An Efficient Model to Predict Network Packets in TVDC Using Machine Learning

Ashmeet Kaur Duggal * and Meenu Dave

Department of Computer Science and Engineering, Jagannath University, Jaipur, India;
Email: meenu.s.dave@gmail.com (M.D.)
*Correspondence: ashmeet04@yahoo.co.in (A.K.D.)

***Abstract*—Internet-based computing allows the sharing of on-demand resources. This computing technique includes data processing and storage to globally separated machines, known as Cloud Computing. Confidentiality and integrity of data on the cloud are vital. The key constraints include effective access control, accessibility, and transmission of files, in a dynamic cloud environment, seeking a Trusted Virtual Data Center (TVDC). So, to overcome challenges such as data security and integrity due to exponentially growing data size, this research paper aims to develop a prediction model using the machine learning approach, which identifies the type of incoming packet on the TVDC. Alternatively, in other words, this system predicts whether the incoming packets on the server in the cloud environment are malicious or not, using the machine learning approach. This research explored artificial intelligence verticals in building systems with learned data structures for efficient data access. This research describes the implementation of machine learning algorithms for an efficient model's prediction of the type of incoming packet on the server. It has achieved 88% accuracy using the Gradient Boosted Tree classifier. Also, in this study, the author compares the results of two algorithms, Decision Tree and Gradient Boosted Tree, and finally selects the most optimal for this prediction.**

***Keywords*—machine learning, Amazon Web Services (AWS), Elastic Compute Cloud (EC2), artificial intelligence, cloud computing, trusted virtual data center**

I. Introduction

We live in exponential times where data is growing 40% a year [1], and by 2023, the size of information will reach 44 zettabytes or 44 trillion gigabytes. The last two decades have advocated the transition from the electronic to the information age. Potential technologies have evolved on the merger of mechanical, electronic, and computer science disciplines. Cloud computing is a breakthrough that presents a subscription-based service to obtain network storage space and computer resources. A great deal of attention from individuals, government, and industry is being exploited by cloud computing. Cloud computing enables ubiquitous, convenient, efficient, and on-demand access to data, information, and resources.
Cloud Computing's primary service models are:

- Infrastructure as a Service
- Platform as a Service
- Software as a Service

Cloud computing has successfully transformed the public and private sectors' investment base from capital expenditure to operational expenditure [2]. Online users and enterprises are powered by cloud computing to store, process, and access data using third-party data centers. To meet fluctuating and non-predictable business demands, it provides a pay-per-use model for faster deployment, execution, and resource management. The worldwide availability of cloud data centers has made ease of access to data and resources among vital sectors, including but not limited to government, banking, finance, academics, healthcare, automotive, media, and sports.

Trusted Virtual Data Center (TVDC): It may be an assortment of cloud resources mainly designed for business needs. The basic resources consist of CPU, databases, Storage, networking, etc. All of them are present inside a virtual space being hosted by some data centres (one or more) [3].

Amazon Web Services (AWS) offers a big list of global cloud-based products that include storage, databases, networking, analytics, security, IoT, and enterprise applications. AWS helps organizations to move faster at lower IT costs. It gets the trust of large enterprises and start-ups to power various workloads that include game development, web, and mobile apps, warehousing, data processing, data storage, and archiving (Amazon EC2). In this research, the AWS Elastic Compute Cloud (EC2) service is used for the development of the prediction model of TVDC and an Intelligent IAM [4].

Elastic Compute Cloud (EC2) helps to provide secure, scalable computing capacity in the cloud. It is a web service interface that permits its users to acquire and configure capacity only with fewer challenges. It provides users with complete control of its computing resources and permits the user to run over Amazon's proven computing environment. Amazon EC2 allows users to scale the capacity, up and down, quickly as his/her computing requirements alter. Amazon EC2 brings changes to computing economics by permitting users to pay as per the capacity he uses. In this study, one instance is launched using the Amazon EC2 service for the prediction model of TVDC [5].

Manuscript received June 3, 2022; revised October 11, 2022; accepted December 4, 2022; published June 7, 2023.

This research paper is organized into five sections: Section I includes the introduction to cloud computing and its services, and Section II for the literature review based on cloud computing services such as TVDC, AWS, EC2, and network traffic analysis using a machine learning approach, Section III specifies the research methodology for the prediction of the type of incoming packets on the server in the cloud environment, Section IV for Implementation, Analysis & Results for the prediction model, which includes the topics such as Data Collection, Data Preparation, Model Training, and Model Evaluation. Finally, the author concludes the paper with future work related to this research study in Section V.

