Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X
Volume 3, Issue 8, 2025

SSH ATTACKS DETECTION USING MACHINE LEARNING: COMPARATIVE ANALYSIS OF DIFFERENT ML MODELS

Ali Hamza[1], Hassan Khalid[2], Talib Nadeem Usmani[3], Muhammad Zunnurain Hussain[*4], Muhammad Zulkifl Hasan[5]

[1,2]*Student, Department of Computer Sciences, FAST-NUCES, Lahore, Pakistan.*
[3]*Honeywell, Duluth, Georgia, USA*
[4]*Department of Computer Science, Bahria University Lahore Campus*
[5]*Faculty of Information Technology, Department of Computer Science, University of Central Punjab*

[1]19L-2204@lhr.nu.edu.pk,[2]19L-1181@lhr.nu.edu.pk,
[3]talibosmani@gmail.com,[*4]zunnurain.bulc@bahria.edu.pk [5]zulkifl.hasan@ucp.edu.pk

DOI :https://doi.org/10.5281/zenodo.16957989

Keywords
*SSH Attack, Brute-Force Attack, Convolution Neural Network, Long Short Term Memory, Multi Layer
Perceptron, Decision Tree, Random Forest, Support Vector Machine, Naive Bayes, K-nearest-neighbours, Logistic Regression, PART, OneR, J48*

Article History
Received on 20 May 2025
Accepted on 08 August 2025
Published on 27 August 2025

Copyright @Author
Corresponding Author: *
Muhammad Zunnurain Hussain

Abstract
Given the ubiquity of SSH as one of most famous communication protocol, SSH attacks can be detrimental. Once, the attacker is able to guess the credentials using brute force attack, they can compromise sensitive information, bring down a business, and whatnot; the realm of possibilities is unimaginable. Therefore, there must be some way of detecting SSH attacks so that the administration might prevent these attacks before the attacker can find the credentials. Hence, this project aims to make a comprehensive study of some state of the art machine learning algorithms used for SSH detection and, in turn, compare the results of these algorithms.

INTRODUCTION

The growth of IT technology is rapidly increasing from the last two decades causing the high use of the internet and a number of users worldwide. No doubt this quick progress in IT technology has provided a number of benefits like number of discoveries, transfer and sharing of information, but as every coin has two sides it has increased the threat of Cyber - attacks. Attackers can target some specific or broad targets and launch attacks on them on regular basis. Motivation for attackers can be simple like financial gain, state sponsored attacks or just to threaten the target or many more. It is not possible to completely save yourself from these cyber attacks but for the prevention of these external attacks every organization should have security policies and should take preventional measures. Attackers are the internet worms they are always crawling the internet searching for targets or performing malicious activities. As most of the users or administrators had set the remote access over the server for their ease, so these servers are the beloved target of malicious users. Therefore, every server including Secure Shell (SSH) should be protected from brute force attacks. Due to weak password authentication these get exploited by malicious users. Attackers first keep searching for such servers which have weak authentication then they try to connect to the server by trying various combinations of passwords, this method is called brute-force. Once the attacker gains the access by successful login, then the attacker becomes the administrator and can do whatever he wants. Mostly they use this for malware installation or to launch attacks.

A. What is SSH?

One of the most commonly used communication protocols over the world wide internet is SSH. It stands for "Secure Shell". Most of the developers, webmasters and system administrators use SSH because it uses an encrypted channel. It provides access to a new cloud service using an encrypted communication channel .

SSH was invented and introduced by "Ylonen and Lonvick in Internet Engineering Task Force's RFC 4254" [2]. It is famous in Windows operating systems and Linux devices. This protocol allows users to gain access to an "interactive shell by authenticating through a remote machine" [2]. As well as remote login, "Secure Shell (SSH) provides functions such as remote command execution, X11 forwarding, and TCP/IP port forwarding" [2]. SSH services are not just limited to operation, these services have various scope and can run on a variety of hosts. "An SSH handshake consists of three major sub-protocols: the transport layer, user authentication, and connection" [12]. The two major classified types of SSH attacks out of its various types are "Server attacks" and "non server attacks" [4]. Server attacks are the one that occur due to compromise of SSH server. The remaining are called "non Server Attacks".

B. What is Brute Force Attack?

An attack is when someone repeatedly tries to log in by guessing the username and password combination. In order to learn the username and password combination for an account on an SSH server, brute force attacks are performed by the attackers. Human-generated passwords are innately vulnerable. As the basic passwords are simpler to remember, more people choose basic passwords for their accounts. The user name is occasionally used as the password instead of changing the machine's default one. As a result, the attacker would test various username and password combinations. An attacker can use certain automatic tools or manually make a guess. A database of frequently used usernames and passwords is present in automated tools which the attacker can use to fetch the original password. Automating a brute-force attack is simple. Attackers can start brute-force attacks with minimum information and assistance. There are several "sophisticated brute-force assault tools, including Hydra, Aircrack-ng, John the Ripper, Rainbow Crack, and others" [3]. One of the most well-liked brute-force attack tools is "Hydra, which comes with Kali Linux by default" [3]. If a username and password combination is successful during brute forcing, the attacker can get a login shell with access to the server's filesystem. As a result, the attacker may engage in a variety of unlawful deeds. Some of them involve testing freshly developed malware, installing a bot that launches a DDoS assault, injecting a virus program, or spreading spam. Penetration assaults are the term used to describe this kind of activity. Successful brute force assaults can be used as a starting point for additional attacks by giving hackers access to data, applications, and resources. Dictionary attack techniques are used in brute-force attacks, which test every conceivable string combination in a sequence.

