**A**

**Minor Project Report**

**On**

**DDoS Attack Prediction Using Machine Learning**

**Submitted to**

**CHHATTISGARH SWAMI VIVEKANAND TECHNICAL UNIVERSITY,**

**BHILAI**

***in partial fulfillment of requirement for the award of degree* of**

**MASTER OF COMPUTER APPLICATION**

**Semester III**

**By**

**Prashant Narayan Dev, 501302122009, CC2880**

**Ankitosh Kumar, 501302122031, CC2833**

**Torsa Pradhan, 501302122016, CC2877**

**Chandan Singh, 501302122005, CC2883**

**Lokendra , 501302122029, CC2873**

**Under the Guidance of**

**Prof. Deepak Pandey Sir**

**Professor**



**DEPARTMENT OF MASTER OF COMPUTER APPLICATION,**

**RUNGTA COLLEGE OF ENGINEERING & TECHNOLOGY,**

**KOHKA-KURUD ROAD, BHILAI, CHHATTISGARH, INDIA**

**Session : 2023-2024**

**D E C L A R A T I O N**

We, the undersigned, solemnly declare that this report on the project work entitled “DDoS ATTACK PREDICTION USING MACHINE LEARNING”, is based on our own work carried out during the course of our study under the guidance of Prof. Deepak Pandey Sir.

We assert that the statements made and conclusions drawn are an outcome of the project work. We further declare that to the best of our knowledge and belief the report does not contain any part of any work which has been submitted for the award of any other degree/diploma/certificate in this University or any other University.

Signature:

Prashant Narayan Dev

501302122009

CC2880

Ankitosh Kumar

501302122031

CC2833

Torsa Pradhan

501302122016

CC2877

Chandan Singh

501302122005

CC2883

Lokendra

501302122029

CC2873

**C E R T I F I C A T E**

This is to certify that this report on the project submitted is an outcome of the project work entitled “DDoS ATTACK PREDICTION USING MACHINE LEARNING”**,** carried out by the students in the **DECLARATION**, is carried out under my guidance and supervision for the award of Degree in **MASTER OF COMPUTER APPLICATION** of Chhattisgarh Swami Vivekanand Technical University, Bhilai(C.G.), India.

To the best of my knowledge the report-

1. Embodies the work of the student(s) themselves,
2. Has duly been completed,
3. Fulfills the requirement of the Ordinance relating to the B.Tech. degree of the University, and
4. Is up to the desired standard for the purpose for which it is submitted.

**Deepak Pandey Sir**

**Professor**

This project work as mentioned above is hereby being recommended and forwarded for examination and evaluation by the University,

**Dr. Ajay Khushwaha**

Head,

Department of Master of Computer Application,

Rungta College of Engineering & Technology,

Kohka - Kurud Road, Bhilai(C.G.), India

**C E R T I F I C A T E B Y T H E E X A M I N E R S**

This is to certify that this project work entitled “DDoS ATTACK PREDICTION USING MACHINE LEARNING”**,**

submitted by-

Prashant Narayan Dev, 501302122009, CC2880

Ankitosh Kumar, 501302122031, CC2833

Torsa Pradhan, 501302122016, CC2877

Chandan Singh, 501302122005, CC2883

Lokendra , 501302122029, CC2873

is duly examined by the undersigned as a part of the examination for the award of **MASTER OF COMPUTER APPLICATION** degree of Chhattisgarh Swami Vivekanand Technical University, Bhilai.

Internal Examiner External Examiner

Name & Signature Name & Signature

Date: Date:

**A C K N O W L E D G E M E N T**

It is a matter of profound privilege and pleasure to extend our sense of respect and deepest gratitude to our project guide **Mr. Deepak Pandey,** Department of Master of Computer Application under whose precise guidance and gracious encouragement we had the privilege to work.

We avail this opportunity to thank respected **Dr. Ajay Kushwaha,** Head of the Department of Master of Computer Application for facilitating such a pleasant environment in the department and also for providing everlasting encouragement and support throughout.

We acknowledge with the deep sense of responsibility and gratitude the help rendered byrespected **Dr. Rakesh Himte,**  Principal, Rungta College of Engineering and Technology, Bhilai for infusing endless enthusiasm & instilling a spirit of dynamism.

We would also like to thank faculty members and the supporting staff of Master of Computer Application department and the other departments in the college, for always being helpful over the years.

Last but not the least, We would like to express our deepest gratitude to our parents and the management of Rungta College of Engineering and Technology, Bhilai respected **Shri** **Santosh Ji Rungta**, Chairman, respected **Dr. Sourabh Rungta**, Director, Technical, and respected **Shri Sonal Rungta,** Director, Finance & Administration for their continuous moral support and encouragement.

We hope that we will make everybody proud of our achievements..

Prashant Narayan Dev, 501302122009, CC2880

Ankitosh Kumar, 501302122031, CC2833

Torsa Pradhan, 501302122016, CC2877

Chandan Singh, 501302122005, CC2883

Lokendra , 501302122029, CC2873

Contents

1. List of Figures 5
2. List of Tables 6
3. Introduction 7
4. Related Work 8
5. Data and Methods 13
6. Results and Discussions 22
7. Conclusion 25

**List of Figures**

Fig 2.3.1 DoS Attack

Fig 3.1.1 An illustration of the proposed solution shows the flow of the supervised learning method used in this model

Fig 3.2.1 Visualization of the dataset CICIDS2017

Fig 3.2.2 Visualization of the dataset CSE-CIC-IDS2018

Fig 3.2.3 Visualization of the dataset SDN-DDoS (ICMP, TCP, UDP) 2020

Fig 3.2.4 Representation of the Total Number of requests versus the IP address of the sender

Fig 3.2.5 Representation of the Number of Attack requests versus the IP address of the sender

Fig 4.2.1 Accuracy Rate for CICIDS2017

Fig 4.2.2 F1-Score for CICIDS2017

Fig 4.3.1 Accuracy Rate for CSE-CIC-IDS2018

Fig 4.3.1 F1-Score for CSE-CIC-IDS2018

Fig 4.4.1 Accuracy Rate for SDN-DDoS 2020

Fig 4.4.2 F1-Score for SDN-DDoS 2020

Fig 4.4.3 AUC-ROC for SDN-DDoS 2020

**List of Tables**

Table 2.3.1 A Survey of DDoS Detection Systems-1

Table 2.3.2 A Survey of DDoS Detection Systems-2

Table 2.3.3 A Survey of DDoS Detection Systems-3

Table 3.2.1 Details of DDoS Attack for CICIDS2017

Table 3.2.2 Specification of tools and duration of DDoS attack for CSE-CIC-IDS2018

