Diagram.

## **4.0 Requirement Elicitation**

There are many techniques for requirements elicitation, as the following;

1. **Interviews**

An interview has a list of questions about a project requirement. The questions are a series of project objective, steps or checklists. Essentially, each step of the Elicitation Interview should have a list of questions to gather information.

1. **Brainstorming Sessions**

Brainstorming is a technique for collective deduction that involves introducing an issue or topic to a group of people and asking them to assess it and offer ideas for how to solve it. The concepts and realizations that are obtained from the participants are documented during the brainstorming session.

1. **Use case Approach**

The use case method is a popular and successful way to extract software requirements. With this technology, users can concentrate on their needs rather than the system's operation.

1. **Quality Function Deployment**

A set of instruments called Quality Function Deployment (QFD) is used to ascertain what a consumer needs from a product. In order to meet the customer's requirements for this product, this procedure assists in converting those requirements into specifications and engineering plans .

1. **Questionnaire**

A list of questions concerning the requirements of a project is called a requirements questionnaire. Answers to a few questions on the project concept are requested from participants. This method aids in determining the requirements for the project.

In this project, interviews will be used as a means of collecting data, due to time constraints and our desire to collect answers directly from participants. We chose four interview questions related to the proposed project and its importance, and we interviewed five people who are users of modern technology and have moderate experience with Internet attacks.

Table 4. 1: Interview Questions and Answers

|  |  |
| --- | --- |
| **Question** | **Answer** |
| **Q1: Do you know the types of cyber-attacks?** | Yes |
| No |
| No |
| Yes |
| No |
| **Q2: Have you ever been exposed to any of these attacks?** | Yes |
| No |
| No |
| Yes |
| No |
| **Q3: Have you ever heard of a denial-of-service attack and a SQL attack?** | Yes |
| No |
| No |
| Yes |
| No |
| **Q4: If you could use a system to detect these attacks, would you use it?** | Yes |
| Maybe |
| Yes |
| Yes |
| Yes |

After the interview, we concluded that most people do not have knowledge of the types of attacks, especially denial of service (DoS) attacks and SQL injections, so it was necessary to think about including an information section that tells users information about these two types of attacks.

## **4.1 System Requirements**

System requirements contain the functional and non-functional requirements. The following sections contain the requirements of the proposed system.

### **4.1.1 Functional Requirements**

There are two main users in the proposed system: The admin and the user. The functional requirements of each user are as the following;

#### **4.1.1.1 User Functional Requirements**

* **Create Account**

The user can register in the system by creating an account in the system. The user has to add his user’s name, email address and password.

* **Log in**

The user can log into the system after registration. The user has to add his user’s name and password to log in.

* **Reset Password**

The user must be able to reset his password if he forgets it when logging into the system.

* **Enter SQL Query**

The user must be able to enter a SQL query to check if there us SQL injection or not.

* **Enter Network data**

The user must be able to enter network data to cheek if there is DoS attack or not.

* **Get Results**

The user must be able to get results about SQL injection and DoS detections after the system completes the detection process.

* **Read Information about Cyber-Attacks**

The user must be able to read information to know more about cyber-attacks.

* **Log Out**

The user must be able to log out of the system.

#### **4.1.1.2 Admin Functional Requirements**

* **Create Account**

The admin can register in the system by creating an account in the system. The admin has to add his user’s name, email address and password.

* **Log in**

The admin can log into the system after registration. The admin has to add his user’s name and password to log in.

* **Reset Password**

The admin must be able to reset his password if he forgets it when logging into the system.

* **Manage User Accounts**

The admin must be able to manage user account in the system by adding a new account or by editing or deleting an existing one.

* **Add Information about Cyber-Attacks**

The admin can add information about cyber-attacks to enable users know more about this type of attacks, especially SQL Injection and DoS Attack.

* **Log Out**

The admin must be able to log out of the system.

### **4.1.2 Non-Functional Requirements**

* **Ease of use**

The user must be able to use the system even if he is not an expert in technology, the user interface should be simple and easy to understand.