C. What is an SSH Attack?

SSH attacks are described as login attempts to gain unauthorized access by repeatedly speculating on a username and password combination. The exploit takes advantage of the common practice among users to select their passwords from a small domain. The hacker attempts every combination of username and password until the right one is discovered. As a result, these assaults disturb network traffic and corrupts the log files. Attacks against the SSH protocol have increased in frequency as a result of the password authentication mechanism since an attacker may easily "guess or brute force the right credentials provided by a lawful system user or administrator" [2]. There are numerous software libraries available that enable the development of SSH clients, and the SSH protocol is open and clearly specified. Because of this, attackers have been able to build automated attack tools that are simple to use and target a particular service using brute-force techniques.

D. History of SSH Attack

Over the past two decades, SSH services came under a lot of cyber-attacks. Due to this reason, SSH server has become a viable location to perform DoS attacks, disseminate spam, and test new viruses. According to a recent survey report, 50% of SSH servers lack updated packages and have weak user passwords, making them vulnerable. In reality, the majority of administrators lack the technical knowledge about how to stop a misuse from occurring. SSH attacks have been the subject of a huge number of investigations in recent years. Some of them were the outcome of research with an academic connection, while others were the product of companies or information security specialists' efforts. In some of these instances, the research of SSH assaults was a subset of a larger study that mainly involved identifying attacker profiles or monitoring their behaviour once they had gained unauthorised access to a system. The Cisco 2014 annual security report explains that "Although brute-force login attempts are by no means a new tactic for cybercriminals, their use increased threefold in the first half of 2013. In a survey reported by the Ponemon 2014 SSH security Vulnerability Report, 51% of the respondent companies have been compromised by SSH brute force attacks in the last 14 months" [8].

E. Why does it still exist?

SSH is one of most prominent protocols globally. SSH has a lot of benefits as it runs different functions, runs commands that interact with kernel, content of one file sent as it is to the next one and protects shell logs file during the communication. As existing one like telnet are with no encryption which is a threat for information confidentiality. On the other hand SSH makes sure of confidentiality, authentication, and encryption in a remote based manner which is why it still exists.

2. Motivation For Detecting SSH Attacks

There are a lot of benefits that are given to our society due to tremendous internet progress. A lot of services are provided by the internet which includes online banking, shopping, earning, education and bring ease in our life. But due to such a large number of liabilities on it, it has few negating points though. It has also become a source of different illegal activities like data theft and breaching people's privacy. The online information is really crucial as banking, earning and many health systems are connected with these internet IOT devices. That is why there is a high need to manage the security on this vaste internet. Consequently SSH attack detection plays a major role for securing the (internet) communication network. In order to have a protected, effective and non-breachable network, detecting SSH attacks plays an essential role. Therefore, a lot of relevant machine learning models can be trained in order to identify the malicious, unauthorized access, activities ,vulnerabilities and exploitation.

3. Methodologies (Literature Review)

In this section we shall discuss the relevant ML algorithms used for detecting SSH attacks. But first, we must cover what is machine learning and what is the importance of using ML algorithms.

A. What is Machine Learning?

"Machine learning is a subset of artificial intelligence that enables computers to learn without being explicitly programmed" [6]. "The purpose of machine learning is to create research systems that can learn from data" [3]. These systems have the capacity to identify intricate patterns and reach informed judgments based on prior knowledge. Machine learning efficiently completes each stage in a maintained workflow to automate processes. Unsupervised and supervised learning numerous techniques are the two main types of machine learning. From tagged data that has already undergone evaluation, supervised learning (SL) creates models that link inputs to desired results. The SL algorithms such as "Decision Tree, Random Forest, SVM, and Naive Bayes [3]" are frequently used for SSH

detection. Unsupervised learning is built on machine learning's capacity to identify and provide hints about challenges that arise in everyday life. Unsupervised learning "relies on the arithmetical method of density approximation" [3] because there are no training samples. These successes have influenced the use of machine learning across several industries. This approach allows intrusion detection systems (IDS) to be upgraded using machine-learning modules in a variety of domains, including cybersecurity. "Machine learning algorithms are becoming effective in intrusion detection systems due to their real-time reaction and adaptive learning process" [6]. They represent unmatched option in comparison to traditional rule-based
algorithms.

B. Machine Learning Algorithms for SSH Attack Detection

1. Convolution Neural Network (CNN)

"CNN is a type of neural network model that works on the principle of weight sharing" [6]. It assigns each node a specific weight according to the correlated features. CNN works best for processing two dimensional data such as images, video file and matrix data. Therefore, it can be used in SSH attack detection as the data is two dimensional. CNN uses multiple feature extraction techniques by the help of hidden layers. It helps in learning the representation of data. The architecture of CNN basically consists of 3 stacks layers known as "the convolution layer, the pooling layer and the fully connected layer" [10]. A lot of CNN models have a basic structure of alternative layers of convolution and pooling. The ending layer is a small fully connected layer usually connected with a sigmoid function or softmax.

2. Long Short Term Memory (LSTM)

"LSTM is a type of recurrent neural network. It was introduced in 1997 by Hochreiter and Hochreiter " [7]. LSTM basically works by a feedback mechanism. Unlike CNN it can process multiple input data such as speeches and videos data. For SSH attacks, LSTM uses 4 stacked layers which are alongside each other and a random value is assigned to each layer. The final output is a thick layer with the softmax or sigmoid function attached to it.

3. Multi Layer Perceptron (MLP)

An ANN model which uses feedforward learning is known as Multi layer perceptron (MLP). "The input layer, the hidden layer, and the output layer are the three node layers that make up an MLP" [7]. With the exception of the first input node, all other nodes of MLP are neurons. These all neurons uses a "non-linear activation function. During training, MLP employs the supervised learning technique known as backpropagation due to its multiple layers and non-linear activation" [7]. MLP is not similar to the working of a single layer perceptron. Those data inputs that can't be separated linearly can be separated by using MLP.

