National College of Ireland

SECURE DEPLOYMENT OF CLOUD INTEGRATED CYBERSECURITY APPLICATIONS: A COMPREHENSIVE CLOUD SECURITY MODEL

MSc Academic Internship
MSc Cyber Security

Anjali Pappachan Mulloly
Student ID: 21218897

School of Computing
National College of Ireland

Supervisor: Eugene Mclaughlin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Anjali Pappachan Mulloly

Student ID: 21218897

Programme: MSc Cyber Security **Year:** 2023-2024

Module: MSc Academic Internship

Supervisor: Eugene Mclaughlin

Submission Due Date: 30/01/2024

Project Title: SECURE DEPLOYMENT OF CLOUD INTEGRATED CYBERSECURITY APPLICATIONS: A COMPREHENSIVE CLOUD SECURITY MODEL

Word Count:4708 **Page Count** 16

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use another author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Anjali Pappachan Mulloly

Date: 30/01/2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or misplaced. It is not sufficient to keep a copy on a computer.	□

Assignments that are submitted to the Programme Coordinator's Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

SECURE DEPLOYMENT OF CLOUD INTEGRATED CYBERSECURITY APPLICATIONS: A COMPREHENSIVE CLOUD SECURITY MODEL

Anjali Pappachan Mulloly
Student ID - 21218897

1 Abstract

The current research is built to analyze and detect the buffer overflow attack on mail using the cloud-integrated cyber security model. However, the model also assesses datasets, identifies buffer overflow incidents and provides accuracy for email-based attacks. Mainly, the use of machine learning techniques is considered an intriguing way of addressing these cloud security challenges. These algorithms implement pattern recognition, anomaly detection, and predictive analytics to analyze and detect any threats. Many difficulties were encountered when developing the model, including integrating AWS with machine learning. These issues were resolved, although the model could only function once the secret key was created and applied to it. This model will establish a connection with AWS at that point. In addition to this, the main objective of the research was to implement machine learning algorithms for detecting buffer overflow attacks over the mail and then integrate the detection results into a cloud-based cybersecurity model that is connected with AWS. Furthermore, the methodology focuses on gathering data sets from CVE details and online portals. Data processing also assists in eliminating the special character and null values and sklearn. Feature selection, validation and training of the model, and AWS performance have been included. It makes use of an integrated machine learning-based model that aids in the analysis of the buffer overflow assault that took place and contributed to the problems with hostile activity. Datasets have been utilized in this model to examine buffer overflows on CSM mail. This will first examine the buffer overflow on the CSM mail, and if the model is successful in obtaining an accuracy of roughly 90% precision in detecting the buffer overflow, it will then forward the result to the cloud-integrated architecture, which aids in demonstrating the outcome that the network attack is taking place. If so, this indicates that the model can identify cloud buffer overflow attacks. The study has been answering research questions related to the particular security needs of different cloud-integrated cybersecurity apps, considering issues. Additionally, suggested CSM, along with current cloud security models and frameworks, includes its benefits and distinguishing features to resolve the security problems of cloud-integrated cybersecurity applications.

2 Introduction

2.1 Background

As cloud computing grows at an exponential rate and as organizations depend more and more on cloud services, ensuring proper cloud security needs to be a top concern. Rule-based procedures are used by security devices and intrusion detection systems; these procedures are effective in around 80% of cases, but they must be updated frequently to keep up with hostile individuals and fraudsters, who may create new attack routes. This limitation arises from the massive amount of data generated in cloud systems, necessitating a more adaptable and

efficient security solution. Using machine learning techniques is an intriguing way to address these cloud security challenges. These algorithms use pattern recognition, anomaly detection, and predictive analytics to find and eliminate any threats. An in-depth understanding of the cloud computing context could be generated by machine learning algorithms through system analysis. Machine learning algorithms may be able to generate a comprehensive understanding of the cloud computing context by analyzing system logs, network traffic logs, historical data, and user actions. With this information, they can identify anomalies, foresee hostile behavior or vulnerabilities, classify various risks, and take care of any security issues. The benefits of employing machine learning to detect risks in cloud security are then discussed. One of the key advantages is the ability to identify threats in real-time. Because machine learning models are so good at processing large volumes of data quickly, threats can be recognized as soon as they appear.