* **Security**

The system must be able to protect users' sensitive data, such as email addresses and passwords.

* **Availability**

The system must be available anytime and from any device using any web browser.

* **Response time**

The system response time for any process should not be more than 10ms.

* **Database security**

To safeguard user data, a secure system database must be constructed and all sensitive data, including passwords and IDs, must be encrypted before being stored in the database.

* **Scalability**

The system should be able to improve response time or improve performance through upgrading and updating.

## **4.2 Personas**

The system contains two types of users:

* The **user** who wants to detect the cyber-attacks he was exposed to, whether it was a SQL injection or a DoS attack. This user does not have to be a technology expert. The user can read information about the two types of attacks, their seriousness, their negative effects, and ways to deal with them.
* The **administrator** who manages the operations in the system, where he manages user accounts (adds an account, deletes, or modifies the data of any account) and is also able to add information about the two types of cyber-attacks that the specific system detects.

## **4.3 System Models**

This section describes different system models. We chose the Object-Oriented (OO) approach to describe the create model.

### **4.3.1 Use Case Diagram**

The use case diagram of the proposed system is represented in Figure 4.1.

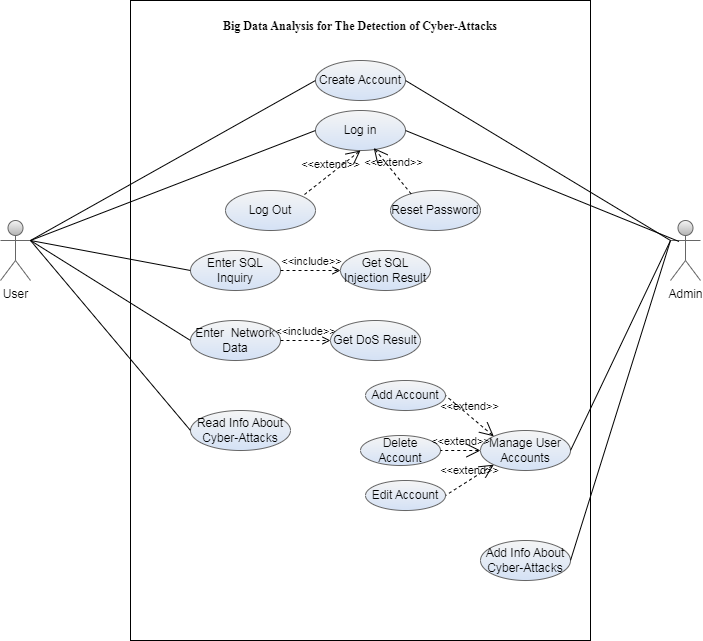


Figure 4. 1: Use case Diagram

### **4.3.2 Sequence Diagram**

The sequence diagram of the registration process is represented in Figure 4.2.

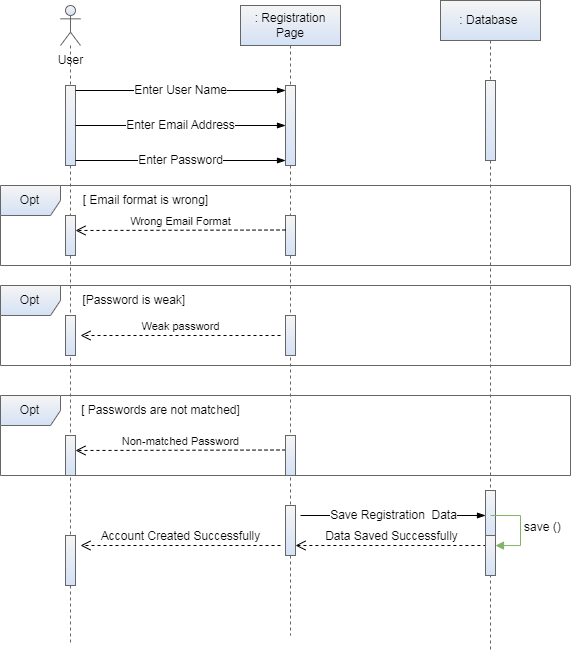


Figure 4. 2: Sequence Diagram for "Registration" Process

The sequence diagram of the login process is shown in Figure 4.3.