4. Decision Tree (DT)

"A well-known machine-learning classifier built with a treelike form is called a decision tree" [6]. It has internal nodes that stand in for attributes, as well as branches and leaf nodes that stand in for the class label. The root node is first chosen, which is a notable property for data separation, in order to create classification rules. The route from the root node to the leaf node is then decided. As the input data is entered, the decision tree classifier recognises associated attribute values and generates decisions as output. By putting a number of basic tests on the provided topic, they produce predictions.

DT requires less data cleansing than other algorithms do. Instead, DT might have a problem with overfitting, "which can be fixed with Random Forest, and the computational difficulty might rise for more class labels" [6].

5. Random Forest (RF)

Under the examples of supervised learning algorithms, Random Forest is another popular machine-learning classifier. Random forest is similarly utilised in machine learning classification issues. Two asymmetric steps make up the classifier's process. The asymmetrical forest is built using the supplied dataset in the first phase, and the prediction is made using the classifier that was built at the second half. As the decision trees were used to train the different models in a randomized manner to lessen correlation between them. RF provides a solid feature importance estimate, it predicts output with high accuracy, and requires little training time. In contrast, RF is fundamentally less usable than a single DT, and training several deep trees can be computationally expensive.

6. Support Vector Machine (SVM)

The next ML model is called the "Support Vector Machine" (SVM) as it seeks to identify the optimal "decision boundary, or hyperplane, that divides n-dimensional space into distinct groups. Its goal is to identify a plane with the greatest margin so that subsequent data points can be identified with greater assurance" [7]. SVM is helpful in high dimensional spaces and functions well when there is a distinct margin of separation between classes. When we have a large dataset, SVM training takes a lot of time.

7. Naive Bayes (NB)

In classification or prediction issues, the probabilistic machine-learning classifier Naive Bayes is frequently used. Based on the Bayes theorem, it works by figuring out how likely it is to identify or forecast a specific class in a given dataset. Class and conditional probabilities are two of its contents. Class probability is the proportion of each instance of a class to all other instances. A class's sample occurrence divided by each characteristic occurrence is the class's conditional probability. Every attribute is assumed to be asymmetric by the Naive Bayes classifier, which also considers the associations between these attributes. The requirement that the predictors should be independent in this model is the biggest drawback despite being quick and simple to use. The predictors are dependent in the majority of real-life scenarios.

8. K-nearest-neighbours (KNN)

An algorithm that can be lazy sometimes or involves an instance learning algorithm is known as K-nearestneighbours (K-NN). This algorithm postpones the creation of the prediction models until after testing. The training examples are stored in the memory by K-NN. "A new instance's distance or similarity to all training instances stored in memory is calculated" [7] while predicting the class of that instance. The method determines the algorithm's class based on the value of K. In its classification process, it takes into account three key factors: distance, record set, and K value. It works by figuring out how far sample points are from training points. The closest neighbour is the location with the shortest distance. In some situations, the value of k is used to measure the distance to the nearest neighbour.

9. Logistic Regression (LR)

"A statistical analysis method called logistic regression uses previous observations from a data set to predict an outcome which is binary in 0 or 1 (yes or no) " [12]. A sigmoid function is used in logistic regression to identify the appropriate output class. "By examining the correlation between one or more already present independent variables, a logistic regression model forecasts a dependent data variable" [12].

10. PART

PART is a machine learning algorithm used in WEKA. It is a rule-based algorithm which is based on Separate-andConquer strategy. Only 2 instances for each rule should be defined. By using these settings in WEKA, this algorithm shows outstanding results for detecting SSH attacks.

11 . OneR

OneR is another rule-based algorithm used in WEKA. It shows outstanding results for most of the datasets for SSH attack detection. The numerical attributes that will be passed in parameters are discretized into several intervals. The maximum sample size of the bucket is set to 6 as the results are the same if lower bucket values like 2, 3, 4 or 5 are used.

12. J48

The term J48 is used for the java version of C4.5 decision tree algorithm. Basically, it is based on a top-down strategy, also known as a recursive divide and conquer strategy.The criteria for splitting data is based on Information Gain and Information Gain Ratio. For ideal working of this algorithm, information gain should be greater than the average information gained. "Information Gain is the ratio of gain and specific value of an attribute." [9]. "The instances are divided into subsets, one for each branch that extends from the root node, after which we choose which attribute to split on at the root node and create a branch for each conceivable attribute value" [9]. Based on the training data, a decision tree is constructed in these algorithms. According to the possible values that each branch of the tree can take, “each branch of the tree reflects a feature in the data that splits the instances into additional branches” [9]. The final class label is represented by the leaves. The final feature hierarchy in the tree structure is determined by the C4.5 method using an information gain model that has been standardized.

13 . Flow Analysis Model

Four tasks are performed by this model: calculating “flow features, sub-protocol training, sub-protocol identification, and detecting SSH dictionary attacks” [6]. The first three functions use flow features and machine learning algorithms to pinpoint a sub-transition protocol's points. The final function identifies a specific attack by comparing the interarrival times of allowed packets, and then determines whether an attack was successful or failed based on the presence of a connection protocol. Through machine learning algorithms, the sub-protocol training function develops the identifier model, and the model is then employed in a function to identify subprotocol transition sites. “As a machine learning algorithm, we use Ward Clustering based on Euclidean Distance. When aggregating comparable sub-flows in an N-dimensional space, the clustering algorithm creates hierarchical clusters, and the distance between sub-flows xi and xj indicates how similar they are” [6]. The challenge of anticipating the number of clusters leads to the usage of the hierarchical clustering algorithm.