2.2 Aim

The project aims to analyze and detect the buffer overflow attack on the mail using the cloud-integrated cyber security model. The model assesses datasets, identifies buffer overflow incidents and provides accuracy for mail-based attacks.

2.3 Objective

The project combines an AWS-connected cloud model with the detection findings of buffer overflows using machine learning. The primary objective of this project is to employ machine learning algorithms for the purpose of detecting buffer overflow attacks on mail and subsequently integrating the detection results into a cloud-based cybersecurity model connected to AWS.

2.4 Research Gap

After performing the comprehensive literature review, I found that there is much research conducted in the field of cybersecurity addressing one or the other issues related to different vulnerabilities or malicious behavior. Here, the researcher identified one such vulnerability that is uploaded on a CVE vulnerability website, which posts different security issues faced by CMS. From there, it identified the vulnerability" Buffer overflow in CSM mail server allows remote attackers to cause a denial of service or execute commands via a long HELO command," on which no one has posted any solution or did work to resolve the issue. After identifying the issues, a literature review is performed to get an idea about the research work done on buffer, and it is noted that not much work has been done on this vulnerability. It is noted that all the existing literature lacks comprehensive insight into buffer overflow detection and the integration of machine learning with cloud AWS. This project fills the gaps by detailing how machine learning works to detect the buffer overflow and, after that, how it is integrated with AWS to get the result.

2.5 Research question

Q1) To investigate the particular security needs of various cloud-integrated cybersecurity apps, considering issues such as data sensitivity, attack methods, authentication and authorization, and regulatory compliance.
Q2) To contrast the suggested CSM with current cloud security models and frameworks, emphasizing its benefits and distinguishing features in resolving the security problems of cloud-integrated cybersecurity applications.

3 Related Work

The Role of Machine Learning in Cloud Security

According to the authors You and Chen (2021), the conventional method is not able to provide security for most of the vulnerabilities identified. Although rule-based techniques used in firewalls and systems for intrusion detection are useful to some extent, they must be updated on a regular basis as fraudsters develop new attack paths. This disadvantage, together with the massive volume of data created by cloud settings, necessitates a more efficient and adaptable security solution. Machine learning algorithms provide an opportunity for detecting and counteracting possible risks by utilizing pattern identification, recognition of anomalies, and predictive analytics observed as a result of You & Chen (2021).

Understanding Machine Learning in Cloud Security

According to Naeem (2023), to generate predictions or conclusions, machine learning algorithms acquire knowledge from current information patterns and detect connections. These algorithms analyze historical data, network traffic logs, system logs, and user behaviors in the context of cloud security to generate a comprehensive picture of a cloud environment. This understanding enables machine learning models to identify abnormalities, forecast malicious behaviors or vulnerabilities, classify various sorts of threats, and respond to possible security issues in real time (Naeem, 2023).

Machine Learning-Based Threat Detection Benefits for Cloud Security

According to the authors Dalal and Rele (2018), Real-time threat detection: Machine learning models excel at processing massive volumes of data in near real-time, allowing risks to be identified as they occur. This proactive strategy shortens the time it takes to respond to security incidents, reducing potential harm (Dalal & Rele, 2018).

Anomaly detection: Algorithms that use machine learning can recognize normal cloud behavior patterns and highlight any variations between these baselines as possible dangers. This capacity to recognize unique attack patterns is critical for remaining one step ahead of hackers (Dalal & Rele, 2018).

Scalability: Because cloud systems expand quickly, security solutions that can adapt to increasing size and complexity are required. Machine learning algorithms can manage this dynamic environment well, detecting threats accurately even as the cloud infrastructure increases or collapses.

False positives are reduced: Traditional rule-based systems frequently create false positives, resulting in excessive alarms and possibly distracting security staff. By properly screening out innocuous actions based on learned behaviour patterns, machine learning algorithms may drastically minimize false positives. According to the author, Traditional rule-based systems frequently create false positives 65% (Dalal & Rele, 2018).