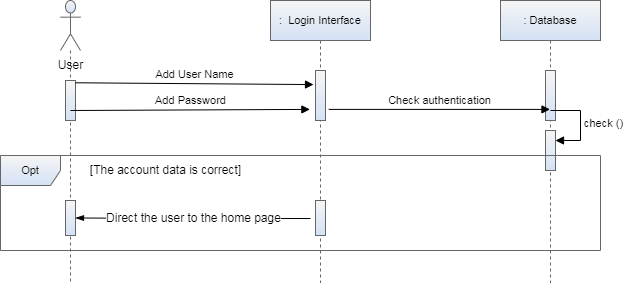


Figure 4. 3: Sequence Diagram for "Login" Process

The sequence diagram of the detection of the SQL Injection and DoS attack is shown in Figure 4.4.

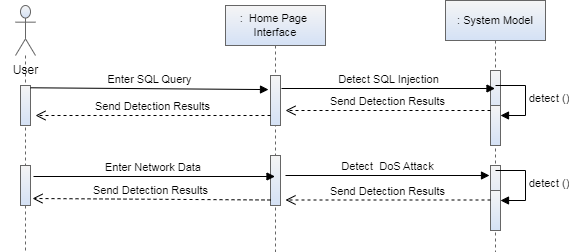


Figure 4. 4: Sequence Diagram for "Attack Detection" Process

### **4.3.3 Activity Diagram**

Figure 4.5 shows the activity diagram of the admin.

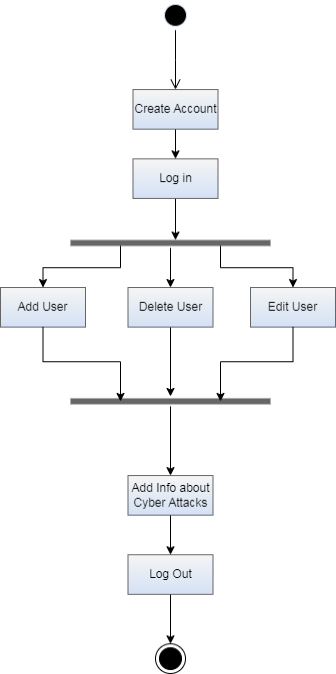


Figure 4. 5: Admin Activity Diagram

Figure 4.6 represents the activity diagram of the user.

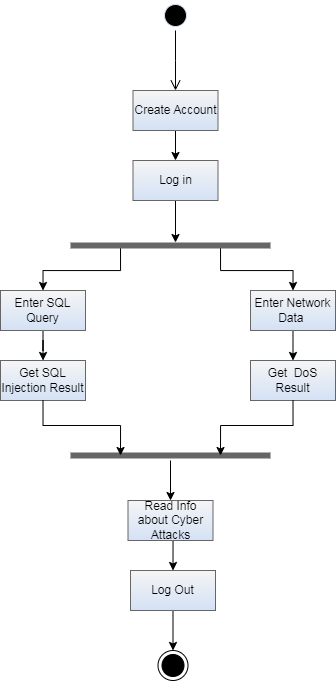


Figure 4. 6: User Activity Diagram

### **4.3.4 Class Diagram**

The class diagram gives an overview of what a system's database and its components look like. In the proposed system, there are three classes which are: User class, Admin class, SQL Query class, Network data class, Result class and Information Class. Figure 4.7 represents the class diagram of the proposed system.