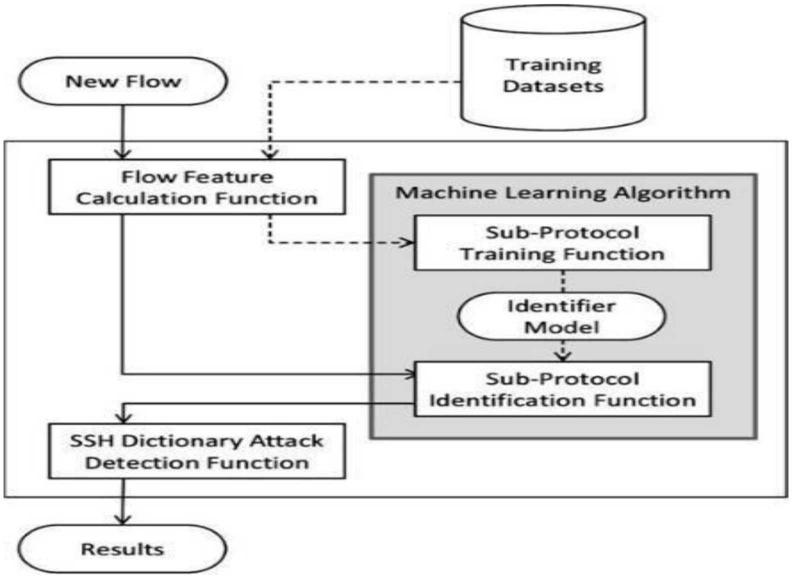

New Flow
Training Datasets
Flow Feature Calculation Function
Machine Learning Algorithm
Sub-Protocol Training Function
Identifier Model
Sub-Protocol Identification Function
SSH Dictionary Attack Detection Function
Results

Figure 1 Flow Analysis of SSH attacks detection

The diagram above shows the working of flow analysis model inorder to detect SSH attacks.

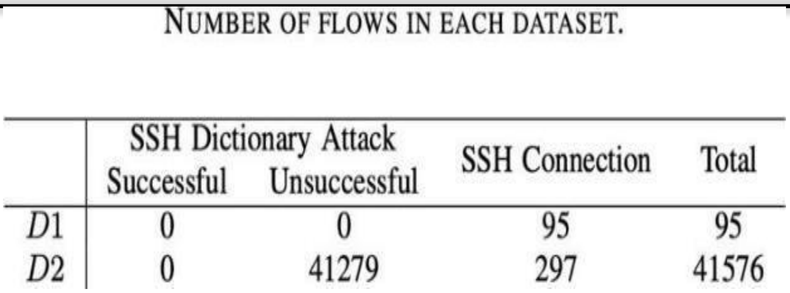
4. Comparative Analysis Of Algorithms

Different experiments were performed on the above mentioned machine learning algorithms for the detection of SSH attacks. The dataset used in those experiments along with the analysis of results are given below:

A. Dataset 1 for Flow Analysis Model

Three kinds of datasets were used for this experiment. For capturing traffic though the observation points, various monitoring tools were used. For tracing the SSH server’s traffic, D1 and D2 were used. The results for the traffic observed along with SSH successful and unsuccessful connection is shown in the table below

:



NUMBER OF FLOWS IN EACH DATASET.

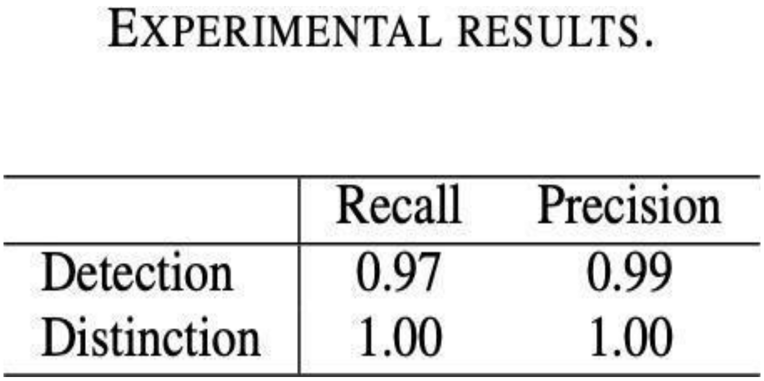
	SSH Dictionary Attack		SSH Connection	Total
	Successful	Unsuccessful		
*D*1	0	0	95	95
*D*2	0	41279	297	41576

Figure 2 Number of Flows in Each Dataset

Results:

According to the experimental findings, this method can detect the success of an SSH dictionary attack with 1.00 precision and 1.00 recall, and can detect an assault with 0.99 precision and 0.97 recall as shown in the table below. Although there were a few mistakes, the results suggest that the flow analysis method may represent a breakthrough in the detection and differentiation of practical SSH attacks.

B. Dataset 2

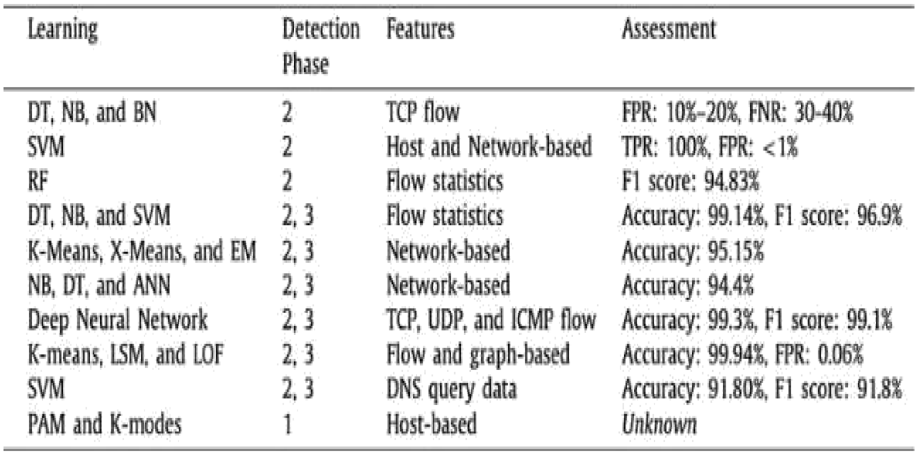


EXPERIMENTAL RESULTS.

