Design of Ensemble Learning Methods for DDoS Detection in SDN Environment

V.Deepa
Dept. of CSE, School of Computing, Kalasalingam Academy of Research and Education
Krishnankoil, Tamilnadu
vkdeepa94@gmail.com

K.Muthamil Sudar
Dept. of CSE,School of Computing Kalasalingam Academy of Research and Education
Krishnankoil, Tamilnadu
k.muthamilsudar@klu.ac.in

P.Deepalakshmi
Dept. of CSE,School of Computing Kalasalingam Academy of Research and Education
Krishnankoil, Tamilnadu
deepa.kumar@klu.ac.in

***Abstract*— Software Defined Network (SDN) is a new approach to build architecture of computer networks that is dynamic, adaptable, manageable and low cost. The SDN paradigm offers virtualized network services, promoting architecture compatible with the current networks that use infrastructure-hosted services computing. In SDN, switches match for the incoming packets in the flow tables but do not process the packets. Denial of Services (DoS) are attacks in which the network is flooded by a large number of packets sent from machines committed. One class of such attacks is Distributed Denial of Service Attacks (DDoS), where several compromised machines aim simultaneously a target. In this paper, we propose an ensemble technique by adopting different machine learning (ML) algorithms namely K-Nearest Neighbor (KNN), Naive Bayes, Support Vector Machine(SVM) and Self-Organizing Map(SOM) to detect anomalous behavior of the data traffic in the SDN controller. Our experimental results show that the ensemble method in machine learning provides better accuracy, detection rate, false alarm rate than the single learning algorithm.**

Keywords — Software Defined Network (SDN), Hybrid Machine Learning (ML), Distributed Denial of Service (DDoS), K-Nearest Neighbor (KNN), Naive Bayes (NB), Support Vector Machine (SVM) and Self-Organizing Map (SOM).

I. INTRODUCTION

SDN architecture [1] is an emerging network that defeats the downsides of the conventional network. Here the control plane and data plane security are partitioned independently. Data plane in the network like switches and routers just forward the packets dependent on forwarding rules. On the other hand, the control plane is consistently incorporated as a controller which controls the information forwarding in the network. With the help of this strong feature, SDN [2] design is separated into the application layer, control plane layer and data plane layer. In the application layer, the protocols and the applications were placed whereas in network layer the control plane is logically placed as centralized one. Data plane layer dwells on switches and routers.

The major threat on SDN controller is DDoS (Distributed Denial-of-service) attack. The detection of DDoS attacks is a difficult task, since attack packets can be confused with legitimate packages. Another difficulty is the need to analyze the large number of packets which affect the accuracy in detection and response time. This type of attack has the objective of rapidly depleting communication and computational power of a target flooding it with large volume of malicious traffic. The choice of an ideal classification technique for identification of DoS and DDoS attacks on controller is widely analyzed.

In this paper, we propose a DDoS detection mechanism which works in control plane and can identify the incoming packet from a particular source as malicious packet or not. Our contributions in this paper are

- Detection of DDoS attack in the control layer of Software Defined Network controller.
- Designing a machine learning model with a single classifier which can classify the intruded and non-intruded packets.
- Combining machine learning techniques as an ensemble method to detect the malicious and non-malicious packets.

The rest of the paper is organized as follows. Section II represents the related works. Section III discusses the Machine Learning Approaches used as classifiers. Section IV represents the DDoS attack in SDN. Section V is about our proposed algorithm with experimental setup detailed in Section VI. Section VII, VIII discusses about performance of proposed work, results and discussions and Section IX states conclusion and some future work.

II. RELATED WORKS

Zargar et al. [3], stated that the major threat in networking environments is DDoS which prevents the legitimate user to access the service for a long time. Saboor et al. [4] proposed the detection of DDoS attack based on correlation algorithm and IAFV (IP Address Feature Value) algorithm. They used different time series with sliding windows for improving the detection rate.

Mabayoje M.A et al. [5] referred the task of defect prediction through ensemble method and conveyed that ensemble method is more accurate than the single classifiers through different performance measures on the efficiency of learning algorithms. Yavuzet al. [6] studied the Genetic Algorithm (GA) and K-nearest Neighbor (KNN) and combined them as a model to detect attacks. Experimental hybrid system provided more accurate results compared to conventional KNN classifier.