Table 3.4.1 Pseudo Code for the Logistic Regression Algorithm

Table 3.4.2 Pseudo Code for the Decision Tree Algorithm

Table 3.4.3 Pseudo Code for the k-NN Algorithm

Table 3.4.4 Pseudo Code for the Random Forest Algorithm

Table 3.4.5 Pseudo Code for the Gradient Boost Algorithm

Table 4.1.1 Performance Metrics of Each Dataset

**1. ABSTRACT**

This research paper embarks on a transformative exploration of the symbiotic relationship between Artificial Intelligence (AI) and the relentless challenge posed by Distributed Denial of Service (DDoS) attacks. Recognizing the pressing need for advanced defense mechanisms in our ever-evolving digital landscape, this study is driven by the central aim of unraveling the nuanced applications of AI in DDoS mitigation.Methodologically, the research conducts an in-depth analysis of prevalent AI-driven DDoS mitigation techniques, ranging from intricate machine learning algorithms to adaptive neural networks. Through meticulous comparative assessments, this study evaluates the practical efficacy of these methodologies, scrutinizing their performance in real-world scenarios. Key factors such as detection accuracy, response time, and adaptability to dynamic attack patterns take center stage in this evaluative process.This research not only critically examines AI's pivotal role in fortifying against DDoS threats but also distills actionable insights for cybersecurity practitioners. Navigating the intricate landscape of AI-powered mitigation, the study seeks to guide decision-makers in strategically adopting advanced technologies. As a culmination, the paper unveils profound results that not only illuminate the transformative potential of AI but set a decisive course for the future resilience of cybersecurity, laying the foundation for proactive, adaptive, and robust defense strategies against the evolving threat landscape.

**Introduction**

**1.1 SaaS (Software as a Service)**

The term "cloud computing" refers to the process of storing data and using online computing resources. Nothing is stored on your computer by it. This phrase describes the on-demand accessibility of computer services including servers, databases, networking, and data storage. Cloud computing's main objective is to make data centers more widely accessible to users. Users can also access data on a faraway server.

In cloud computing, there are three types of service models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

SaaS is a hosted software model which allows end-users to utilize the software and apps/web-based applications on the cloud via the internet without the need to install them on his/her machine. By utilizing SaaS, consumers may pay a per-user subscription cost without having to spend a significant sum of money on the installation and upkeep of required software and hardware. As long as they have a device with internet connectivity, users can access the service from anywhere in the globe at any time. Without having to worry about updates, users are always using the most recent version of the program.

The service provider is in charge of managing the cloud's infrastructure. Together with ensuring the security of applications and customers' access to them, the service provider also controls software and service agreements. Companies are driven to concentrate on the development of their software by recurring income. SaaS's inherent multi-tenancy and virtualization technologies give service providers centralized control and maximum resource usage. Examples of SaaS include email and office software.

**1.2 SaaS Security Challenges**

Despite these advantages, the security of the SaaS presents a great problem for companies, on the level of Confidentiality, integrity, authentication, location of data, etc. The major challenge that a company or user encounters when converting their software to a SaaS architecture is the lack of confidence in the security of their data owing to the worry of data leakage. Customers of cloud services are not in charge of managing the cloud infrastructure. The user should be aware of the security precautions the cloud provider has put in place.

The data security of the SaaS (Software as a Service) environment is a complex issue that covers several areas such as confidentiality, availability, integrity, authorization, backup and recovery, and transfer. Several solutions have been proposed, such as using SSL encryption, MAC, searchable encryption, homomorphic tokens, and flexible security policies. However, these solutions are limited by factors such as false certificate authorities, data error location, limited applicability to SQL-like databases, and data availability.

Similarly, application security is also a critical issue that has been extensively studied, but fine-grained solutions to tackle these issues are limited. User authentication is a crucial part of application security, and various approaches have been proposed, such as user identity verification, biometric cues-based authentication, behavior-based authentication, touch screen pattern, keystroke analysis, and implicit authentication. However, these approaches are limited by accuracy, training time, and the impact of different factors such as footwear and soft input methods.

**2 Related Work**

**2.1 Attacks and their possible countermeasures in SaaS**

Cyber Attacks on SaaS-based cloud computing services are a growing concern in today’s digital landscape. In this model, as the software and associated data are stored on servers maintained by the service provider, and accessed remotely by users, it makes an attractive target for cyber attackers, who can potentially access a large number of sensitive information and systems in a single attack.

There are now 28 assaults on SaaS, which is this problem. It has to be made clear that not all assaults are possible with this service paradigm.

**2.2 Our Main Focus**

After carefully examining numerous SaaS-based issues, we decided to focus on DDoS attacks. This choice was made after analyzing several factors, including the rising frequency and severity of DDoS attacks, and the potential negative effects these attacks could have on SaaS applications and services, As enterprises and individuals increasingly rely on SaaS-based technologies, we also took this into account. To guarantee the availability and security of these services, we needed solid, scalable solutions. We are aiming to make a fast and efficient machine learning solution that can aid in defending SaaS applications and services against this evolving danger by concentrating on DDoS attack mitigation.

**2.3 Detection of DDoS Attacks in Cloud-based Services Using Machine Learning Techniques**

**2.3.1 DDoS Attack**

A Distributed Denial of Service (DDoS) attack is a type of cyberattack in which an attacker floods the target system with traffic, making it unreachable to normal users. DDoS attacks on cloud computing services based on software as a service (SaaS) can target the service provider's infrastructure, interrupting access to the SaaS application for all customers.

Severe downtime, slow response times, service interruptions, financial loss, reputational harm, and complete outages are examples of DDoS attack symptoms that can have a big impact on a company or organization.

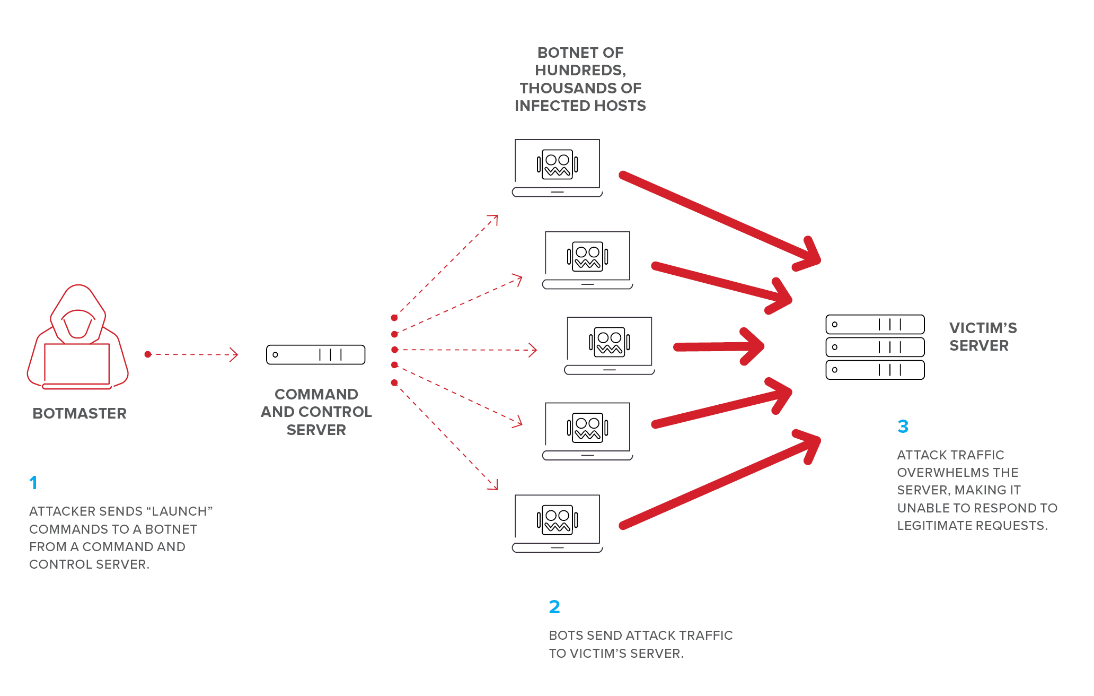


Fig 2.3.1 DoS Attack

source: F5 Networks

Note: The above-shown example is not a distributed attack, since it only targets one organization