	Recall	Precision
Detection	0.97	0.99
Distinction	1.00	1.00

Figure 3 Experimental Results

“The dataset was constructed using a honeypot architecture and an ML-based detection method” [4].

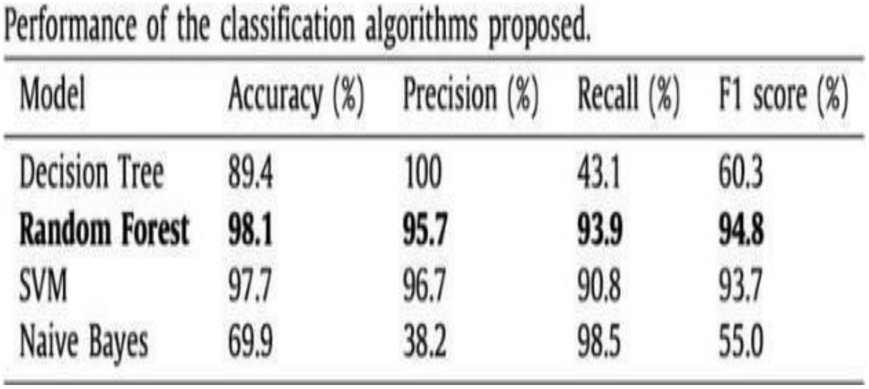


Learning	Detection Phase	Features	Assessment
DT, NB, and BN	2	TCP flow	FPR: 10%-20%, FNR: 30-40%
SVM	2	Host and Network-based	TPR: 100%, FPR: <1%
RF	2	Flow statistics	F1 score: 94.83%
DT, NB, and SVM	2, 3	Flow statistics	Accuracy: 99.14%, F1 score: 96.9%
K-Means, X-Means, and EM	2, 3	Network-based	Accuracy: 95.15%
NB, DT, and ANN	2, 3	Network-based	Accuracy: 94.4%
Deep Neural Network	2, 3	TCP, UDP, and ICMP flow	Accuracy: 99.3%, F1 score: 99.1%
K-means, LSM, and LOF	2, 3	Flow and graph-based	Accuracy: 99.94%, FPR: 0.06%
SVM	2, 3	DNS query data	Accuracy: 91.80%, F1 score: 91.8%
PAM and K-modes	1	Host-based	*Unknown*

Figure 4 Dataset Summary

Results:

The entire SSH session information needed to categorise and create a model for it was extracted from the database by the "ML-based detection (MLD) system" [4]. SVM and RF were the ML algorithms that produced the best outcomes. SVM obtained "90.8% of recall, suggesting that it was able to accurately sort the 90.8% of all positive cases in the dataset, and 96.7% of precision, indicating that of all cases categorised as positive, 96.7% were positive" [4]. In contrast, RF achieved 95.7% precision while producing more false positives than SVM. However, RF achieved 93.9 % recall, showing that it had fewer false negatives.



Performance of the classification algorithms proposed.

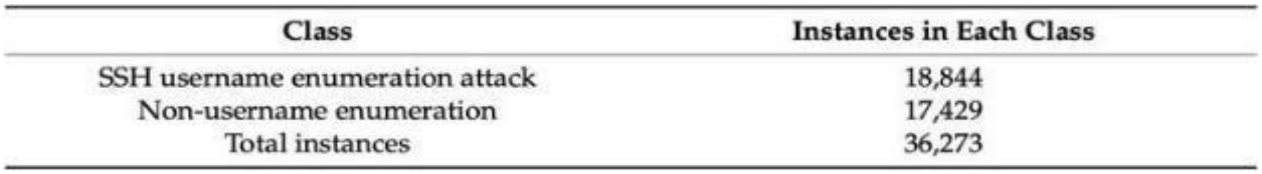
Model	Accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
Decision Tree	89.4	100	43.1	60.3
Random Forest	**98.1**	**95.7**	**93.9**	**94.8**
SVM	97.7	96.7	90.8	93.7
Naive Bayes	69.9	38.2	98.5	55.0

Figure 5 Performance Metrics

In order to prevent the potential consequences in the event that threats are not accurately detected, a lesser number of false negatives is preferred in security issues rather than both classification faults. Despite the fact that both models produce comparable results, we have picked the best classification model produced by the RF classifier since it has less false negatives and principally because it is computationally less expensive than SVM. In conclusion, the best classifier for this dataset is Random Forest. This method has good prediction accuracy and is ideal for contexts where threats are being detected. Because of its high accuracy, speed, and affordability, Random Forest is being used by researchers more frequently for predictive data modelling. This algorithm is the greatest option for hosts with limited capacities who require a real-time response in particular.

C. Dataset 3

"The dataset is gathered from a closed network using network monitoring tools such as tcp dump and Wireshark installed in the data collection points" [5]. A total of 36,273 raw packets with 25 characteristics and label exclusivity were gathered.

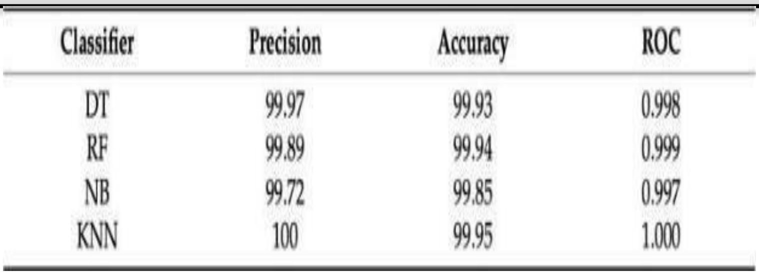


Class	Instances in Each Class
SSH username enumeration attack	18,844
Non-username enumeration	17,429
Total instances	36,273

Figure 6 Class Instances

Results:

We looked at four different machine-learning classifiers. These classifiers included the machine-learning techniques such as knearest neighbour, naive Bayes, random forest, and decision tree. A subset of 80% of the given dataset was used to create four classification models, while the remaining 20% was utilised to test the models. For each classifier, the train-test split ratio was balanced. "Precision, recall, and overall accuracy" [5] were computed as the performance metrics to assess how well the generated models performed.