3.1 Summary

Literature	Key focus	Gaps
The Role of Machine Learning in Cloud Security	• The incapacity of conventional security measures to cope with the constantly changing risks in cloud environments. • The enormous amount of data created in cloud	• There is little data on how well machine learning algorithms work in practice or in cloud security. • Insufficient research has been done on the possible drawbacks and weaknesses of

	systems necessitates the need for more flexible and effective security solutions. • The ability of machine learning algorithms to use anomaly detection, pattern recognition, and predictive analytics to identify and reduce security threats.	machine learning-based cloud security systems. • There is a requirement for more research on the ideal methods and factors to take into account when incorporating machine learning algorithms into already-in-use cloud security frameworks.
Understanding Machine Learning in Cloud Security	• Machine learning algorithms are used to analyze past data, system logs, network traffic logs, and user activity to improve cloud security. • The capacity of machine learning models to identify anomalies, predict malevolent actions or weaknesses, categorise dangers, and react instantly to security breaches.	• There is little data on how well machine learning algorithms work in cloud systems to identify and stop sophisticated and complicated security threats. • Here are the aspects where the researcher lags in his/her research possible drawbacks and biases of machine learning models in cloud security, including results interpretability and false positives/negatives. • More case studies and real-world applications are required to show how machine learning affects cloud security in the actual world. • Inadequate investigation of the moral ramifications of using machine learning algorithms to cloud security decision-making.
Machine Learning-Based	• The advantages of	• A brief explanation of

Threat Detection Benefits for Cloud Security	cloud security machine learning for real-time threat detection. • The capacity of machine learning algorithms to identify irregularities in the patterns of cloud behaviour. • The scalability of cloud security in dynamic contexts using machine learning methods. • The use of machine learning methods to reduce false positives in threat detection.	the various machine learning algorithms and methods utilised in cloud security threat identification. • Inadequate examination of the possible drawbacks or difficulties associated with integrating machine learning-based threat detection in cloud security. • There is a lack of research on the resources needed and the cost-effectiveness of incorporating machine learning techniques into cloud security systems. • A cursory analysis of the privacy and data protection issues of using machine learning for cloud threat detection.

4 Research Methodology

4.1 Data gathering

The dataset for this research is taken from CVE details and an online portal that list all the vulnerabilities faced in CSM for the expert, students, and analyst to have access to the information and device some security measure individual similarly the researcher collected the vulnerability issue in CSM from there to work on. For the literature review, the pieces of literature are mainly journals and articles that are collected from Google Scholar and between the time ranges of 2015 and the present.

4.2 Dataset

The dataset which is used in this project is a buffer overflow malware attack on the mail. Using this dataset, the project can showcase the result in a more accurate way and detect the attack that happens on the mail, which is important for this project.

4.3 Research Design

For this research, the step-by-step procedure has been followed in order to get the proper result, which is to start from the dataset that what kind of dataset this project requires, after that, which tool or which language will be suitable for this being selected that the integration of AWS with threat detection model and in the end the process of working (Jansen, 2023).

- Data collection: Data is collected from CVEdetail.com website, which serves users with the details of the present vulnerabilities faced by CSM (Cloud service management).
- Data-Preprocessing: Removing the special character and null values, sklearn. Preprocessing library.
- Gathering information on dataset: After preprocessing again using the info command and then generating different column plots for better understanding.
- Feature Selection: List of features selected for the detection model of datatype object ('cve_id', 'vendor_project', 'date_added', 'required_action', 'due_date', 'grp', 'pub_date', 'class', 'vector', 'complexity').
- Validating and training the model: Dropping the severity column and then implementing the different machine learning models (logistic regression, decision tree, random forest, gradient boosting, Ada boosting, SVM, KNN, Gaussian naïve Bayes, neural network and extra trees).
- Model performance: Calculating the accuracy, precision, and recall of all the models.
- AWS integration: Importing boto3 library for integration, and uploading the detection model, JSON file on cloud.

4.4 Creation of a Cloud Security Model

In order to tackle the issues and offer recommendations for the safe implementation of cloud-integrated cybersecurity applications, a thorough cloud security framework will be created according to a combination of results. The best practices and tactics for improving the safety measures of cloud-using enterprises will be incorporated into the model (Nolle, 2023).