DDoS attacks can be conducted by individuals or groups with a variety of motives, including extortion, activism, or personal revenge. Attackers may use more complex methods to avoid detection and magnify the attack. DDoS assaults can be launched using a variety of approaches, including botnets, amplification attacks, and application-layer attacks.

**2.3.2 Problem Statement**

The growing popularity of cloud computing services based on Software as a Service (SaaS) has made them an appealing target for cyber assaults, notably Distributed Denial of Service (DDoS) attacks. DDoS assaults may create major disruptions and financial losses for both service providers and users. The difficult part is figuring out the attack, reducing its impact, and putting precautions in place to stop it from happening again.

Therefore, there is an urgent need to create reliable and strong detection techniques for DDoS assaults in cloud computing services based on SaaS.

**2.3.3 Why is it critical to resolve this issue?**

**Stakeholder Impact:** A DDoS assault might cause significant interruptions to customers, workers, partners, and vendors. When a website or service goes down, it can have an impact on sales and the brand's reputation. Both national security and public safety can be harmed by critical infrastructure or government services.

**Problem-related risks:** DDoS assaults are becoming more common, as are the dangers associated with them. Attackers are becoming more competent, and attacks are becoming larger and longer in duration. Organizations that do not take adequate protection against DDoS assaults suffer several issues.

**Consequences of failing to take appropriate steps to combat DDoS attacks:** The repercussions of ignorance can be severe, including lost income, brand image damage, and cleaning and recovery costs. Secondary costs connected with outages and interruptions may include diminished personnel productivity and lost business opportunities.

**2.3.4 Proposed Solution**

Machine learning models have demonstrated the potential in recognizing abnormal network traffic patterns, which might be used to detect and mitigate DDoS assaults. The detection performance of machine learning models, on the other hand, is dependent on the quality of the training data and the features utilized for classification. Therefore, we propose to develop a robust machine learning-based approach that can accurately detect DDoS attacks in SaaS-based cloud computing services by leveraging appropriate features and training data.

**3 Data and Methods**

**3.1 Overview of Implementation and Design of the Model**

Firstly, DDoS raw data is collected from open sources. A total of three datasets are collected. After collection, data is processed to construct the final datasets for modeling. Data processing includes data cleaning, the transformation of data types, and dataset splitting. This is followed by the model selection process. The models are trained with all tree datasets using five different algorithms; logistic regression, k-nearest neighbor, decision tree, random forest, and gradient boost. Finally, the model is tested with unseen data. The results are evaluated using several performance metrics.

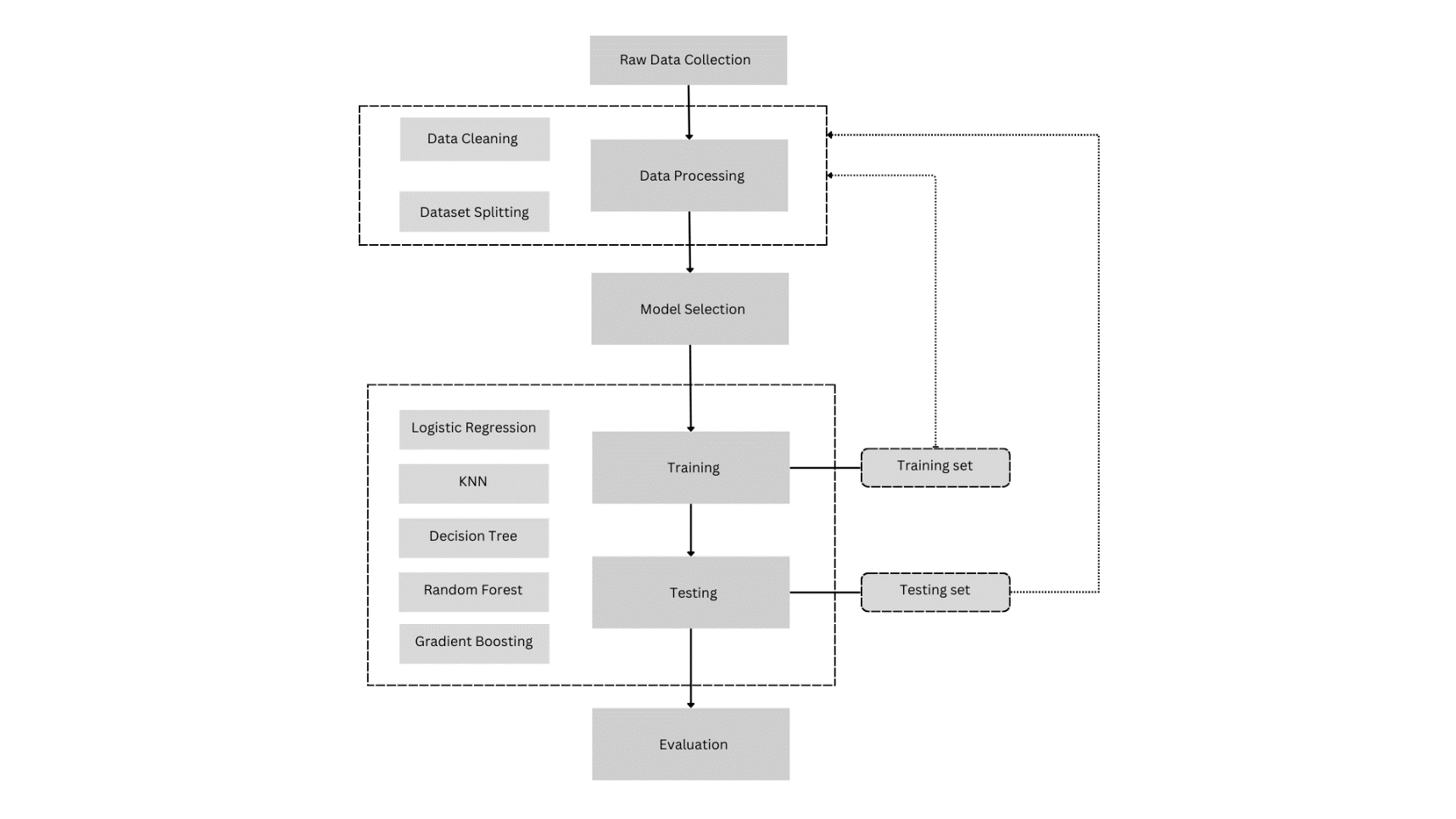


Fig 3.1.1 An illustration of the proposed solution shows the flow of the supervised learning method used in this model