Classifier	Precision	Accuracy	ROC
DT	99.97	99.93	0.998
RF	99.89	99.94	0.999
NB	99.72	99.85	0.997
KNN	100	99.95	1.000

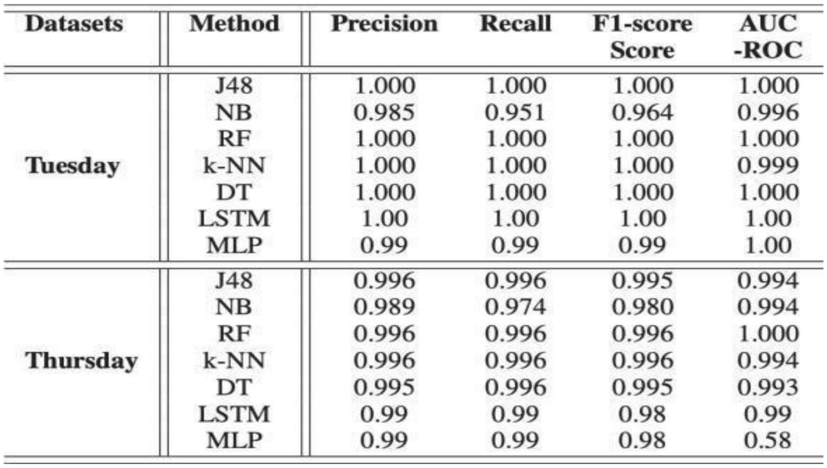
Figure 7 Performance Metrics

By the analysis of results gathered, it was concluded that when ports information is included, the performance of Naive Bayes is much improved. Since the features of naive Bayes are entirely independent, adding port information results in large performance gains. By adding more qualities to the collection of naive Bayes' attributes, the independence of those features can be made up for, which enhances performance.

D. Dataset 4

A popular dataset for SSH attack detection known as CICIDS2017 is used for this experiment. The dataset has been maintained using 11 criteria and includes both benign data and common assaults. Results:

The RMSprop optimizer with a learning rate of 0.0001 is used for the LSTM experiment, while the other parameters are left at their default settings. Due to the multiclass classification, categorical cross entropy is used as the loss function. The dataset was divided into training and testing sets, with training sets using 80% of the data and testing sets using 20%. The fit() function received the validation dataset as an argument. The model was trained on courses for attacks and benign behaviour. The suggested model has a low rate of false positive (FP) and false negative (FN) detection and can accurately classify the attack classes.



Datasets	Method	Precision	Recall	F1-score Score	AUC -ROC
Tuesday	J48	1.000	1.000	1.000	1.000
	NB	0.985	0.951	0.964	0.996
	RF	1.000	1.000	1.000	1.000
	k-NN	1.000	1.000	1.000	0.999
	DT	1.000	1.000	1.000	1.000
	LSTM	1.00	1.00	1.00	1.00
	MLP	0.99	0.99	0.99	1.00
Thursday	J48	0.996	0.996	0.995	0.994
	NB	0.989	0.974	0.980	0.994
	RF	0.996	0.996	0.996	1.000
	k-NN	0.996	0.996	0.996	0.994
	DT	0.995	0.996	0.995	0.993
	LSTM	0.99	0.99	0.98	0.99
	MLP	0.99	0.99	0.98	0.58

Figure 8 Performance Metrics

Now, the MLP model's effectiveness and accuracy in detecting SSH attacks were examined, and the results were contrasted with those of the LSTM model. In addition, various experiments using the MLP model for multiclass classification were done. There were four MLP hidden layers employed. Due to the multiclass classification issue, categorical cross entropy is chosen as the loss function in this MLP experiment using RMSprop with default settings. In training, 80% of the data was used, and in testing, 20%. The fit() method sets the validation. A dropout of 0.1 was utilised to prevent over-fitting.

E. Dataset 5

For this experiment, “four classification algorithms which are 5-Nearest Neighbour (5-NN), two variants of C4.5 decision trees (C4.5D and C4.5N), and Naive Bayes (NB) were selected” [6]. The experiment was run using versions of C4.5 with default WEKA parameter values (designated C4.5D) and with Laplace smoothing enabled but tree-pruning disabled (designated C4.5N). Results:

C4.5N performs better than C4.5D among the two decision trees and has a lower standard deviation. In the C4.5N, the decision tree structure's first level is where numOfNetflows is chosen. “The average packet size is frequently not particularly big and below a threshold value in the tree structure because a brute force attack consists of a series of login attempts” [6].