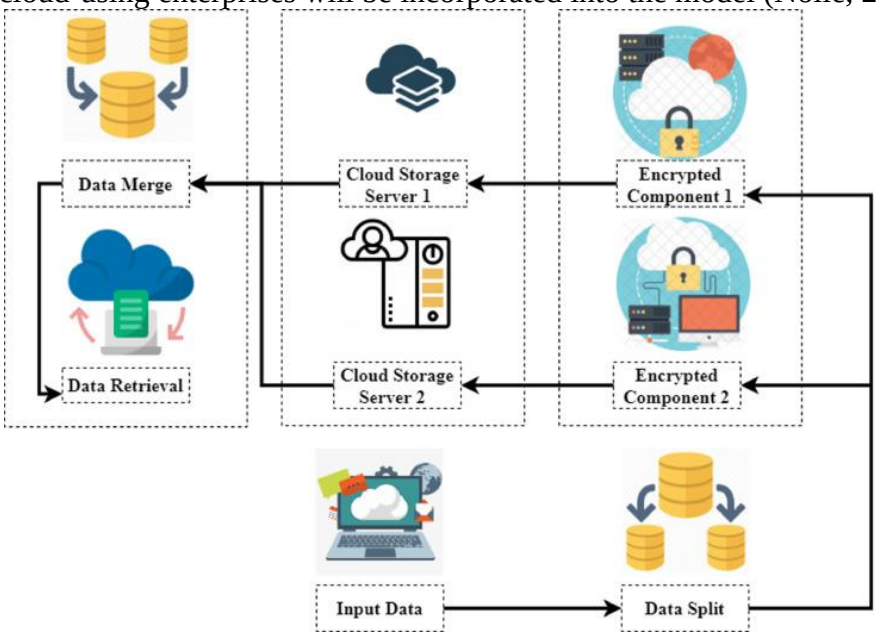

Figure 1 Cloud security flow

5 Design Specification

5.1 Virtualization Environment

One important architectural strategy in the suggested cloud security paradigm is environment virtualization. It makes it possible to create or use an environment that shares a real server. Environment Virtualization guarantees that a breach of security does not jeopardize the entire public cloud architecture by separating from one another. It improves security and performance by enabling the allocation of specific resources for every application (IBM, 2021).

5.2 IDPS

An Intrusion Detection and Prevention System, or IDPS, is a crucial part of the paradigm that has been suggested. It keeps an eye on network activity, spots possible threats or irregularities, and reacts quickly to neutralize or stop them. The IDPS can efficiently identify and address changing cyber threats by utilizing machine learning techniques. The IDPS is essential to preserving the safety and confidentiality of a cloud-based environment since it can identify and stop malicious activity or attempts at unauthorized access (Specialist et al., 2022).

5.3 Secure APIs and Integration

Applications and cloud services may be seamlessly integrated with one another thanks to APIs, or application programming interfaces. The suggested approach places a strong emphasis on the usage of properly approved and authorized secure APIs to guard against illegal access to cloud-based services. Secure integration makes sure that possible dangers are kept out of the exchange of information between different cloud-based system components (Cobb, 2022).