**3.2 Datasets**

**3.2.1 CICIDS2017**

The CICIDS2017 dataset is the result of a 5-day simulation from Monday to Friday, and it includes network traffic in both packet and bidirectional flow forms. It is worth mentioning that for every one of the flows, 80 characteristics were gathered, including more information about the IP addresses and attacks of the simulated multiple attackers. Within usual limits, scripts are used to mimic default and user behavior. The first day is considered typical, and traffic is only moderate. The simulated assaults used DDoS data, and the attack scenario was acquired from CICIDS 2017.

|  |  |  |
| --- | --- | --- |
| Assault Situation | Target | Intruder IP |
| DDoS Low Orbit Ion Cannon (LOIC) | Ubuntu 16 205.174.165.68 | 205.174.165.69 |
| 205.174.165.71 |
| 205.174.165.70 |

Table 3.2.1 Details of DDoS Attack for CICIDS2017

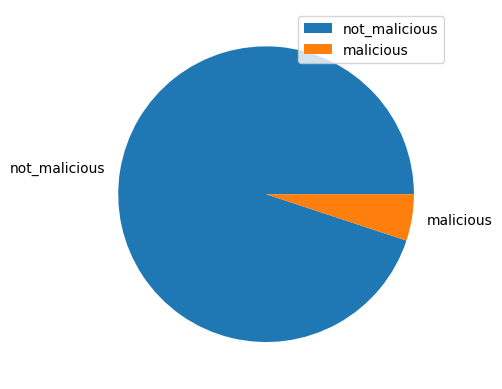


Fig 3.2.1 Visualization of the dataset CICIDS2017

**3.2.2 CSE-CIC-IDS2018**

The CSECIC2018 dataset is a network traffic dataset derived from a 24-hour online traffic trace on Wednesday. The dataset comprises 44 characteristics for each flow and includes both packet and flow-based data.

The dataset was compiled using the open-source program SiLK and contains both legitimate and malicious traffic. Malicious traffic encompasses a variety of assaults, including Distributed Denial of Service (DDoS), scanning, and brute-force attacks.

The data was gathered in a controlled environment, and scripts were utilized to mimic typical user and attacker behavior.

|  |  |  |  |
| --- | --- | --- | --- |
| Tools | Period | Intruder | Target |
| DDOS Low Orbit Ion Canon (LOIC) for HTTP, UDP, or TCP requests | 24 hours | Kali Linux | Windows Vista 7, 8.1, 10 (32-bit), and 10 (64-bit) |

Table 3.2.2 Details of DDoS Attack for CSE-CIC-IDS2018

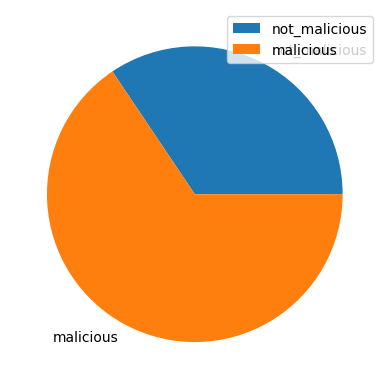


Fig 3.2.2 Visualization of the dataset CSE-CIC-IDS2018

**3.2.3 SDN-DDoS (ICMP, TCP, UDP) 2020**

The SDN-DDoS (ICMP, TCP, UDP), 2020 dataset is made up of network traffic created by a DDoS assault simulation. The simulation was run with scripts that mimicked default and user behavior within usual restrictions. The dataset comprises 10,000 flows of traffic for ICMP, TCP, and UDP assaults. Each flow has 17 characteristics, such as source and destination IP protocol types, addresses, and port numbers. The dataset is intended to aid in the development and testing of machine learning algorithms for DDoS detection in Software Defined Networking (SDN) settings.

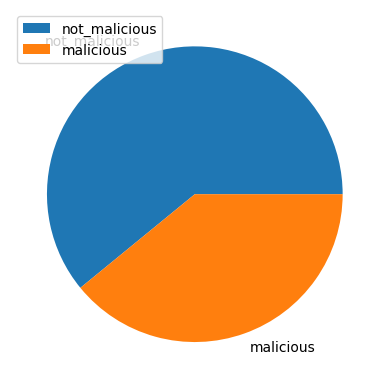


Fig 3.2.3 Visualization of the dataset SDN-DDoS (ICMP, TCP, UDP) 2020

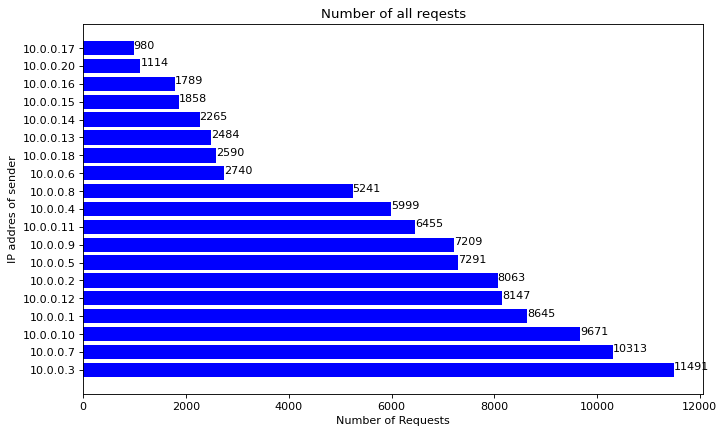


Fig 3.2.4 Representation of the Total Number of requests versus the IP address of the sender

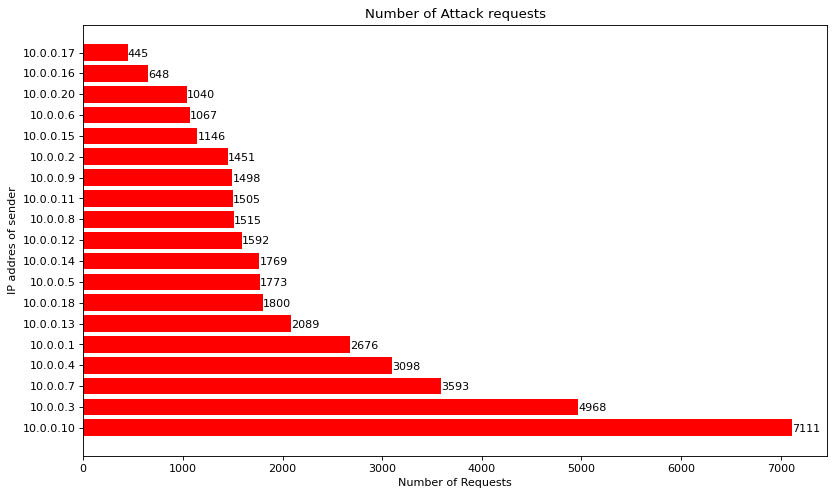


Fig 3.2.5 Representation of the Number of Attack requests versus the IP address of the sender

**3.3 Data Preprocessing**

We have labeled all the ‘DDOS’ attacks as 1 and all other attacks or intrusions as 0. Since our datasets have many redundant features we removed them as those features make it difficult to understand and analyze the dataset. Also, we have preprocessed our dataset after visualization to reduce any kind of Overfitting of the models.