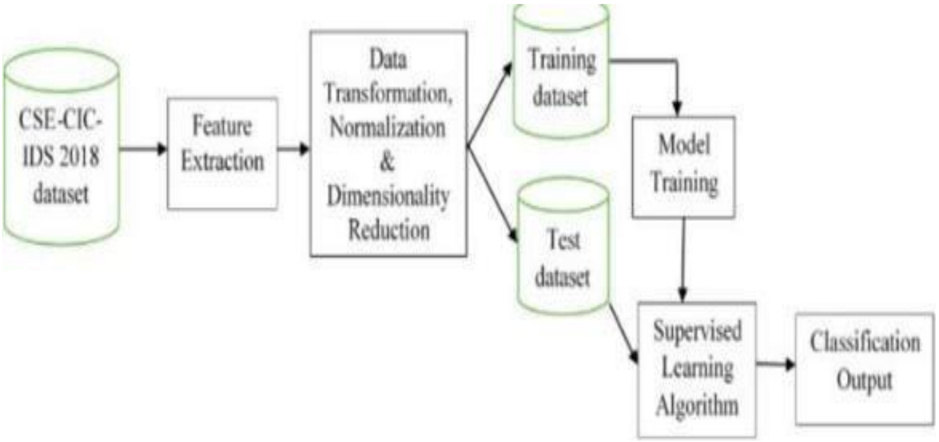


Figure 9 Decision Tree Structure

Results:

The experiment was run using two alternative methodologies, first with all 78 features and then with the fewest features

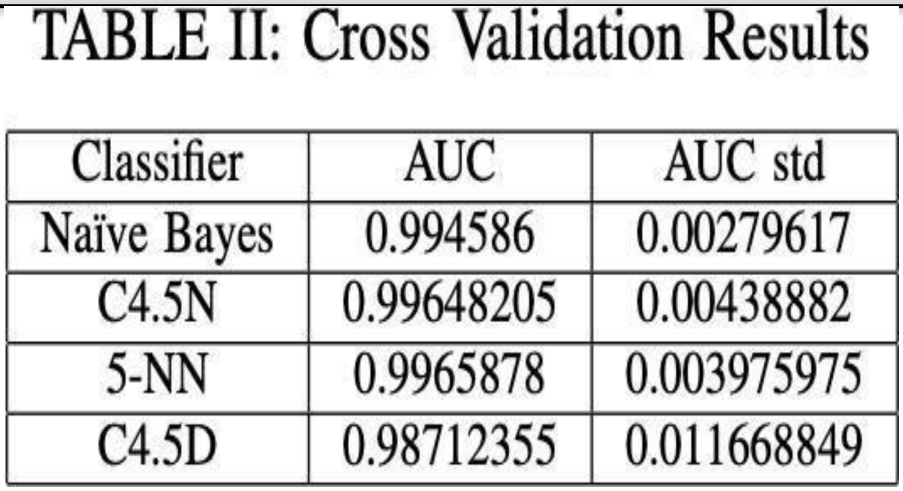


TABLE II: Cross Validation Results

Classifier	AUC	AUC std
Naïve Bayes	0.994586	0.00279617
C4.5N	0.99648205	0.00438882
5-NN	0.9965878	0.003975975
C4.5D	0.98712355	0.011668849

Figure 10 Cross Validation Results

F. Dataset 6
10-fold stratified cross-validation was used in all of the aforementioned algorithms (apart from k-NN) to distinguish between training and test sets of data. The dataset was randomly selected for each fold. The dataset was once again randomly chosen before each fold was applied.
Results:

The two performance indicators with the highest scores are J48 and PART as shown in the table below. J48 uses information gain and entropy to select the best features. It implies that the decision tree algorithms incorporate feature selection. The best performance of J48 might be attributed to its feature selection algorithm. In order to identify the ideal set of features, the feature selection method was used.

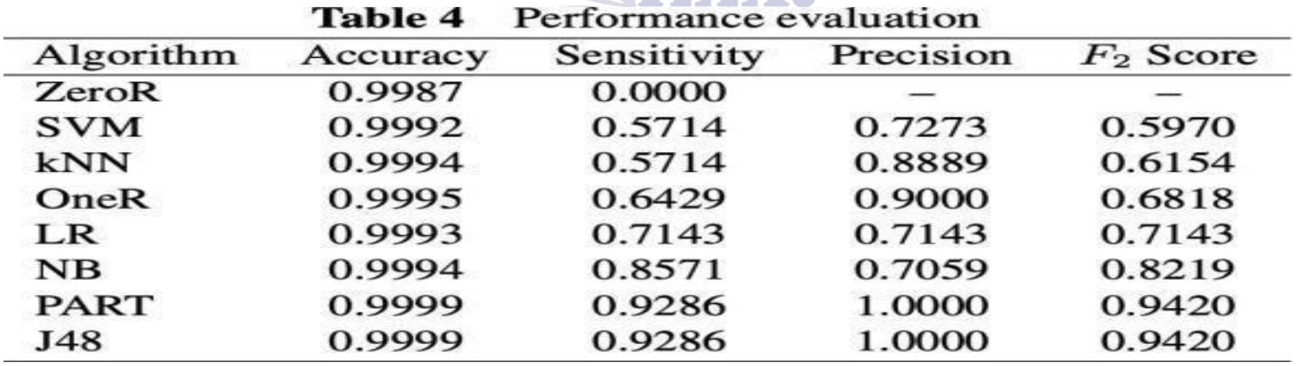


Table 4 Performance evaluation

Algorithm	Accuracy	Sensitivity	Precision	F_2 Score
ZeroR	0.9987	0.0000	—	—
SVM	0.9992	0.5714	0.7273	0.5970
kNN	0.9994	0.5714	0.8889	0.6154
OneR	0.9995	0.6429	0.9000	0.6818
LR	0.9993	0.7143	0.7143	0.7143
NB	0.9994	0.8571	0.7059	0.8219
PART	0.9999	0.9286	1.0000	0.9420
J48	0.9999	0.9286	1.0000	0.9420