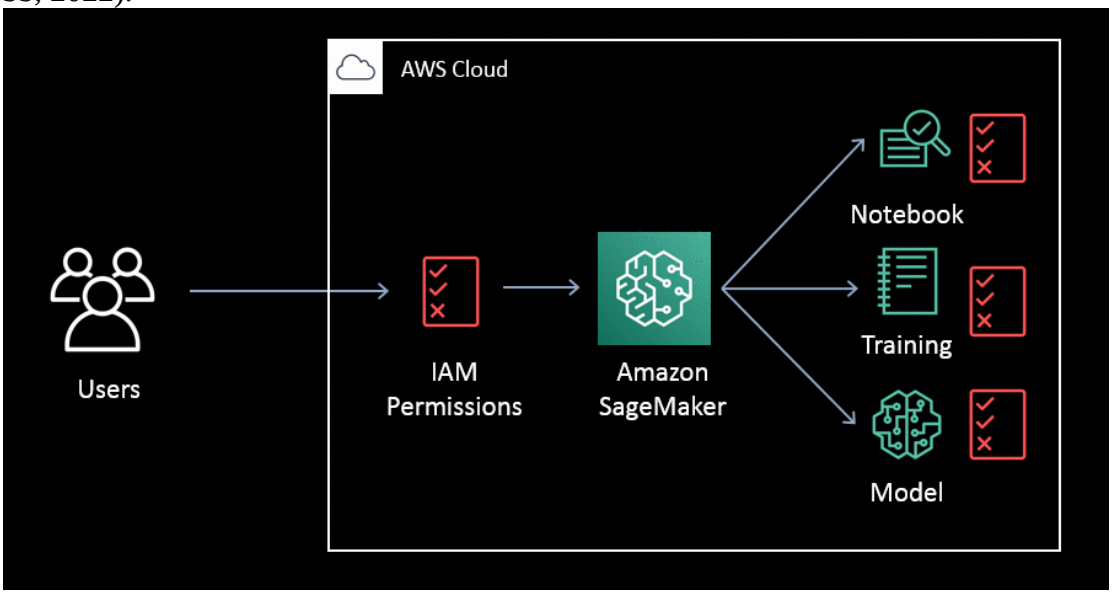

Figure 2 Cloud integration

5.4 Data Protection

In the suggested model, these two crucial methods are used to protect private data kept on the cloud. Data is password-protected before it is saved in a cloud server as well as is only decrypted when accessed by approved individuals thanks to Aws provided code. This keeps

data from being tampered with or accessed without authorization, even in the case of a breach (Zhang et al., 2021).

5.5 Backup and Disaster Recovery

The suggested architecture heavily relies on backup and disaster recovery strategies. To guarantee that important data and programs can be recovered from possible data corruption or system failure, regular backups are crucial. After a disruptive occurrence, the backup data needs to be easily accessed and securely stored in order to bring the cloud environment back to normal.

6 Implementation

In order to create a comprehensive model of cloud security that can recognize and prevent attacks in a cloud environment, the implementation focuses on the secure deployment of cybersecurity apps connected to the cloud. To do this, a range of machine learning techniques are applied, such as ensemble models, neural networks, K-Nearest Neighbours (KNN), gradient boosting, Random Forest, extra trees, decision trees, logistic regression, support vector machines (SVM), and Gaussian Naive Bayes.

Libraries and Data Preparation

The dataset for the machine learning models is first subjected to the data preparation phase, which involves dividing gathering information into sets for training and testing, scaling numerical features, encoding variables with categories, and managing missing values. The necessary libraries, such as pandas, numpy, matplotlib, seaborn, and sci-kit-learn, are imported in the first step. These libraries offer functions for modeling, evaluation, visualization, and preparation of data.

For data preprocess the first step is to get the information of the data set such as the numer of columns, different datatypes, null values and many more as this allow the developer to set the dataset as per the requirement of this reseach.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 774 entries, 0 to 773
Data columns (total 16 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   cve_id              774 non-null    object 
 1   vendor_project      774 non-null    object 
 2   product             773 non-null    object 
 3   vulnerability_name  774 non-null    object 
 4   date_added          774 non-null    object 
 5   short_description   768 non-null    object 
 6   required_action     774 non-null    object 
 7   due_date            774 non-null    object 
 8   notes               0 non-null      float64
 9   grp                 774 non-null    int64  
 10  pub_date            765 non-null    object 
 11  cvss                609 non-null    float64
 12  cwe                 760 non-null    object 
 13  vector              609 non-null    object 
 14  complexity          609 non-null    object 
 15  severity            609 non-null    object 
dtypes: float64(2), int64(1), object(13)
memory usage: 96.9+ KB
```

Dropping the null valued as it creates the irregularity in the data set. This will not provide the exact result while performing the analysis.