We have identified and removed all the irrelevant features as it increases complexity and decreases the performance of the model, removing redundant features decreases the computational resources as well as computation time. So, for CICIDS – 2017,2018, the number of features or columns used for training has been reduced from 78 to 39, and for the SDN dataset we have reduced the features from 23 to 16, This reduction in the feature set is a result of the preprocessing operations, and it increases the performance and overall computation time of our model.

**3.4 Model Selection**

Once the data is processed, the next step involves selecting the most appropriate algorithm for DDoS detection and classification. The chosen algorithms, including Logistic Regression, K-Nearest Neighbors (KNN), Decision Tree, Random Forest, and Gradient Boosting, are specifically tailored for their suitability in detecting DDoS attacks. The selection process takes into account factors such as the algorithms' performance, computational requirements, interpretability, and robustness.

**3.4.1 Logistic Regression**

Logistic regression is a statistical analysis technique that is specifically designed for situations where the dependent variable is binary. It helps in understanding the relationship between a binary outcome and other independent variables that can have non-binary values.

Table 3.4.1 Pseudo Code for the Logistic Regression Algorithm

Algorithm 1 Logistic Regression

given α, {(x i , y i )}

step 1: initialize a = <1,……,1>T.

step 2: perform feature scaling on the examples’ attributes

repeat until convergence.

Step 3: for each j = 0, .., n:

a`j = aj+ α Σi(yi − ha(xi ))Xj^i

for each j = 0, .., n:

aj = aj, output a

**3.4.2 Decision Tree**

The root node of a decision tree is used to evaluate observations and classify them based on their associated attributes. Each node in the tree represents a distinct property, with possible values corresponding to various categories or outcomes.

Table 3.4.2 Pseudo Code for the Decision Tree Algorithm

Algorithm 2 Decision Tree

∀ attributes a1, a2, …, an.

Step 1: Find the attribute that best divides the training data using the information gained.

Step 2: a\_best ← the attribute with the highest information gain.

Step 3: Create a decision node that splits on a\_best.

Step 4: Recursively add the nodes from the sub-lists created by splitting on a\_best to the node.

The approach estimates the information gained for each feature in the training dataset starting from the root node. The discriminatory strength of features in distinguishing between target groups is measured by information gain. Higher information gain features are more useful for effectively identifying observations. The root node is then replaced with the feature that provides the most information gain, and the algorithm continues to partition the dataset into subsets based on the feature that was chosen.

**3.4.3 K-Nearest Neighbours (KNN)**

It is a classifier based on instances. When the k-NN algorithm is used, examples in a dataset are stored in a dimensional space, and new instances are tagged based on their similarity to existing instances. These are referred to as neighbors. If x is the most similar class for the surrounding observations, a new instance of x is created. A distance function is used to determine the similarity of two instances. The distance function utilized in this model is Euclidean.

Table 3.4.3 Pseudo Code for the k-NN Algorithm

Algorithm 3 K-Nearest Neighbours

Step 1: Let S = {a1, a2, …, an}, where S represents the training set and a represents article documents

Step 2: k ← the desired number of nearest neighbors

Step 3: Compute d(x,y) between new instance i and all a ∈ S

Step 4: Select the k closest training samples to i

Step 5: Classi ← best voted class end

**3.4.4 Random Forest**

Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. It improves the accuracy and robustness of the model by reducing overfitting and increasing generalization.

Table 3.4.4 Pseudo Code for the Random Forest Algorithm

Algorithm 4 Random Forest

Require IDT (a decision tree inducer), T (the number of iterations), S (the training set), µ (the subsample size), N (the number of attributes used in each node)

t ← 1

repeat

St ← Sample µ instances from S with replacement.

Build classifier Mt using IDT(N) on St

t++

until t > T

**3.4.5 Gradient Boosting**

Gradient Boosting is another ensemble method that builds an additive model by sequentially adding weak learners, such as decision trees, to correct the mistakes made by previous models. It is known for its high predictive power and ability to handle complex relationships in the data.

Table 3.4.5 Pseudo Code for the Gradient Boost Algorithm

Algorithm 5 Gradient Boost

Step 1: Initialize the residuals to be the target variable.

Step 2: Initialize the model to be the mean of the target Variable.

Step 3: Loop over the number of trees to train.

a. Train a decision tree on the residuals.

b. Use the decision tree to predict the residuals.

c. Update the model by adding a fraction of the prediction multiplied by learning rate.

d. Update the residuals by subtracting the prediction from the target variable.

Return the final model.

**3.5 Training and Testing**

After the algorithm selection, the datasets are divided into a training set and a testing set. This division allows for the evaluation of the model's performance on unseen data, providing a realistic assessment of its effectiveness in real-world scenarios. The models are then trained using the training set and subsequently tested using the testing set to measure their ability to accurately identify and classify DDoS attacks.

For the SDN-DDoS2020 dataset, we have split the dataset as 80:20 for training and testing respectively. And for CICIDS2017 and CSE-CIC-IDS2018 we have split the dataset as 70:30 for training and testing respectively.

**3.6 Evaluation**

Following the training and testing phase, the trained models are evaluated using various performance metrics, including Accuracy, Precision, Recall, F1-score, and AUC-ROC.

**3.6.1 Accuracy**

Accuracy measures the overall correctness of the model's predictions and is defined as the ratio of the number of correct predictions to the total number of predictions.

Accuracy = (Number of Correct Predictions) / (Total Number of Predictions)

**3.6.2 Precision**

Precision measures the proportion of correctly predicted positive instances (true positives) out of all instances predicted as positive (both true positives and false positives).

Precision = (True Positives) / (True Positives + False Positives)

**3.6.3 Recall (Sensitivity or True Positive Rate)**

Recall measures the proportion of correctly predicted positive instances (true positives) out of all actual positive instances (true positives and false negatives).

Recall = (True Positives) / (True Positives + False Negatives)

**3.6.4 F1 Score**

The F1 score combines precision and recall into a single metric and is useful when you want to find a balance between precision and recall. It is the harmonic mean of precision and recall.

F1 Score = 2 \* (Precision \* Recall) / (Precision + Recall)

**3.6.5 AUC-ROC**

AUC-ROC is a metric used for binary classification problems. It measures the model's ability to discriminate between positive and negative instances.