II. Literature Review

This paper presents IBM's Trusted Virtual Datacenter (TVDC) technology developed to resolve the issue of solid isolation and integrity guarantees, thus significantly enhancing security and systems management capabilities in virtualized environments. It represents the first effort to incorporate trusted computing technologies directly into virtualization and systems management software. The author presents and discusses various components that constitute TVDC: The Trusted Platform Module (TPM), the virtual TPM, IBM hypervisor security architecture (sHype), and the associated systems management software. This research has proven the feasibility of managing Trusted Virtual Domains across servers, networks, and storage resources. However, several research challenges must be addressed to facilitate TVDC deployment and operation. Some of the obstacles to the virtualization are complexity at the customer's end to onboard to TVDC, interaction of different management constraints, such as availability and resource management, with security constraints, such as anti-collocation rules [6].

Trusted Virtual Data Center (TVDC) was developed to address the need for solid isolation and integrity guarantees in virtualized environments. In this paper, the author has implemented controlled access to network storage based on security labels and by implementing management prototypes that demonstrate the enforcement of isolation constraints and integrity checking. Also, the author has extended the management paradigm for the TVDc with a hierarchical administration model based on trusted virtual domains [7].

In this research paper, the author discusses Amazon Web Services (AWS), one of the world's best cloud service providers. AWS is the most trusted and reliable source of providing cloud-computing services. It provides a wide variety of services with well-managed security as well. It is a secure cloud platform that offers various global cloud-based products. Since these products are delivered over the internet, users have on-demand access to the computation resources, storage, network, database, and other IT resources — and the tools to manage them. Customers can immediately provision and launch AWS resources. AWS environment can be reconfigured and updated on demand, temporarily or permanently. Cloud computing suffers from several problems like loss of data, account hijacking, insecure interfaces and APIs, cost-effectiveness, and many more, but using AWS, we can effortlessly master all these problems [8].

Cloud Computing is an emerged model already popular among almost all enterprises. It offers us the concept of on-demand services where users use and scale cloud resources on demand. AWS is a cost-effective model. The prime concern in this model is Security and Storage in the cloud. This is one of the primary reasons why many enterprises choose AWS cloud computing. In this paper, the author reviews security research in the cloud security and storage services of the AWS cloud platform. Also, the author has presented the working of AWS cloud computing. AWS is the most trusted provider of cloud computing, which provides excellent cloud security, and at the same time, it offers excellent cloud storage services. The main objective of this paper is to make cloud computing storage and security a core operation and not an add-on operation [9].

The Cloud computing technology allows users to remotely store their data in the cloud and offers on-demand applications and computing services from a shared pool of configurable computing resources. The protection of the outsourced data in the cloud environment entirely depends on the security of the cloud computing system and network. Though numerous efforts have been made to secure data on the cloud computing system, evaluating data security on the network between the cloud provider and its users is still a very challenging task. Network traffic analysis for cloud auditing is of critical importance so that users can resort to an external audit party to verify the data security on the network between the cloud provider and its users. In this research paper, the author presents the following vital technologies to analyze network traffic in the cloud computing environment: IP geolocation, Router IP analysis, and online data mining [10].

MLDB.ai [11] is an open-source database designed for machine learning. Enterprises generate large amounts of structured data stored in cloud databases. Solutions like MLDB offers speed, scalability, ease of use, integration, and deployment. Analysis of such data opens potential business opportunities and overcomes business challenges.

Existing in-database machine learning solutions like cuttlefish [12] offer constant running time and memory usage, but they lack failure management and possess descriptive limitations.

We performed an exhaustive literature survey on artificial intelligence techniques that influence data management [13]. The techniques include:

- Natural language interfaces for databases
- Machine learning techniques for data integration and cleaning
- Machine learning services to enhance database interaction
- Self-managing operational aspects
- Self-managing database internals
- In-database machine learning techniques

- Machine learning techniques for implementing database internals

III. Research Methodology

In this paper, the author has proposed an artificially intelligent trusted virtual data center, which is an assortment of resources in the cloud environment. The flowchart describing the methodology followed during this research is defined in the above Fig. 1. In this research, a TVDC is created using AWS services like EC2 to create an instance of its type using one of the Amazon Machine Image (AMI). The AMI that the author is using in this research is the Dataiku tool which is used for the analysis. The complete procedure can be best explained in the form of an algorithm as shown in the below given Fig. 2.