```
cve_id                0
vendor_project        0
product               0
vulnerability_name    0
date_added            0
short_description     0
required_action       0
due_date              0
notes                 0
grp                   0
pub_date              0
cvss                  0
cwe                   0
vector                0
complexity            0
severity              0
dtype: int64
```

Libraries for Threat detection:

In [1]:

```
import pandas as pd
import numpy as np
import warnings
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split


from sklearn.feature_selection import SelectKBest, f_classif


from sklearn.metrics import accuracy_score, precision_score, recall_score


from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier, AdaBoostClassifier, StackingClassifier, ExtraTr
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.neural_network import MLPClassifier



warnings.filterwarnings("ignore")


warnings.filterwarnings("default", category=DeprecationWarning)
```

Figure 3 Libraries for Threat Detection

Here different libraries are imported to serve the results such as pandas are used for data manipulation and analysis, providing high-level data structures. Another library is matplotlib, a comprehensive data visualization library for creating visuals, seaborn is used to present the statistical data visualization library in Python.

Libraries for AWS Integration:

In [2]:

```
import boto3
import pandas as pd

s3Cloud = boto3.client('s3')
```

Boto3 is an AWS software development used for Pythons it allows the developer to interact with AWS services using Python code.

Model Training and Evaluation

```
In [18]:
models = {
    'Logistic Regression': LogisticRegression(max_iter=10000),
    'Decision Tree': DecisionTreeClassifier(),
    'Random Forest': RandomForestClassifier(),
    'Gradient Boosting': GradientBoostingClassifier(),
    'AdaBoost': AdaBoostClassifier(),
    'SVM': SVC(probability=True),
    'K-Nearest Neighbors': KNeighborsClassifier(),
    'Gaussian Naive Bayes': GaussianNB(),
    'Neural Network': MLPClassifier(max_iter=10000),
    'Extra Trees': ExtraTreesClassifier()
}

results = []


for name, model in models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)

    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred, average='weighted')
    recall = recall_score(y_test, y_pred, average='weighted')

    results.append({
        'Model': name,
        'Accuracy': accuracy,
        'Precision': precision,
        'Recall': recall
    })
```

Figure 4 Models

Ensemble Model

With an accuracy of around 88.43%, the ensemble model performs the best. The combined model's precision score is somewhat greater than its accuracy, indicating forecasts that may be trusted.

KNN (K-Nearest Neighbours)

The KNN model shows almost identical accuracy while having substantially better precision compared to the ensemble model, indicating a high percentage of positive predictions that are true.

Neural Network

With lesser precision and a third-place accuracy ranking, the Neural Network may be over-predictive in some classes.

Gradient Boosting

The Gradient Boosting model indicates reliable positive predictions, and it has the highest precision among the top models, however, it is somewhat less accurate than the Neural Network.

Random Forest and Extra Trees

These models have almost similar scores and do well in terms of accuracy but might use some fine-tuning in terms of precision.

Decision Trees

The accuracy of the Decision Tree model is comparable to that of Random Forests and Extra Trees.

Support Vector Machines (SVM) and Logistic Regression

These models imply more false-positive predictions, with similar accuracy ratings but lower precision.

Gaussian Naive Bayes

The model with the highest accuracy score, the lowest precision, and the most reliable positive predictions is the Gaussian Naive Bayes.

Here is the table of all the results extracted from the different models.

Table 1 Results

Algorithm	Accuracy	Precession	Recall
Logistic Regression	0.99	0.98	0.99
Decision Tree	0.99	0.98	0.99
Random Forest	0.99	0.98	0.99
Gradient Boosting	0.99	0.98	0.99
Random Forest and Extra Trees	0.99	0.98	0.99
Ada Boosting	0.99	0.98	0.99
Naïve Bayes	0.99	0.98	0.99
Extra tree	0.99	0.98	0.99
Ensemble Model	0.99	0.98	0.99
Neural Network	0.72	0.73	0.73
KNN	0.61	0.61	0.61
SVM	0.58	0.40	0.58

As the table shows that models like logistic regression, decision tree, random forest, gradient boosting, random forest, ada boosting, naïve bayes, extra tress and ensemble models have 99% accuracy and 98% precession and 99% recall and the rest that is svm, KNN and neural network have the low accuracy. Among all models SVM has the lowest accuracy, precision and recall. As for this project these models will not be taken for further analysis.

6.1 Result

The researcher developed these different models for the research and calculated their accuracy, precession and recall. As the results below show, most of the implementation models provide 99% per cent. The result model will generate the result in JSON format which contains the detection accuracy details that show if the attack that happened in the mail is very effective or is happening or not.