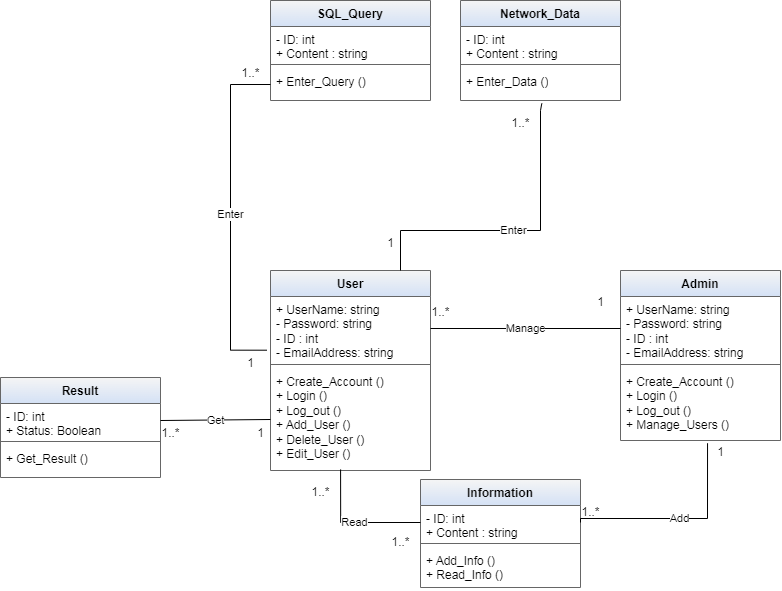


Figure 4. 7: Class Diagram

This chapter contained the system requirements determining and system analysis phases. This chapter provides a description of both functional and non-functional requirements and it also contain the system models. The next chapter will contain the system design phase.

# **CHAPTER 5: SYSTEM DESIGN**

This chapter documents the detailed design of the project. The project will be designed after choosing the appropriate design methodology. We chose the Prototype methodology to design the project interfaces and get a general idea of how the proposed project will work.

## **5.0 Product Features**

It is essential that the user interface is well designed because it is the link between the end user and the software. So, the user interface should be easy, simple and usable. On the other hand, the system must meet the user's need to detect cyber security threats (SQL injection and denial of service attack). The user must be able to interact with the proposed system so that he can enter his data to create an account and log in. There are the features of the proposed system:

* SQL Injection Detection
* DoS Attack Detection
* Detailed Information about Cyber Attacks

## **5.1 User Interface**

* SQL Injection Detection

The user must be able to enter SQL queries into the system in order to finally obtain the scan result. The output will be the detection results of SQL injection.

* DoS Attack Detection

The user must be able to enter network data into the system in order to finally obtain the scan result. The output will be the detection results of DoS attack.

* Detailed Information about Cyber Attacks

The user can also access the information entered by the admin to read more about denial-of-service attacks and SQL injection.

Figure 5.1 represents the registration interface of the proposed project. The user has to add his user’s name, email address and password.