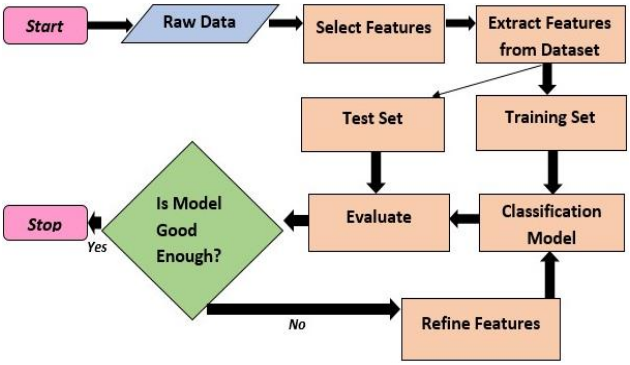

Figure 1. Flowchart describing the methodology followed during the study.

ALGORITHM 1: Working of Trusted Virtual Data Center (TVDC)

Input: ACK Flag Count, Source Port, Average Packet Size, Forward Packet Length Std, Packet Length Mean, Subflow Fwd Bytes.

Output: Network Packet Prediction of TVDC.

Step1: Start by creating a TVDC
Step 2: Open the AWS Management Console
Step 3: Sign-in to the AWS Account (Root User)
Step 4: Select the AWS Service (EC2)
Step 5: Select the AWS Region (Mumbai, ap-south-1)
Step 6: Launch an Instance (Virtual Server)
Step 7: Name the Instance
Step 8: Select the Instance Type (m5.xlarge)
Step 9: Select an AMI (Dataiku)
Step 10: Execute the Instance by running its public IP Address
Step 11: The AMI gets started (Login)
Step 12: Import the dataset
Step 13: Use the tool Puttygen and Putty for dataset connectivity
Step 14: Prepare/clean the dataset
Step 15: Split the dataset (80% - 20%)
Step 16: Train the dataset on various machine learning algorithms
Step 17: Evaluate the prepared dataset using the performance metrics
Step 18: Analyze the results for Network Packet Prediction of TVDC

Figure 2. Algorithm showing the working of TVDC.

We have created 11 machine learning models to predict of the incoming packets on the server by Trusted Virtual Data Center.

Decision Tree (DT) is preferred more because of its easy computation and a better understanding of why a variable has a higher priority. Besides, DT can be represented in a graphical format making it convenient for non-technical users to comprehend. Also, DT is intuitive as it recognizes the dataset's hidden internal pattern, so the time of pre-processing of data is saved.

On the other hand, Gradient Boosted Tree (GBT) is also quite efficient and user-friendly while working with multiple operations. It produces the prediction model using regression techniques. It gives similar results as DT based on the accuracy of the algorithm. Nevertheless, the problem is that GBT takes a lot of time to build each tree.

Logistic Regression (LR) uses the sigmoid function and is preferred over other models when working with binary classification problems. It is considered to be highly accurate. Although it is a regression algorithm, it is an excellent algorithm for classification. The Support Vector Machine (SVM) is also a highly accurate algorithm that operates with associated learning algorithms required to analyze the data used for regression and classification analysis. Not as high as DT and GBT, but SVM can still be considered a reliable algorithm.

XGBoost (XGB) is a DT based algorithm that uses a gradient boosted model to work upon the data. It is also a highly accurate model for prediction problems for unstructured data.

The Extra Trees (ET) algorithm operates upon constructing a large number of unpruned DT from the given data set.

K Nearest Neighbor (KNN) is a non-parametric algorithm that uses the k-closest training examples from the dataset to predict the required prediction. It is slightly

less accurate and has precision on the lower side as compared to DT and GBT.

Random Forest (RF) is an ensemble learning model that works by building a multitude of decision trees on the given test and gives out the required prediction result accordingly. It has accuracy on the lower side in this case.

Artificial Neural Networks (ANN) is an algorithm that uses the concept of biological neural networks to give out the prediction result on the given data set. It can be used in deep learning applications. It has a very low accuracy and precision level in this case, and hence it is not preferred over other algorithms.

Stochastic Gradient Descent (SGD) is an iterative method for optimizing the objective function during the prediction of the data set. Since it works with approximations, it is not considered to be highly accurate.

Lasso least Angle Regression (LLAR) uses the unique structure of the lasso problem. A forward stepwise regression model is not considered very accurate when used as a prediction model on a dataset.