978-1-5386-9353-7/19/$31.00 ©2019 IEEE

Saurav Nanda et al. [7] used Bayesian Network and achieved an accuracy of 91.68 % which indicates that out of 278,598 attacks, their model was able to accurately predict 254,834 attacks. Niyaz, Quamar et al. [8] used deep learning methods to detect the DDoS attack in SDN environment. They had collected the traffic from Home Wireless Network (HWN) scenario. And they got 96.65% accuracy.Gisung Kim, et.al [9] proposed a hybrid learning model to detect the DDoS attack and to protect the OpenFlow switches. Using C4.5 decision tree, they created misuse detection model and then they (1-class SVM) created anomaly detection models to find that their model work well for unknown attacks also. Lohit Barki et al. [10] have used different machine learning techniques such as Naive Bayes, K-nearest neighbor, K-Means, K-medoids to detect the DDoS attack. They found that Naive Bayes model works well compared to other considered algorithms with highest accuracy.

From the literature survey, we understood that the Ensemble Learning is a technique where multiple learners are trained or designed to solve the same problem. Here, the data is given to the classifiers and each classifies the data, finally prediction of the intruder data is detected. The ensemble models may produce high performance in terms of accuracy and false alarm rate.

III. ML BASED CLASSIFIERS

A. *K – Nearest Neighbor (KNN)*

KNN is the method of partitioning the dataset where each set belongs to malicious or non-malicious attack with the nearest mean. To make the real predictions, prior training is required. Here, we used the K-nearest neighbor algorithm to find the K neighbors to be classified, then calculated the probability of each attribute in K neighbors as the weight of the attribute. To determine the output variable, new instance (X) i.e., the dataset is given, the Euclidean distance is measured in terms of identifying the related class variable as given in equation (1).

$$\text{distance}(X1, X2) = \sqrt{\sum_{i=1}^{n}(x1i - x2i)^2} \quad (1)$$

```
function KNN (InputLabelVector)
  trainset, testset=[ ], [ ]
  split=0.67, predict=[ ]
loadDataset ('filename',split,trainset=[ ], testset= [ ] )
for i =1 to trainset do
for j = 1 to testingset do
dist = [ ]
dist+=[pow[(trainset(i)-testset(j),2],math.sqrt(dist)]
neighbo= initialneighborvalue[ InputLabelVector(I,j)]
def initialNeighborValue
sort=sort(distance.neighbo)
   return(sort)
```

Fig 1: Program of KNN classifier

B. *Naive Bayes (NB)*

Naive Bayesian classifier is a statistical as well as probabilistic classifier in machine learning. This means that the presence or absence of a specific feature of a class is not related to the presence or absence of any other feature (absolute independence of features). Here, we find the probability of event A given event B; A and B are events and P(B) ≠ 0.Given the class variable as indicated in equation (2).

$$P(A|B) = P(B|A)P(A)/P(B) \quad (2)$$

```
function NB(InputLabelVectors)
  trainset, testset=[ ], [ ]
  split=0.67 , predict= [ ]
loadDataset ('filename',split,trainset=[ ],testset=[])
  summaries = summarizeByclass(trainset)
predictions.append(summaries)
def summarizeByclass
n=2, m=2
      for i in range(n):
        for j in range(m):
vect= [n][m];
if a[i][j]>0:
   classify (vect[-1]=[ ])
else
classify. append (vect)
return classify
```

Fig 2: Program of Naïve Bayes classifier

C. *Support Vector Machine (SVM)*

Support Vector Machine (SVM) is the supervised learning technique for solving classification problems. Assume training instances and m support vector instances. Suppose support vectors are $\{(v^1, y^1),(v^2, y^2),(v^3,y^3)\ldots\ldots\ldots\ldots(v^m, y^m)\}$, and their weights are $\{w^1, w^2,\ldots\ldots w^m\}$ with test point x.Then equation (3) is used to compute weight w from x to each of the support vectors as follows.