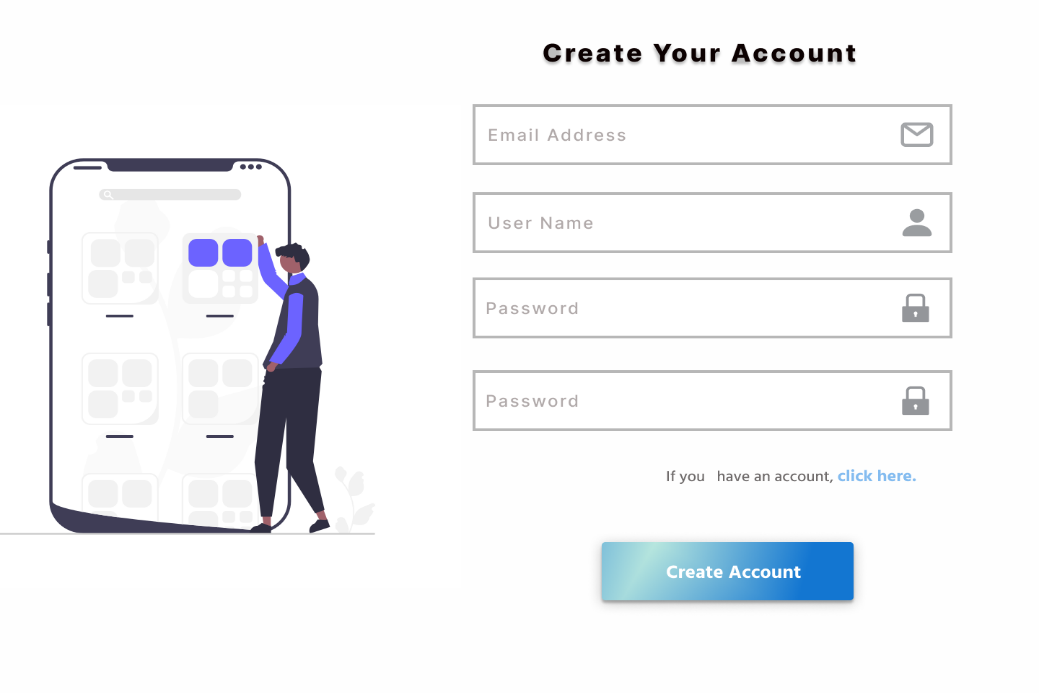


Figure 5. 1: Registration Interface

Figure 5.2 shows the login interface. The user has to add his email address and password. If the user forgets his password, he can click on "Forget Password" link. If the user doesn't have an account, he can click on "Create Account" link.

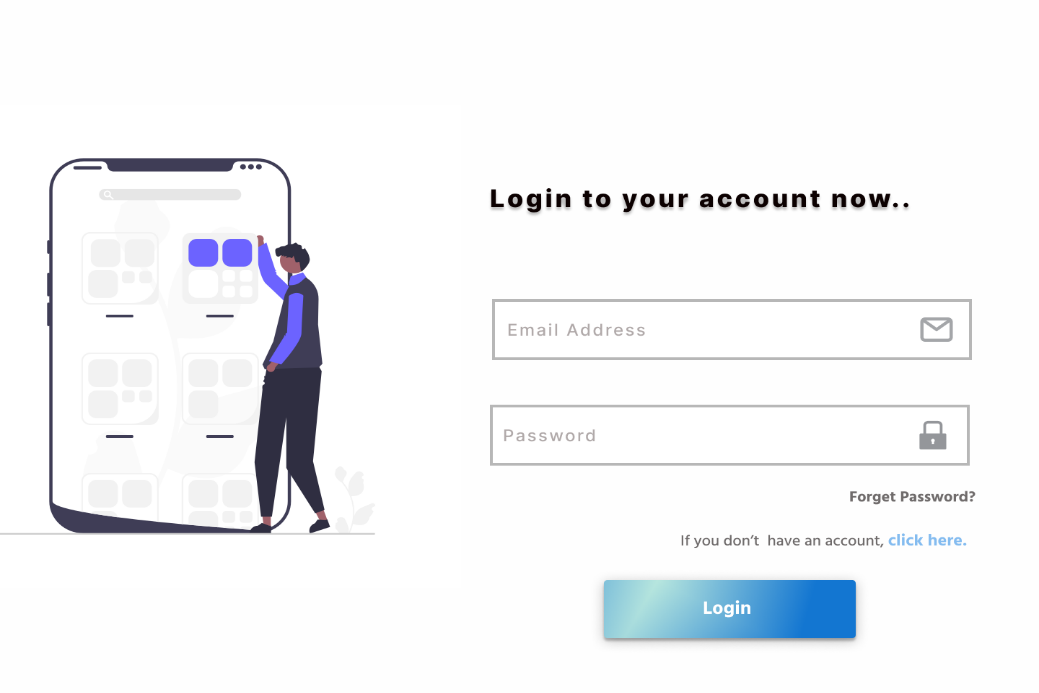


Figure 5. 2: Login Interface

The user can enter a SQL query to check if there us SQL injection or not, as shown in Figure 5.3. The user must be able to get results about SQL injection after the system completes the detection process.