IV. Implementation and Results

This proposed research work is implemented in a lab-based environment where the author has used python programming language to implement machine learning mathematical models. The proposed work has been implemented on python version 3.7.2 using the Jupyter notebook. In this research, the author has created a trusted virtual data center using AWS services like EC2 by launching an instance over the cloud environment using the Amazon Machine Image (AMI), which is the Dataiku tool that is used for analysis. For connectivity of the dataset with the AMI, the author uses the Puttygen and Putty software tools. In this research, a predictive model is developed, which clearly and efficiently identifies the type of incoming packet (malicious/non-malicious) on the server in the cloud environment.

We have implemented the Sample, Explore, Modify, Model, and Assess (SEMMA) framework for machine learning. Description of SEMMA machine learning stages are as follows:

- Sample — Data collections and Selection
- Explore — Perform exploratory data analysis and visualization
- Modify — Perform data pre-processing, scaling, and transformation
- Model — Generate a highly accurate model for the prediction of the response variable
- Assess — Interpretation and evaluation of mathematical models based on accuracy, precision, recall, f1 score, log loss, and roc curve.

A. Development of Prediction Model for TVDC

In this part, we have implemented a prediction model to identify the type of incoming packet on the server using a network traffic dataset. We import required libraries, including NumPy, pandas, sklearn, pandautils, and principal component analysis from pre-processing. We use the default dictionary and data frames (2D-Size mutable).

1) Data collection

Several datasets exist, such as DARPA, KDD, ISC, and ADFA, used by various researchers for analysis, but based on a study in [14], many datasets are internal and private and, on the other hand, are heavily anonymized and lack statistical characteristics. For this research, it was necessary to move away from one-time and static datasets toward more dynamic dataset, which not only reflect traffic composition and intrusions of that time but also is extensible and modifiable. Hence, the dataset used for analysis in this study is DDoS Evaluation Dataset (CIC-DDoS2019) [15]. CIC datasets stand for the Canadian Institute for Cyber-security datasets. These datasets are used all over the world by various researchers, universities, and private industries for analysis [16]. Dataset used in this research includes captured network traffic with more than 80 features extracted from the captured traffic using CIC Flow Meter — V3 software [17]. This dataset is a structured dataset that includes two types of file formats one is Packet Captured (PCAP), and the other is Comma Separated Values (CSV). In this research, the CSV files are used for analysis. The shape of the data used for analysis is 10,000×88, where 10,000 are the observations and 88 are the features used for analysis [18–20].

2) Data preparation

The next step is to clean the data by removing values, removing outliers, handling imbalanced datasets, changing categorical variables to numerical values, etc. Although the CSV files already have the tags, and it is more convenient to use them as input files in machine learning, the files have been carefully examined. The problem was that some data was missing or infinite, which is not valid in machine learning models. So, to resolve this issue, those specified columns like SimilarHTTP, Fwd URG Flags, PSH Flag Count, and Bwd Avg Bytes/Bulk were removed from the dataset using the feature handling function on the Dataiku tool, which is used for analysis. Finally, removing four features from the dataset using the feature handling, the prepared dataset which is used for analysis is 10,000 × 84, where 10,000 are the observations and 84 are the features used for analysis for the prediction of the type of incoming packets on the server in the cloud environment.

We divide our dataset into two new sets. One set will be considered a training set, and another will test the generalization capability. The author uses the pandasutilities to generate the training and testing set using the split_train_valid function. This function takes an input data frame and converts it into two other datasets based on the 80-20 rule. Our training set will contain 80% observations, and random 20% observations will be saved into the test set.

3) Model training

To generate the best model for the prediction of the incoming packets on the server using the network traffic logs, the author implemented the following algorithms:

- Decision Trees (DT)
- Gradient Boosted Trees (GBT)
- Logistic Regression (LR)

- Support Vector Machine (SVM)
- XGBoost (XGB)
- Extra trees (ET)
- K Nearest Neighbors (KNN)
- Random Forest (RF)
- Artificial Neural Network (ANN)
- Stochastic Gradient Descent (SGD)
- Lasso-Least Angle Regression (LLAR)

The author started training the machine learning models with various parameters to achieve the best mathematical model to predict the packet type. Finally, the best two were taken into consideration so as to achieve optimal results.

4) Model evaluation

Once the data is collected and prepared, the model is trained on various machine learning algorithms. Further, the model is evaluated based on six performance metrics. The complete procedure of model evaluation is described in the below-given sections. Finally, the best two models were taken into consideration for comparative study.