It calculates the area under the Receiver Operating Characteristic (ROC) curve, which shows the trade-off between the true positive rate (TPR) and false positive rate (FPR) at different classification thresholds. The ROC curve plots TPR against FPR, and the AUC-ROC represents the area under this curve. A higher AUC-ROC value indicates better model performance.

These metrics provide valuable insights into the models' overall effectiveness in detecting DDoS attacks and help in assessing their performance across different evaluation criteria. By comparing the performance of different algorithms and analyzing the evaluation results, a detailed report can be generated.

This report provides valuable information on the effectiveness of each algorithm in detecting DDoS attacks, enabling informed decisions on the selection of the most appropriate algorithm for future DDoS detection efforts.

**4 Results and Discussion**

**4.1 Overview**

The goal of this evaluation is to analyze the performance of the different DDoS datasets in terms of their capacity to detect intrusion (via a DDoS attack). From the results, we can conclude that all three dataset performance is best overall, achieving an accuracy rate of 99% across all models, and an F1-Score of 99%, denoting that a model trained with all three datasets are performing well, as it is correctly predicting threats (precision) and captures all relevant cases of malicious traffic (recall) at a 99% rate across all models.

From a model point of view Random Forest, Decision Tree, and Gradient Boost are performing best for all the datasets (CICIDS-2017,2018 and SDN dataset) using this model we are achieving accuracy up to 99%, and we have also achieved precision and F1-score of 100%. Our Logistic Regression performance is the lowest among all datasets, there the lowest accuracy we are achieving is 59%, an F1-Score of 42%, and a precision of 47% using the SDN dataset. So, using all these models we are achieving decent evaluation metrics(accuracy, precision, F1-score).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Datasets | Evaluation Parameters | Logistic Regression | K-NN | Random Forest | Decision Tree | Gradient Boost |
| CICIDS2017 | Accuracy | 0.989 | 0.999 | 0.999 | 0.999 | 0.999 |
| Precision | 0.979 | 0.999 | 0.999 | 0.999 | 0.999 |
| Recall | 0.998 | 0.999 | 0.999 | 0.999 | 0.999 |
| F-Measure | 0.989 | 0.999 | 0.999 | 0.999 | 0.999 |
| Computation Time | 9.446 sec | 0.044 sec | 26.951 sec | 1.351 sec | 80.521 sec |
| CSE-CIC-IDS2018 | Accuracy | 0.999 | 0.999 | 0.999 | 0.999 | 1.000 |
| Precision | 0.998 | 0.999 | 0.999 | 0.999 | 1.000 |
| Recall | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| F-Measure | 0.999 | 0.999 | 0.999 | 0.999 | 1.000 |
| Computation Time | 9.017 sec | 0.031 sec | 4.333 sec | 0.304 sec | 36.978 sec |
| SDN DDoS 2020 | Accuracy | 0.596 | 0.891 | 1.000 | 1.000 | 0.993 |
| Precision | 0.477 | 0.887 | 1.000 | 1.000 | 0.987 |
| Recall | 0.378 | 0.825 | 1.000 | 1.000 | 0.996 |
| F-Measure | 0.422 | 0.855 | 1.000 | 1.000 | 0.992 |
| Computation Time | 0.570 sec | 0.020 sec | 12.337 sec | 0.479 sec | 26.863 sec |

Table 4.1.1 Performance Metrics of Each Dataset

**4.2 CICIDS2017**

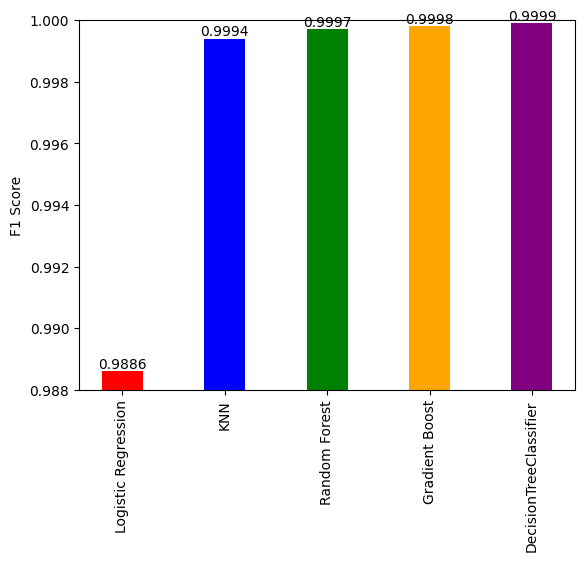
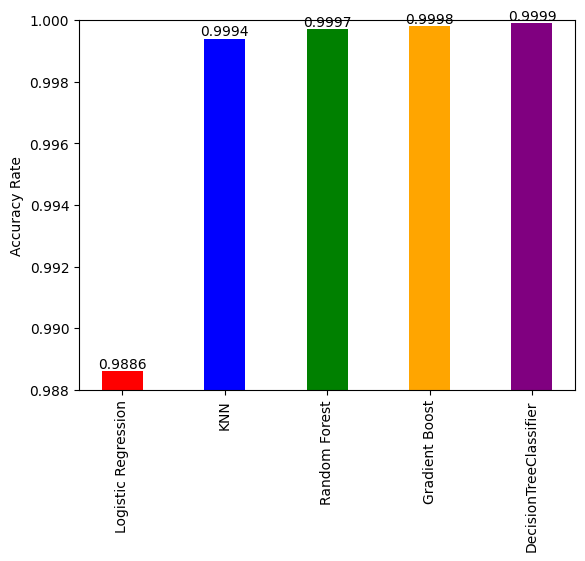
  