Figure 5. 3: SQL Injection Detector Interface

The user can enter the network data to detect if there is a DoS attack or not, as shown in Figure 5.4. The user will be able to get results about DoS detections after the system completes the detection process.

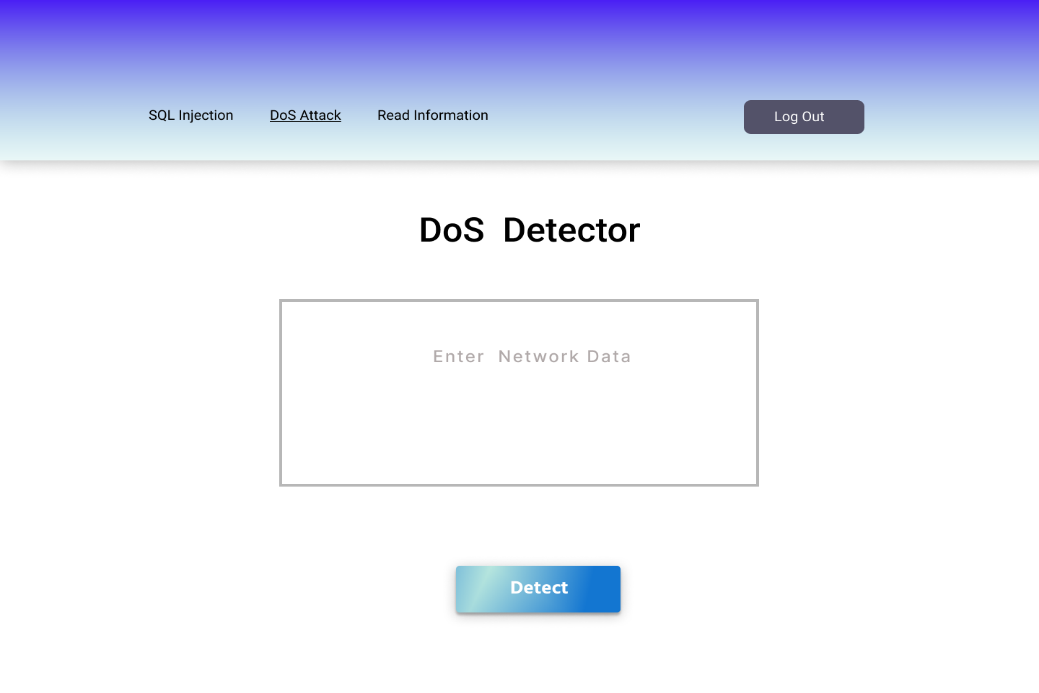


Figure 5. 4: DoS Detector Interface

The user can read information about DoS attack and SQL Injection attack, as shown in Figure 5.5.



Figure 5. 5: Information Interface

The admin can manage the user's account by editing or deleing any account, as shown in Figure 5.6.