The model was interpreted based on various factors like Recall, Precision, F1 score, Log Loss, and ROC curve. The various factors used here are defined below:

Accuracy: It is a metric that is used to check the classification models. It is the ratio of predictions that the model made correctly. It is measured in percentage.

$$\text{Accuracy} = (TP + TN) / (FP + TP + FN + TN)$$

Precision: It refers to the ratio of positively predicted values that were correctly predicted to the total positively predicted values.

$$\text{Precision} = TP / (FP + TP)$$

Recall: It refers to the ratio of positively predicted values that were correct to the sum of the positively predicted values which were correctly predicted and the negatively predicted values which were incorrectly predicted.

$$\text{Recall} = TP / (TP + FN)$$

F1 Score: It refers to the harmonic mean of the recall and the precision. It is a measure to test the accuracy.

$$\text{F1 Score} = 2 \times ((\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall}))$$

Log Loss: It is one of the most valuable metrics based on probabilities used for the classification. Lesser log-loss means that model's performance is good.

$$LogLoss = \frac{-1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} x_{ij*\log(P_{ij})}$$

ROC Curve: It stands for Receiver Operating Characteristic Curve. It is a relationship that provides the ability to diagnose a Binary Classifier system. It is created by plotting the rate of positively predicted values that were correctly predicted to the rate of positively predicted values that were incorrectly predicted.

$$TPR = TP / (TP + FN)$$

where,

TP: True Positive
TN: True Negative
FP: False Positive
FN: False Negative
N: Number of Samples
M: Number of Attributes
Y^{ij}: indicates whether i^{th} sample belongs to J^{th} class or not
P^{ij}: indicates probability of i^{th} sample belonging to j^{th} class
TPR: True Positive Rate

The results of the above mentioned performance metrics is given in the below given Table I. It describes the model evaluation over 6 performance metrics of the best two algorithms.

TABLE I. CALCULATED METRICS AND ASSERTIONS ML MODELS

Model	Accuracy	Precision	Recall	F1Score	Log Loss	ROC
Gradient Boosted Trees	0.88	0.78	0.77	0.77	0.22	0.99
Decision Tree	0.85	0.76	0.87	0.76	0.31	0.98

Let us consider Gradient Boosted Trees for the classification of prediction model. We build an additive model and optimized differentiable loss functions. Using forward stage-wise fashion, we fit regressions trees on each stage at the negative gradient. Table II describes the parameters used to achieve the highest accuracy [21–23].

5) Predictive analysis using gradient boosted tree

The below given Fig. 3 shows the performance metrics using the Gradient Boosted Tree (GBT). Or, in other words, it shows the detailed metrics of the prediction done using the GBT Classification algorithm on the defined dataset. The results showed that it has an accuracy of 88%, ROC AUC Score of 99%, precision, recall and F1 score of approximately 77% and the log loss was 22%.

TABLE II. GRADIENT BOOSTING PARAMETERS FOR CLASSIFICATION

Parameter Name	Parameter value
Loss	Deviance
Learning_rate	0.1
N_estimators	100
Max_depth	3
Random_state	1337
Verbose	0

Figure 3. Metrics and assertions using GBT.

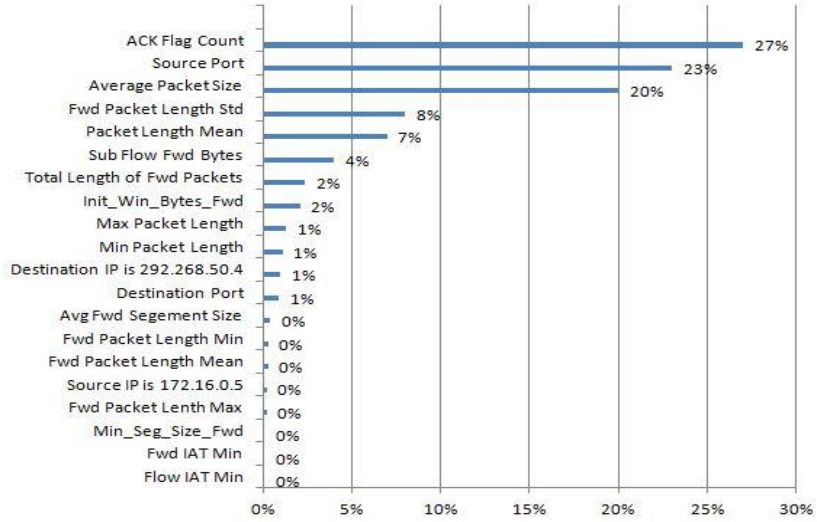

Figure 4. Importance of Variables using GBT.