Figure 11 Performance Evaluation

G. Dataset 7
The "CIC- IDS 2018 benchmark dataset, which shows realtime network traffic and includes recent assaults both good and bad, was used. This dataset comprises information about 25 individuals based on the HTTP, HTTPS, FTP, SSH, and email protocols"[6] and was created using benign background traffic that was collected using the B-profile system. The network traffic was gathered over the course of five days, with normal activity traffic being dumped on one day and inserted attacks on other days. "Brute Force FTP, Brute Force SSH, Denial of Service, Heartbleed, Web Attack, Infiltration, Botnet, and Distributed Denial of Service are just a few of the different injection attacks" [6]. The figure below shows the working of this model. possible, however accuracy wasn't significantly impacted. The aforementioned experimental data shows that the convolutional neural network model is more effective at detecting SSH-Brute force attacks

than conventional machine learning techniques. It can also be seen in the figure below:

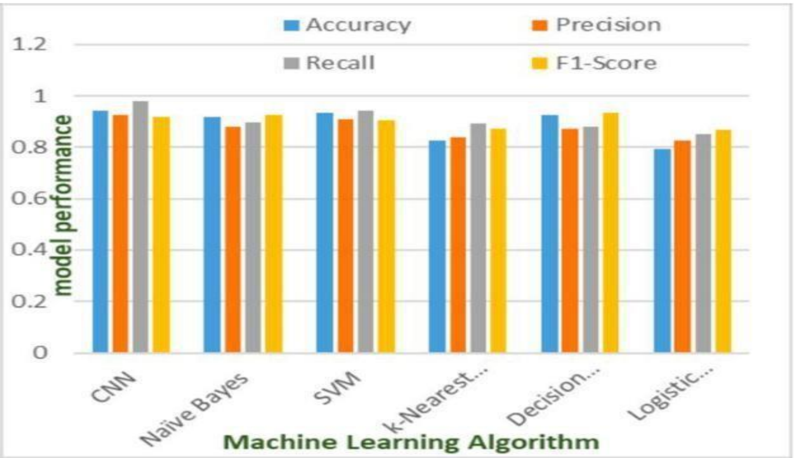


Figure 12 Performance Metrics Graph

5. Metrics Used For Comparing Results

For performing these experiments, following metrics were used in order to compare the results.

1. Accuracy

It means what is the total probability of correct predictions.

Following formula is used to find accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Precision

It means what is the probability of positive predictions detected by the system to be correct. Following formula is used to find precision:

$$\text{Precision} = \frac{TP}{TP + FP}$$

3. Recall

It means what is the probability of the positive cases being correct out of all cases being detected by the system. Following is the formula used to find recall:

$$\text{Recall} = \frac{TP}{TP + FN}$$

4. F1-Score

It shows the relation between precision and recall in detecting positive cases. Following is the formula used to find F1-score:

$$\text{F1 Score} = \frac{2\,.\,\text{precision}\,.\,\text{recall}}{\text{precision} + \text{recall}}$$

6. Conclusion

Our comparative analysis of different state of the art machine learning algorithms shows that using the above mentioned machine-learning algorithms to detect SSH attacks is a very successful method as most of the algorithms obtained 90+ percent accuracy. We also came to know that different machine learning algorithms perform well in various scenarios and datasets. Some algorithms show better results in one dataset as compared to other datasets used. Therefore, an intrusion detection system can be implemented using these algorithms in order to avoid brute force attacks on SSH.

REFERENCES

J. Park, J. Kim, B. B. Gupta, and N. Park, “Network Log-based SSH brute-force attack detection model,” Computers, Materials & Continua, vol. 68, no. 1, pp. 887–901, 2021.

I. Koniaris, G. Papadimitriou, and P. Nicopolitidis, “Analysis and visualization of SSH attacks using honeypots,” Eurocon 2013, 2013.

A. Satoh, Y. Nakamura, and T. Ikenaga, “SSH dictionary attack detection based on Flow Analysis,” 2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet, 2012.

G. K. Sadasivam, C. Hota, and B. Anand, “Classification of SSH attacks using machine learning algorithms,” 2016 6th International Conference on IT Convergence and Security (ICITCS), 2016.

J. T. Martínez Garre, M. Gil Pérez, and A. Ruiz-Martínez, “A novel machine learning-based approach for the detection of SSH botnet infection,” Future Generation Computer Systems, vol. 115, pp. 387–396, 2021.

A. Z. Agghey, L. J. Mwinuka, S. M. Pandhare, M. A. Dida, and J. D. Ndibwile, “Detection of username enumeration attack on SSH protocol: Machine Learning Approach,” Symmetry, vol. 13, no. 11, p. 2192, 2021.

M. D. Hossain, H. Ochiai, F. Doudou, and Y. Kadobayashi, “SSH and FTP brute-force attacks detection in Computer Networks: LSTM and Machine Learning Approaches,” 2020 5th International Conference on Computer and Communication Systems (ICCCS), 2020.

M. M. Najafabadi, T. M. Khoshgoftaar, C. Calvert, and C. Kemp, “Detection of SSH brute force attacks using aggregated Netflow Data,” 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), 2015.

G. K. Sadasivam, C. Hota, and B. Anand, “Detection of severe SSH attacks using honeypot servers and machine learning techniques,” Software Networking, vol. 2017, no. 1, pp. 79–100, 2017.

S. K. Wanjau, G. M. Wambugu, and G. N. Kamau, “SSH-Brute Force Attack Detection Model based on Deep Learning,” International Journal of Computer Applications Technology and Research, vol. 10, no. 01, pp. 42–50, 2021.

G. Fahrnberger, “Realtime Risk Monitoring of SSH brute force attacks,” Innovations for Community Services, pp. 75–95, 2022.

M. M. Najafabadi, T. M. Khoshgoftaar, C. Kemp, N. Seliya, and R. Zuech, “Machine learning for detecting brute force attacks at the network level,” 2014 IEEE International Conference on Bioinformatics and Bioengineering, 2014.

J. Lee and H. Lee, “An SSH predictive model using machine learning with web proxy session logs,” International Journal of Information Security, vol. 21, no. 2, pp. 311–322, 2021.