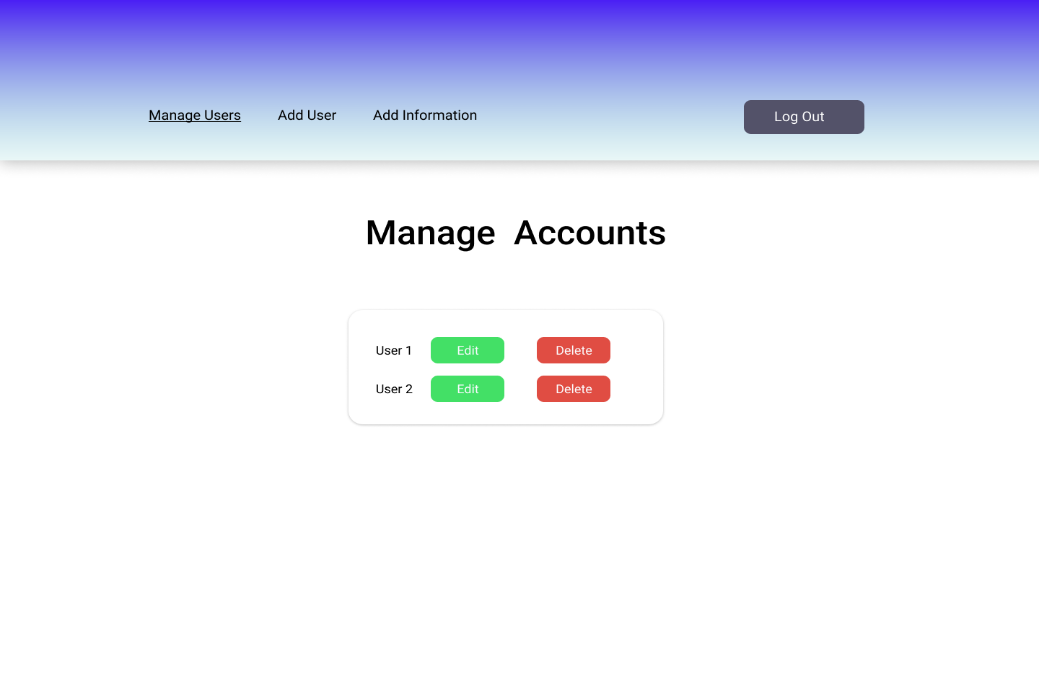


Figure 5. 6: Manage Users Interface

## **5.2 Data Storage**

Figure 5.7 represents the entity relationship diagram which represents the system database.

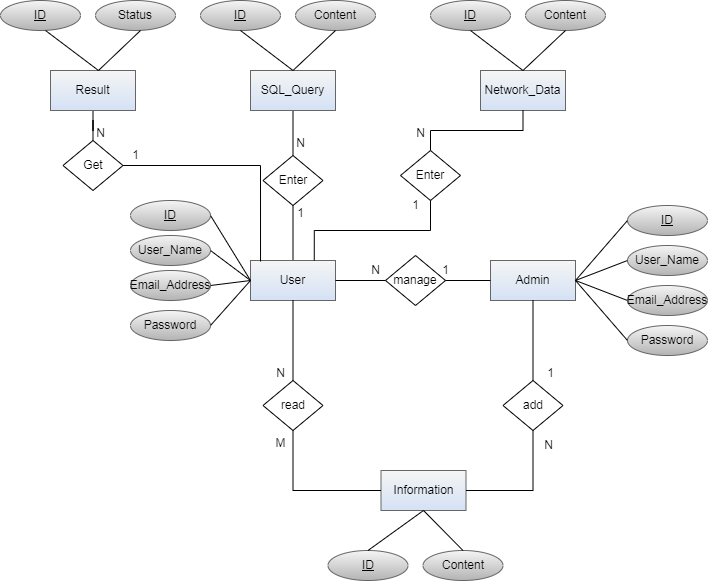


Figure 5. 7: ER Diagram

## **5.3 High Level Design**

Figure 5.8 represents the system design architecture, where the user enters the SQL query or the network data and the system detects if there are any SQL injection or DoS attacks.

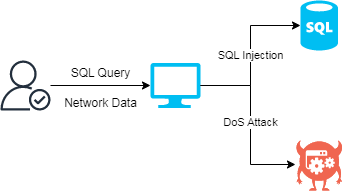


Figure 5. 8: System Design

# **CHAPTER 6: CONCLUSIONS AND FUTURE WORK**

Big data analysis has become a potent instrument in the cyber security space, helping enterprises to more effectively identify and stop cyber-attacks. The amount of data created by several sources has increased exponentially, making traditional security methods inadequate for detecting and thwarting complex threats.

We suggest using a web-based system in the proposed project to identify SQL injection and DoS attacks. The suggested system will enable the user to enter the data needed for the detection using user interfaces. Through the use of artificial intelligence, a model will be taught to recognize SQL injection and denial of service attacks (DoS) in any input data. Both the model and the suggested website will be developed using open-source software. It is anticipated that the proposed technology will achieve high accuracy in identifying the aforementioned attacks.