 Fig 4.2.1 Fig4.2.2

Fig 4.2.1 Accuracy Rate for CICIDS2017

Fig 4.2.2 F1-Score for CICIDS2017

**4.3** **CSE-CIC-IDS2018**

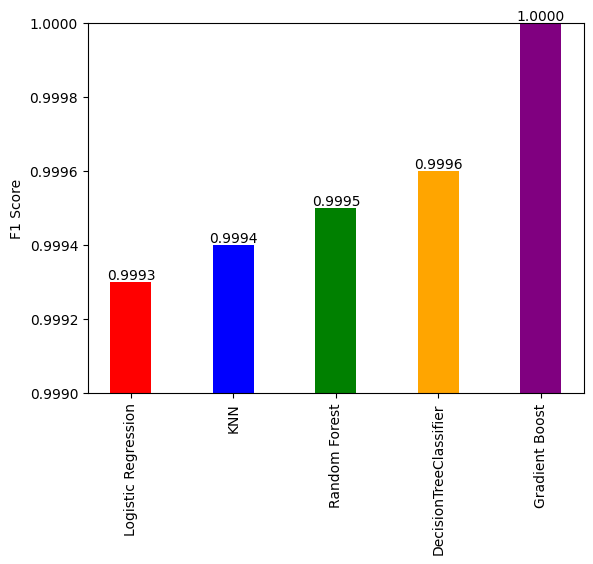
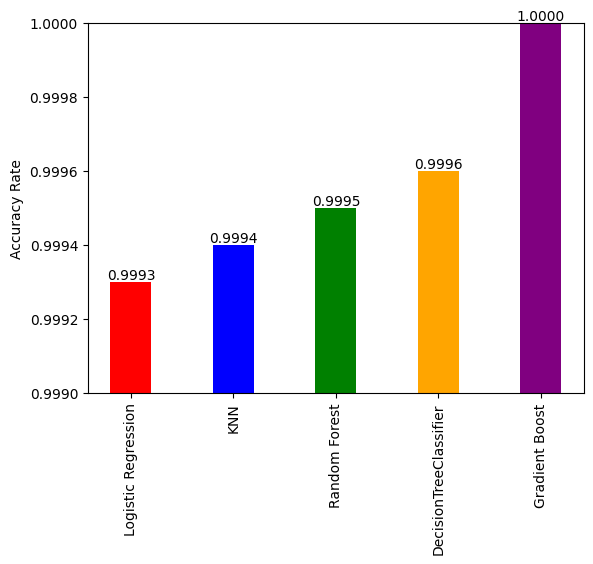


Fig 4.3.1 Fig4.3.2

Fig 4.3.1 Accuracy Rate for CSE-CIC-IDS2018

Fig 4.3.1 F1-Score for CSE-CIC-IDS2018

**4.4 AUC-ROC for SDN-DDoS (ICMP, TCP, UDP) 2020**

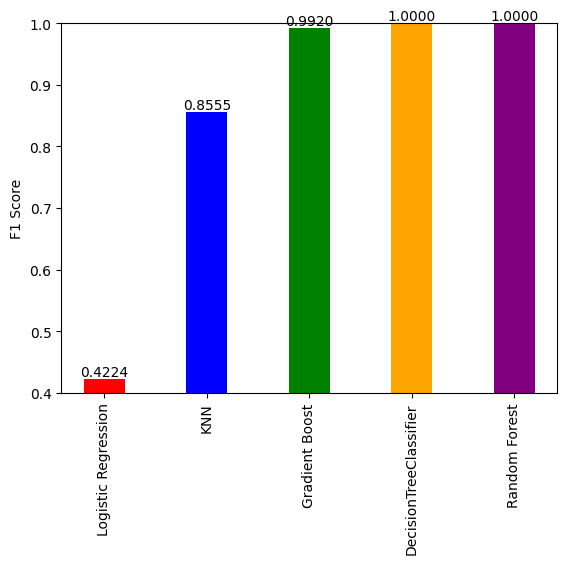
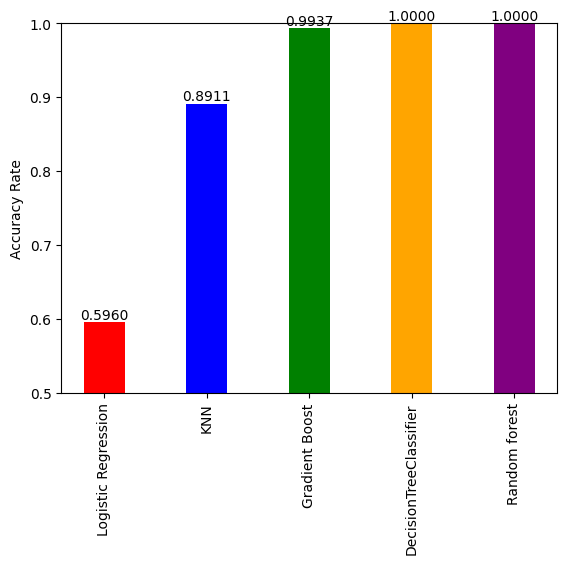


Fig 4.4.1 Fig 4.4.2

Fig 4.4.1 Accuracy Rate for SDN-DDoS 2020

Fig 4.4.2 F1-Score for SDN-DDoS 2020

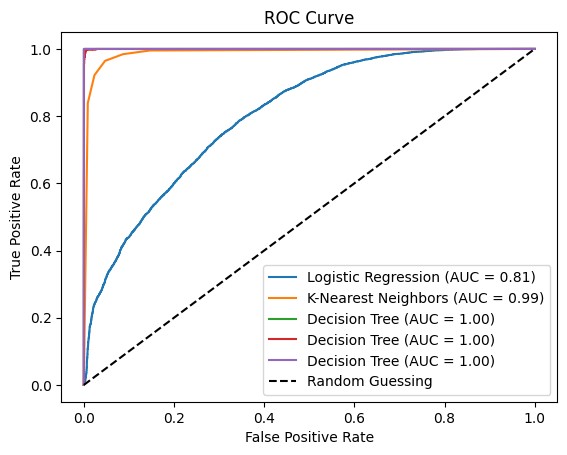


Fig 4.4.3 AUC-ROC for SDN-DDoS 2020 **5 Conclusion**

Firstly, we have discussed the SaaS services and the security challenges in SaaS Services. In the related work, the first part is the review of attacks and their possible countermeasures in SaaS Services. After conducting this, we have decided to focus on DDoS attacks due to the rising frequency, the potential negative effect, and the severity of these attacks on SaaS applications and services. So the second part discusses the survey of recently developed DDoS Detection Systems on some well-known datasets.

Our Report proposes an approach for detecting DDoS attacks in cloud environments through the utilization of machine learning techniques. We have developed a robust DDoS attack detection system. Our evaluation is based on three prominent datasets CICIDS2017, CSE-CIC-IDS2018, and SDN-DDoS (ICMP, TCP, UDP) 2020. The models we made are trained with all tree datasets using five different algorithms; logistic regression, k-nearest neighbor, decision tree, random forest, and gradient boost.

The utilization of multiple datasets, including CICIDS2017, CSE-CIC-IDS2018, and SDN-DDoS (ICMP, TCP, UDP) 2020 provides a comprehensive evaluation of our system's performance across various attack scenarios. This enhances the reliability and generalizability of our findings, reinforcing the effectiveness of our proposed solution. By using logistic regression, KNN, Random Forest, Decision tree, and Gradient Boost machine learning algorithms our model can effectively analyze network traffic patterns, identifying anomalies associated with DDoS attacks. Among these; Random Forest, Decision Tree, and Gradient Boost are performing the best and Logistic Regression performance is the lowest for all the datasets using this model.

Through our experiments, we have achieved highly promising results in accuracy rates, with performance exceeding 99%. This demonstrates the effectiveness of our proposed system in accurately identifying and preventing DDoS attacks within cloud environments.

In summary, our research contributes to DDoS attack prevention in cloud environments by introducing a machine learning-based detection system. With accuracy rates surpassing 99% and utilizing diverse datasets, our approach proves to be highly reliable and capable of providing robust security measures against DDoS attacks at the source side in the cloud.