The above given Fig. 4 shows the analysis with the Gradient Boosted Tree (GBT) classification Algorithm. The results showed the percentage of importance of variables using the GBT algorithm. It was found that ACK Flag Count (27%), Source Port (23%), Average Packet Size (20%), Fwd Packet Length Std (8%), Packet Length Mean (7%), Subflow Fwd Bytes (4%), the remaining variable importance percentage was less than 2%.

The below given Table III shows the confusion metrics using the classification Gradient Boosted Tree algorithm where, the results show the actual and predicted values for the different packets used in the dataset received by the server such as Syn, UDP, NetBIOS, MSSQL, Portmap, LDAP, BENIGN and UDPLag.

TABLE III. CONFUSION METRICS USING GBT

	Predicted								
Actual	**Syn**	**UDP**	**NetBIOS**	**MS SQL**	**Portmap**	**LDAP**	**BENIGN**	**UDPLag**	
Syn	100%	0%	0%	0%	0%	0%	0%	0%	**100%**
UDP	0%	97%	0%	3%	<1%	0%	0%	0%	**100%**
NETBIOS	0%	0%	78%	0%	22%	0%	0%	0%	**100%**
MS SQL	0%	<1%	0%	>99%	0%	0%	0%	0%	**100%**
Portmap	0%	0%	55%	<1%	44%	0%	0%	0%	**100%**
LDAP	0%	0%	0%	0%	0%	100%	0%	0%	**100%**
BENIGN	0%	0%	0%	0%	0%	0%	100%	0%	**100%**
UDPLag	0%	100%	0%	0%	0%	0%	0%	0%	**100%**

6) Predictive analysis using decision tree

The below given Fig. 5 shows the performance metrics using the Decision Tree (DT) Algorithm. Or, in other words, it shows the detailed metrics of the prediction done using the Decision Tree algorithm on the defined dataset. The results showed that it has an accuracy of 85%, ROC AUC Score of 98%, precision and recall of approximately 76% each and the log loss of 31%.

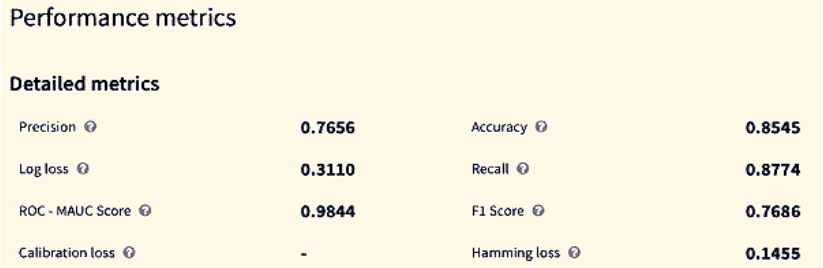

Figure 5. Metrics and Assertions using DT.

The below given Fig. 6 shows the analysis with the Decision Tree algorithm. The result showed the percentage of importance of variables using the Decision Tree algorithm. It was found that Source Port (18%), Init_Win_bytes_forward, Destination IP is 192.168.50.4, and Max Packet was (17%) each, Average Packet Size (14%), Flow IAT Min (12%), Fwd Packet(s) (5%), and the remaining variables were less than 5%.

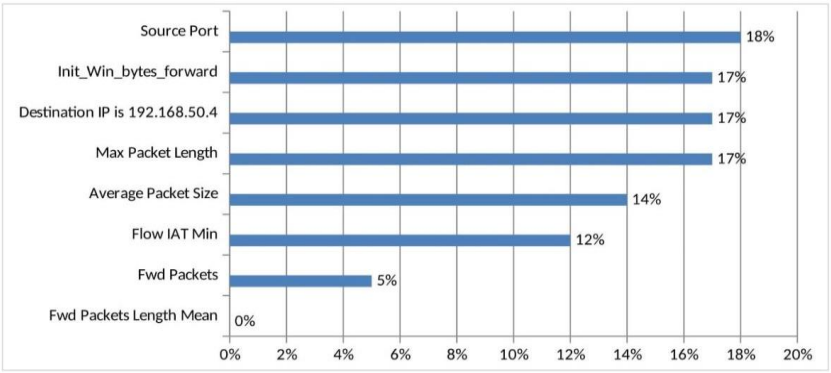

Figure 6. Importance of variables using Decision Tree.

The below given Table IV shows the confusion metrics using the Decision Tree where, the results show the actual and predicted values for the different packets used in the dataset received by the server such as Syn, UDP, NetBIOS, MSSQL, Portmap, LDAP, BENIGN, UDPLag.