## **5.0 Future Work**

* Starting the implementation of the proposed website.
* Building and training the proposed model.
* Testing and evaluating the proposed system.
* Creating and developing mobile applications running on Android and iOS.
* Training the model to detect more cyber-attacks.
* Developing new machine learning algorithms that are better able to identify patterns in data that may indicate a cyber-attack.
* Developing new statistical analysis techniques that are better able to identify anomalies in data that may indicate a cyber-attack.
* Developing new heuristics for identifying suspicious activity.
* Developing new ways to integrate big data analytics with other cybersecurity technologies, such as intrusion detection systems and intrusion prevention systems.

## **5.1 Conclusion**

In the first stage of Project 1, the group collected the requirements that were needed for the project that was being suggested. After then, a thorough analysis of pertinent literature about projects with related goals was carried out. After the project features were determined, the requirements analysis phase was started. During this phase, system diagrams and functional and non-functional demands were drawn out. In the end, a Prototype is created to provide a thorough representation of the interfaces inside the project. It is expected that the project will be finished in the allotted time frame and produce worthwhile results that are in line with the desired goals.

# **REFERENCES**

[1] O.Ben Fredj, A.Mihoub, M.Krichen,O.Cheikhrouhou & A.Derhab, CyberSecurity attack prediction: a deep learning approach, In 13th international conference on security of information and networks (pp. 1-6), 2020.‏

[2] M.Al-Omari, M.Rawashdeh, F.Qutaishat,M.Alshira’H & N.Ababneh, An intelligent tree-based intrusion detection model for cyber security, Journal of Network and Systems Management, 29, 1-18, 2021.‏

[3] A.Pokhrel, V.Katta & R.Colomo-Palacios, .Digital twin for cybersecurity incident prediction: A multivocal literature review, In Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops (pp. 671-678), 2020.‏

[4] M.Ahsan, R.Gomes, M.Chowdhury & K.Nygard, Enhancing machine learning prediction in cybersecurity using dynamic feature selector, Journal of Cybersecurity and Privacy, 1(1), 2021.‏

[5] I.Sarker, Machine learning for intelligent data analysis and automation in cybersecurity: current and future prospects, Annals of Data Science, 10(6), 1473-1498, 2023.‏

[6] S.Chen, Z.Wu & PChristofides, A cyber‐secure control‐detector architecture for nonlinear processes, AIChE Journal, 66(5), e16907, 2020.‏

[7] I.Jemal, O.Cheikhrouhou, H.Hamam & A.Mahfoudhi, Sql injection attack detection and prevention techniques using machine learning, International Journal of Applied Engineering Research, 15(6), 569-580, 2020.‏

[8] A.Yusof, N.Udzir & A.Selamat, Systematic literature review and taxonomy for DDoS attack detection and prediction, International Journal of Digital Enterprise Technology, 1(3), 292-315, 2019.‏

[9] S.Krishnan, A.Sabu, P.Sajan & A.Sreedeep, SQL injection detection using machine learning, 2021.

[10] F.Deriba, A.Salau, S.Mohammed, T.Kassa & W.Demilie, Development of a compressive framework using machine learning approaches for SQL injection attacks, PRZEGLĄD ELEKTROTECHNICZNY, 2022.

[11] F.Lima Filho, F.Silveira, A.de Medeiros Brito Junior, G.Vargas-Solar & L.Silveira, Smart detection: an online approach for DoS/DDoS attack detection using machine learning, Security and Communication Networks, 2019.

[12] S.Pande, A.Khamparia, D.Gupta & D.Thanh, DDOS detection using machine learning technique, In Recent Studies on Computational Intelligence: Doctoral Symposium on Computational Intelligence (DoSCI 2020) (pp. 59-68), 2021.‏‏‏‏