**5.1 Further Works**

After training all the models using all the datasets, we have made a web app using the Python libraries such as Streamlit and Pickle, using the web app any user who feels his system is under any kind of DDOS attack can use the web app to determine whether the traffic was malicious or non-malicious. The user gives captures the web traffic from wireshark or some other tools and gives that file to our app where it pulls the data from the file and determines the result.

In the future, we are open to making a Chrome extension based on this idea, which actively observes the web traffic and determines the assault.

**Bibliography**

1. Díaz de León Guillén, Miguel Ángel, Morales-Rocha, Víctor, and Fernández Martínez, Luis Felipe. ‘A Systematic Review of Security Threats and Countermeasures in SaaS’. 1 Jan. 2020 : 635 – 653.

2. Mohamad Mulham Belal, Divya Meena Sundaram, Comprehensive review on intelligent security defenses in the cloud: Taxonomy, security issues, ML/DL techniques, challenges and future trends, Journal of King Saud University - Computer and Information Sciences, Volume 34, Issue 10, Part B, 2022, Pages 9102-9131, ISSN 1319-1578, https://doi.org/10.1016/j.jksuci.2022.08.035.

3. Thabit, Fursan & Al-ahdal, Abdulrazzaq & Alhomdy, Sharaf & Jagtap, Sudhir. (2020). Exploration of Security Challenges in Cloud Computing: Issues, Threats, and Attacks with their Alleviating Techniques. Journal of Information and Computational Science. 10. 35-59.

4. Sail, Soufiane & Halima, Bouden. (2017). SaaS Cloud Security: Attacks and Proposed Solutions. Transactions on Machine Learning and Artificial Intelligence. 5. 10.14738/tmlai.54.3194.

5. Bachelor’s Thesis, Bachelor of Engineering, Information and Communication Technology, Turku University of Applied Sciences, Prabin Pandey, Security Attacks in Cloud Computing.

6. Chouhan, Pushpinder & Yao, Feng & Yerima, Suleiman & Sezer, Sakir. (2015). Software as a Service: Analyzing Security Issues.

7. S. Alam, M. Muqeem and S.A. Khan, Review on security aspects for cloud architecture, International Journal of Electrical and Computer Engineering 8(5) (2018), 3129–3139.

8. V. Casola, A. De Benedictis, M. Rak and E. Rios, Security-by-design in clouds: A security-SLA driven methodology to build secure cloud applications, Procedia Computer Science 97 (2016), 53–62. doi:10.1016/j.procs.2016.08.280.

9. R. Charanya, M. Aramudhan, K. Mohan and S. Nithya, Levels of security issues in cloud computing, International Journal of Engineering and Technology 5(2) (2013), 1912–1920.

10. L. Coppolino, S. D’Antonio, G. Mazzeo and L. Romano, Cloud security: Emerging threats and current solutions, Computers & Electrical Engineering 59 (2017), 126–140. doi:10.1016/j.compeleceng.2016.03.004.

11. P. Derbeko, S. Dolev, E. Gudes and S. Sharma Security and privacy aspects in MapReduce on clouds: A survey, Computer Science Review 20 (2016), 1–28. doi:10.1016/j.cosrev.2016.05.001.

12. J.B. Hong, A. Nhlabatsi, D.S. Kim, A. Hussein, N. Fetais and K.M. Khan, Systematic identification of threats in the cloud: A survey, Computer Networks 150 (2019), 46–69. doi:10.1016/j.comnet.2018.12.009.

13 S. Iqbal, M.L.M. Kiah, B. Dhaghighi, M. Hussain, S. Khan, M.K. Khan, and K.-K.R. Choo, On cloud security attacks: A taxonomy and intrusion detection and prevention as a service, Journal of Network and Computer Applications 74 (2016), 98–120. doi:10.1016/j.jnca.2016.08.016.

14 J. Jang-Jaccard and S. Nepal, A survey of emerging threats in cybersecurity, Journal of Computer and System Sciences 80(5) (2014), 973–993. doi:10.1016/j.jcss.2014.02.005.

15 A. Joshi, S.T. King, G.W. Dunlap, and P.M. Chen, Detecting past and present intrusions through vulnerability-specific predicates, in ACM SIGOPS Operating Systems Review, Vol. 39, ACM, 2005, pp. 91–104.

16. M.A. Khan, A survey of security issues for cloud computing, Journal of Network and Computer Applications 71 (2016), 11–29. doi:10.1016/j.jnca.2016.05.010.

17. N. Khan and A. Al-Yasiri, Identifying cloud security threats to strengthen cloud computing adoption framework, Procedia Computer Science 94 (2016), 485–490. doi:10.1016/j.procs.2016.08.075.

18. K. Krombholz, H. Hobel, M. Huber and E. Weippl, Advanced social engineering attacks, Journal of Information Security and Applications 22 (2015), 113–122. doi:10.1016/j.jisa.2014.09.005.

19. P.R. Kumar, P.H. Raj, and P. Jelciana Exploring data security issues and solutions in cloud computing, Procedia Computer Science 125 (2018), 691–697. doi:10.1016/j.procs.2017.12.089.

20. P. Mishra, E.S. Pilli, V. Varadharajan and U. Tupakula, Intrusion detection techniques in a cloud environment: A survey, Journal of Network and Computer Applications 77 (2017), 18–47. doi:10.1016/j.jnca.2016.10.015.

21. C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel, and M. Rajarajan, A survey of intrusion detection techniques in the cloud, Journal of Network and Computer Applications 36(1) (2013), 42–57. doi:10.1016/j.jnca.2012.05.003.

22. M. Mulazzani, S. Schrittwieser, M. Leithner, M. Huber and E. Weippl, Dark clouds on the horizon: Using cloud storage as an attack vector and online slack space, 2011.

23. A. Philpott, Identity theft – Dodging the own-goals, Network Security 2006(1) (2006), 11–13. doi:10.1016/S1353-4858(06)70323-3.

24. A. Singh and K. Chatterjee, Cloud security issues and challenges: A survey, Journal of Network and Computer Applications 79 (2017), 88–115. doi:10.1016/j.jnca.2016.11.027.

25. G. Somani, M.S. Gaur, D. Sanghi, M. Conti and R. Buyya, DDoS attacks in cloud computing: Issues, taxonomy, and future directions, Computer Communications 107 (2017), 30–48. doi:10.1016/j.comcom.2017.03.010.

26. N. Srinivasu, O.S. Priyanka, M. Prudhvi and G. Meghana, Multilevel classification of security threats in cloud computing, International Journal of Engineering and Technology (UAE) 7(1.5) (2018), 253–257.

27. A. Vasudeva and M. Sood, Survey on sybil attack defense mechanisms in wireless ad hoc networks, Journal of Network and Computer Applications 120 (2018), 78–118. doi:10.1016/j.jnca.2018.07.006.

28. M. Wu and Y.B. Moon, Taxonomy of cross-domain attacks on CyberManufacturing system, Procedia Computer Science 114 (2017), 367–374. doi:10.1016/j.procs.2017.09.050.