$$X = \sum_{i=1}^{m} y^i * w^i * \text{sim}(x, v^i) \quad (3)$$

where, $\text{sim}(x,v^i)$ is the similarity between x and v^i.

```
function SVM (InputLabelVector)
trainset, testset=[ ], [ ]
  split=0.67
loadDataset ('filename',splited,trainset=[],testingset=[])
for i = 1 to trainset do
   for j=1 to testingtset do
classify = initialclassifyvector InputLabelVectorI,j)]
    end for
  end for
plot.show (classify)
```

```
def initialClassifyVector
  σ = 0.004
  if σ value exists then
        identify intruder packet
        block
    else
        allow the connection
end if
return (value )
```

Fig 3: Program of SVM classifier

D. Self-Organized Map (SOM)

Self-Organized Map (SOM) is an unsupervised learning model which works well for the attack classification based on Artificial Neural Networks (ANN). Majorly used in pattern recognition with elimination of noisy signals, SOM involves mapping of high dimensional representation to low dimensional distribution. SOM reorganizes and builds the map using the input data , available during training of the system and does classification of input vector using the neurons in the map.

If L(*t*) is the learning rate and σ is the neighborhood neuron vector which remains on the same position (can be centered). Best Matching Unit (BMU) moving around based on computation can be the distance to all neurons *i*can be calculated using equation (4)

$$Distance = \sqrt{\sum_{i=0}^{m}(x - w)\text{^}2} \quad (4)$$

where, x is considered as a input vector. SOM has x and major attributes to make reliable predictions for the mapping of points.

```
function SOM (InputLabelVector)
   D={x1,x2.....xm}      //Input vectors
   W={w1,w2,.....wk}     //Weight vectors
Initialize wЄ W to random values, α, t, tmax, x
   for t=1 to tmaxdo
     random x Ɛ D
     d(x,w) = min { d(x,w) | x Ɛ D }
for all w in neighbourhood h do // learn rate = 0.01
update the weight w= w+ α ( x – w)
reduce learning rate α
    end for
  end for
end
```

Fig 4: Program of SOM classifier

IV. DDOS ATTACKS ON SDN

Shin et al., [11] analyzed flooding by DDoS attacks against SDN network. In traditional networks, creating a flow table i.e., flow entry for a new incoming packet needs no additional time to process. In contrast, SDN controller takes new packet to analyze and then make in new flow entry. The flow entries are continuously updated. The comparison of subsequent packets with the other packet adds more time than the average time is processed to determine the intruder packets. To identify attacks, we can check the difference in respond times of first and following packets. So the intruder packets can be eliminated with proper measures.

V. PROPOSED WORK

Ensemble method [12] learns using multiple classifier system and combines several machine learning algorithms that leverage the efficiency of models to achieve better accuracy than of the individual models. An ensemble method combines a set of learners for data analysis.

In order to handle this DDoS attack, we propose a combination of two machine learning based models as KNN with SOM, Naive Bayes with SOM and SVM with SOM. Fig.5 shows the architecture of the proposed model.