TABLE IV. CONFUSION METRICS USING DECISION TREE

	Predicted								
Actual	**Syn**	**UDP**	**NetBIOS**	**MS SQL**	**Portmap**	**LDAP**	**BENIGN**	**UDPLag**	
Syn	100%	0%	0%	0%	0%	0%	0%	0%	100%
UDP	0%	84%	<1%	14%	0%	0%	<1%	1%	100%
NETBIOS	0%	0%	86%	0%	14%	0%	0%	0%	100%
MS SQL	0%	<1%	0%	98%	0%	0%	0%	<1%	100%
Portmap	0%	0%	66%	<1%	33%	0%	<1%	0%	100%
LDAP	0%	0%	0%	0%	0%	100%	0%	0%	100%
BENIGN	0%	0%	0%	0%	0%	0%	100%	0%	100%
UDPLag	0%	0%	0%	0%	0%	0%	0%	100%	100%

7) Results of predictive analysis

The implementation of the above-proposed method yielded pretty decent results. It is observed that for the prediction of type of incoming packet on the cloud server Gradient Boosted Tree gave better results as compared to the Decision Tree. This can be made clearer by the below given screenshot Fig. 7 which clearly defines that GBT algorithm's performance was nearly perfect based on the ROC AUC performance metric.

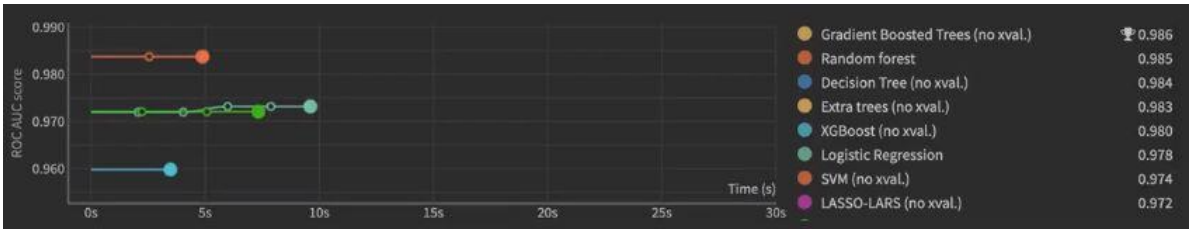

Figure 7. Screenshot showing the result of GBT based on ROC AUC.

The overall results show that the process of predicting the type of incoming packet on the server in the cloud environment proved to be efficient based on the above-mentioned results. Finally, the model deployment is done on the AWS cloud environment using the AWS services.

V. Conclusion and Future Scope

In this research, a trusted virtual data center is created in the cloud environment using the AWS service, such as EC2. Over this trusted virtual data center, the researcher predicts the type of incoming packet on the virtual server created in the cloud environment and finds out whether the incoming packet is malicious or non-malicious. So, based on the above-mentioned prediction, a prediction model is proposed which uses the SEMMA framework. Also, a dataset has been collected, prepared, and evaluated based on various performance metrics such as accuracy, precision, recall, F1 score, log loss, and ROC AUC. As per the dataset, the researcher has decided to use the Supervised Machine Learning algorithms of predictive modeling to make a prediction. Finally, two techniques have been used by the researcher for making optimal predictions; Gradient Boosted Tree and Decision tree. A comparative result of both techniques on the same dataset has been defined in the analysis results.
A nearly perfect, efficient, and optimal prediction model is achieved by the Gradient Boosted Tree algorithm with an accuracy of 88% and ROC AUC of 99%.

In this research, the researcher has tried to bridge all research gaps identified related to the security parameter on the trusted virtual data center. The current research process has identified a few more cavities that need to be addressed by future research scholars through their novel contributions.

The current research has proposed a prediction model which identifies whether the type of incoming packet on the cloud server is malicious or not. Further predictions can be made on the identification of the kind of attacks that can hit the cloud server by future research scholars.

Further, research can also be done to optimize the internal working of the trusted virtual data centre, such as when the file is uploaded by the user on the cloud server; the techniques that can be optimized are file storage, load balancing, file partitioning, data de-duplication, file compression, digital signature generation, encryption, decryption, chunk management, feedback control mechanism so that the complete system works in an efficient manner.

Conflict of Interest

The authors declare no conflict of interest.

Author's Contribution

Ashmeet Kaur Duggal carried out this research work under the supervision of Meenu Dave. Ashmeet Kaur Duggal performed the experimental analysis for this prediction model. Meenu Dave provided critical feedback and helped shape the final version of the manuscript, and both the authors approved the final version of the manuscript.