```
[{"Model":"Logistic Regression","Accuracy":0.9935483871,"Precision":0.9874551971,"Recall":0.9935483871},{"Model":"Decision
Tree","Accuracy":0.9935483871,"Precision":0.9874551971,"Recall":0.9935483871},{"Model":"Random
Forest","Accuracy":0.9935483871,"Precision":0.9874551971,"Recall":0.9935483871},{"Model":"Gradient
Boosting","Accuracy":0.9935483871,"Precision":0.9874551971,"Recall":0.9935483871},
{"Model":"AdaBoost","Accuracy":0.9935483871,"Precision":0.9874551971,"Recall":0.9935483871},{"Model":"Gaussian Naive
Bayes","Accuracy":0.9935483871,"Precision":0.9874551971,"Recall":0.9935483871},{"Model":"Extra
Trees","Accuracy":0.9935483871,"Precision":0.9874551971,"Recall":0.9935483871},{"Model":"Ensemble
Model","Accuracy":0.9935483871,"Precision":0.9874551971,"Recall":0.9935483871},{"Model":"Neural
Network","Accuracy":0.7225806452,"Precision":0.7308460448,"Recall":0.7225806452},{"Model":"K-Nearest
Neighbors","Accuracy":0.6129032258,"Precision":0.6161324953,"Recall":0.6129032258},
{"Model":"SVM","Accuracy":0.5806451613,"Precision":0.4032258065,"Recall":0.5806451613}]
```


Figure 5 Result

6.2 Discussion

In today's digital world, cloud computing has become a commonly utilized technology. It has several advantages, including flexibility, cost-effectiveness, and scalability. Nonetheless, security risks are also raised by the growing usage of cloud services. Due to their frequent inability to keep up with the quickly changing threat landscape, traditional security solutions are vulnerable to possible breaches. Machine learning techniques have become a feasible cloud security approach to solve these issues.

The capacity of machine learning to identify abnormalities is one of its main benefits for cloud security. These algorithms (Logistic regression, Decision tree, ADA boosting, random forest, gradient m boosting, naïve bayes, extra tree, ensemble model) are able to create benchmarks for typical cloud activity and recognize deviations from these standards to get. As a result, security systems can keep up with hackers and identify distinct attack patterns that conventional rule-based systems could overlook.
Conventional rule-based systems frequently sound needless alerts, wasting resources and perhaps causing security personnel to get distracted. Based on observed patterns of behavior, machine learning algorithms may efficiently weed out benign activity, reducing false positives and freeing up security personnel to concentrate on real threats.
From an academic standpoint, the work adds to the corpus of knowledge already available on machine learning and cloud security. It offers concrete proof of how well machine learning algorithms identify and reduce dangers in cloud systems. The report also emphasizes how crucial it is to use machine learning methods in the development and deployment of cloud integrated.

The researcher's first approach while looking at the issue to sort the buffer as it is the root cause that is providing the attackers a loophole to attack the system and different users. So the first approach is to deal with the buffer for that the buffer size is increased and a machine learning model is prepared to perform the intrusion detection caused by the buffer. And prepared the different machine learning models to calculate the results to the detection. And the results show that almost all the models provide accuracy up to 99% except the SVM, KNN and neural networks which are below 72%. Once this model is complete the researcher sets the AWS connection and generates the JSON file which is uploaded at the cloud to provide the solutions to the public who wants to access it. As this vulnerability in the CMS is also found from the portal so it will allow them to look at possible solution and mitigate the issue.

7 Conclusion and Future Work

In conclusion, by offering more effective and flexible security solutions, machine learning significantly improves cloud security. Firewalls and detection systems for intrusions that employ conventional rule-based methodologies are helpful, but they need to be updated often to stay up to speed with new attack strategies. Machine learning algorithms are able to recognize trends, spot irregularities, and carry out predictive analytics in order to quickly identify and mitigate such threats. Cloud security may reap several advantages from the application of machine learning-based identification of threats. First, machine learning models are excellent at processing large amounts of data in almost real-time, making it possible to identify dangers as soon as they arise. Here the major purpose of the researcher is to come up with a solution to deal with a threat detection model for buffer flow. And for the different machine learning models are prepared in order to get the accuracy of the models, in which SVM, Neural network and the KNN gave results below 72% rest all the other models used gave 99 % accuracy, 98% precision and 99% recall which suggests that the detection of buffer low done by these models is quite to perfect and then as soon as the models developer a JSON file also generated that will be uploaded to the AWS cloud architecture which allows the general public who are interested in working on these vulnerabilities will access to the proposed solution and detection model for their work.

For future tech, taking a proactive stance reduces possible harm by speeding up the reaction to security issues. Another benefit of machine learning for cloud security is scalability. Security solutions need to be flexible enough to grow with cloud systems, which are rapidly becoming larger and more sophisticated infrastructures. This dynamic

environment may be handled by machine learning algorithms, which can reliably identify risks even as the infrastructure for the cloud expands or contracts. Furthermore, in comparison to conventional rule-based systems, machine learning techniques can considerably minimize false positives. False positives can divert security personnel's attention from real dangers and frequently cause needless alarms. Machine learning algorithms are capable of greatly reducing false positives by filtering out harmless behaviors according to learned behaviour patterns.

References

- Bhandari, P. (2023) Data collection: Definition, Methods & Examples, Scribbr. Available at: https://www.scribbr.com/methodology/data-collection/ (Accessed: 18 November 2023).
- Cobb, M. (2022) 12 API security best practices to protect your business: TechTarget, App Architecture. Available at: https://www.techtarget.com/searchapparchitecture/tip/10-API-security-guidelines-and-best-practices (Accessed: 18 November 2023).
- cochrane, cochrane (2021) Data synthesis and analysis - cochrane. Available at: https://cccrg.cochrane.org/sites/cccrg.cochrane.org/files/public/uploads/AnalysisRestyled_FINAL%20June%2020%202016.pdf (Accessed: 18 November 2023).
- Dalal, K.R. and Rele, M. (2018) 'Cyber security: Threat detection model based on machine learning algorithm', 2018 3rd International Conference on Communication and Electronics Systems (ICCES) [Preprint]. doi:10.1109/cesys.2018.8724096.
- Druva, D. (2021) What is multi-layered security?, Druva. Available at: https://www.druva.com/glossary/multi-layered-security (Accessed: 18 November 2023).
- IBM, I. (2021) What is virtualization?, IBM. Available at: https://www.ibm.com/topics/virtualization (Accessed: 18 November 2023).
- Jansen, D. (2023) What is research design? 8 types + examples, Grad Coach. Available at: https://gradcoach.com/research-design/ (Accessed: 18 November 2023).
- Naeem, H. (2023) 'Analysis of Network Security in IOT-based cloud computing using machine learning', International Journal for Electronic Crime Investigation, 7(2). doi:10.54692/ijeci.2023.0702153.
- Nolle, T. (2023) What is cloud security architecture?, Security. Available at: https://www.techtarget.com/searchsecurity/definition/cloud-security-architecture (Accessed: 18 November 2023).
- Specialist, R.M.I. et al. (2022) What is intrusion detection and prevention system? definition, examples, techniques, and best practices, Spiceworks. Available at: https://www.spiceworks.com/it-security/vulnerability-management/articles/what-is-idps/ (Accessed: 18 November 2023).
- You, W. and Chen, B. (2021) 'Protocols for cloud security', Machine Learning Techniques and Analytics for Cloud Security, pp. 293–312. doi:10.1002/9781119764113.ch14.
- Zhang, E., Groot, J.D. and Lord, N. (2021) What is Data Encryption? (definition, Best Practices & More), Digital Guardian. Available at: https://www.digitalguardian.com/blog/what-data-encryption (Accessed: 18 November 2023).