References

[1] Data Growth, Business Opportunities, and the IT Imperatives. [Online]. Available: https://www.emc.com/leadership/digital-universe/2014iview/executive-summary.htm

[2] L. Qian, Z. Luo, Y. Du, and L. Guo, "Cloud computing: An overview," in *Proc. IEEE International Conference on Cloud Computing, Part of the Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 2009, vol. 5931, https://doi.org/10.1007/978-3-642-10665-1_63.

[3] Trusted Virtual Data Center. [Online]. Available: https://researcher.watson.ibm.com/researcher/view_group.php?id=2855

[4] Overview of Amazon Web Services. [Online]. Available: https://docs.aws.amazon.com/whitepapers/latest/aws-overview/introduction.html

[5] Amazon EC2. General Purpose. [Online]. Available: https://aws.amazon.com/ec2/instance-types/

[6] S. Berger, R. Cáceres, D. Pendarakis, R. Sailer, E. Valdez, R. Perez, and D. Srinivasan, "TVDc: Managing security in the trusted virtual datacenter," *ACM SIGOPS Operating Systems Review*, vol. 42, no. 1, pp. 40–47, 2008.

[7] S. Berger, R. Cáceres, K. Goldman, D. Pendarakis, R. Perez, R. Rao, and E. Valdez, "Security for the cloud infrastructure: Trusted virtual data center implementation," *IBM Journal of Research and Development*, vol. 53, no. 4, pp. 1–12, 2009, doi: 10.1147/JRD.2009.5429060.

[8] T. Singh, "The effect of Amazon Web Services (AWS) on Cloud-Computing," *International Journal of Engineering Research & Technology (IJERT)*, vol. 10, no. 11, 2021, doi: 10.17577/IJERTV10IS110188

[9] N. Kewate, A. Raut, M. Dubekar, Y. Raut, and A. Patil, "A review on AWS-cloud computing technology," *Ijraset Journal For Research in Applied Science and Engineering Technology*, 39802, 2022, https://doi.org/10.22214/ijraset.2022.39802

[10] S. Shetty, "Auditing and analysis of network traffic in cloud environment," in *Proc. 2013 IEEE Ninth World Congress on Services*, 2013.

[11] Mldb.ai inc. MLDB is the Machine Learning Database. [Online]. Available: https://mldb.ai/Software 2.0

[12] A. Karpathy. (2017). [Online]. Available: https://medium.com/@karpathy/software-2-0-a64152b37c35

[13] G. Saake, D. Broneske, G. C. Durand, B. Gurumurthy, A. Meister, M. Pinnecke, and R. Zoun. (2019). Advanced topics in databases. Otto-von-Guericke University of Magdeburg. [Online]. Available: https://www.dbse.ovgu.de/-p-578-EGOTEC-ti601v9tahh7ts9gofp6iljnh7/_/0_overview.pdf

[14] I. Sharafaldin, A. Lashkari, and A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018)*, 2018, pp. 108–116, doi: 10.5220/0006639801080116

[15] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed Denial of Service (DDoS) attack dataset and taxonomy," in *Proc. IEEE 53rd International Carnahan Conference on Security Technology*, Chennai, India, 2019.

[16] Canadian Institute for Cybersecurity. Datasets. [Online]. Available: https://www.unb.ca/cic/datasets/index.html

[17] Canadian Institute for Cybersecurity. DDoS Evaluation Dataset (CIC-DDoS2019). [Online]. Available: https://www.unb.ca/cic/datasets/ddos-2019.html

[18] Canadian Institute for Cybersecurity. Applications. [Online]. Available: https://www.unb.ca/cic/research/applications.html

[19] A. Lashkari, G. Gil, M. Mamun, and A. Ghorbani, "Characterization of tor traffic using time based features," in *Proc. the 3rd International Conference on Information Systems Security and Privacy (ICISSP 2017)*, 2017, pp. 253–262, doi: 10.5220/0006105602530262

[20] G. D. Gil, A. H. Lashkari, M. Mamun, and A. A. Ghorbani, "Characterization of encrypted and VPN traffic using time-related features," in *Proc. the 2nd International Conference on*

Information Systems Security and Privacy (ICISSP 2016), 2016, pp. 407–414.

[21] J. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.

[22] T. Hastie, R. Tibshirani, and J. Friedman, *Elements of Statistical Learnin*g, 2nd ed. Springer, 2009.

[23] J. Friedman. (1999). Stochastic Gradient Boosting. Scikit-learn. [Online]. Available: http://scikit-learn.org/