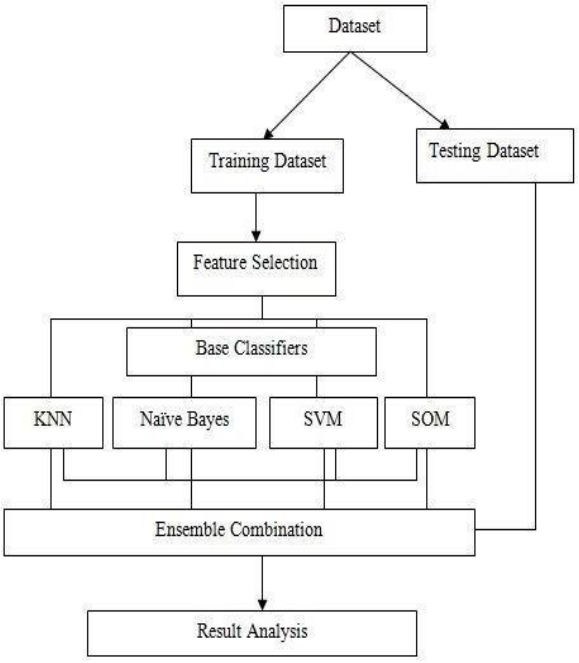


Fig 5: Proposed System Architecture

KNN, Naive Bayes, SVM are the kind of supervised learning whereas SOM is an unsupervised learning technique. Initially, we implemented the KNN, Naive Bayes, SVM and SOM separately. In order to improve the performance, we implemented ensemble method techniques and the algorithms as KNN-SOM, NB-SOM, SVM-SOM[13] with an expectation of the better accuracy, detection rate, false alarm rate.

VI. EXPERIMENTAL SETUP

Mininet [14], an emulation tool helps to create a virtual hosts, controllers, switches, hosts. For the proposed work, we have chosen POX controller [15], a light-weight, python-based controller. Here, it is used for the creation of wireless network setup where one server randomly communicates with another server at a time. The network setup created using MiniEdit is shown in fig 6 (a-c)

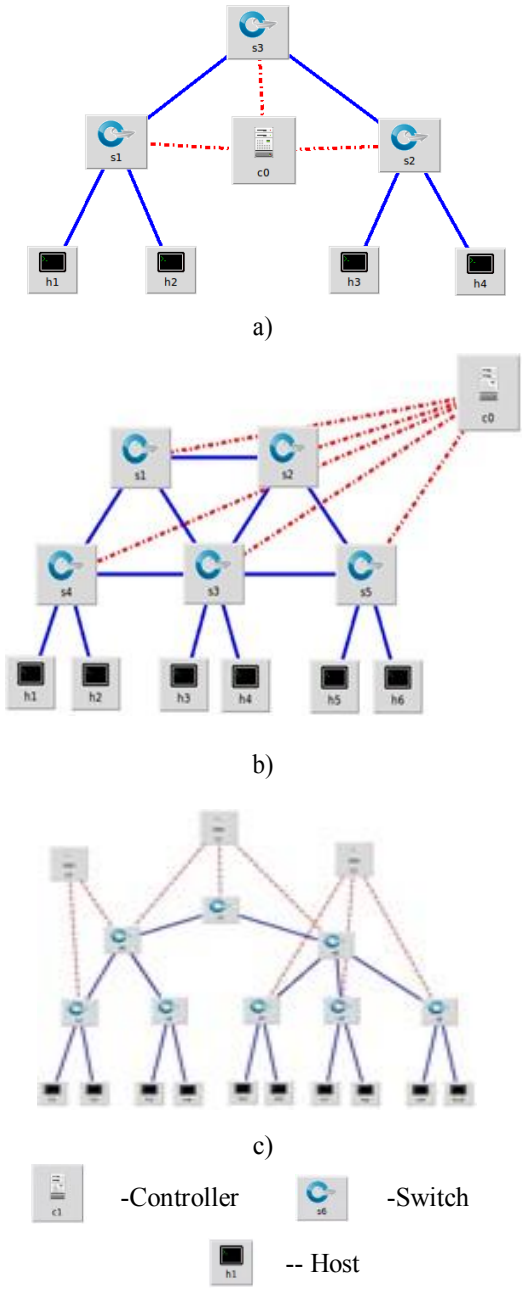


Fig 6 : a),b)and c) represents the network setup

After creating the network setup, the Center for Applied Internet Data Analysis (CAIDA 2016) [16] datasets which contains thousands of records of TCP, UDP, ICMP packets were fed to network. Then we extracted qualitative and quantitative features to determine the performance of each classifier. With the help of traffic classifier module, we separate the traffic based on the time difference of source and destination. Here, the threshold of the time is set to be 0.004sec. If time difference <0.004, packet is transferred from the source to legitimate user. Otherwise it is considered as attack, and then the corresponding connection is closed and rules are updated in the flow tables.

To analyze the performance of proposed system, we have implemented the KNN, NB, SVM and SOM along with our proposed algorithm KNN-SOM, NB-SOM, SVM-SOM to detect the DDoS attack.

VII. PERFORMANCE MEASURES

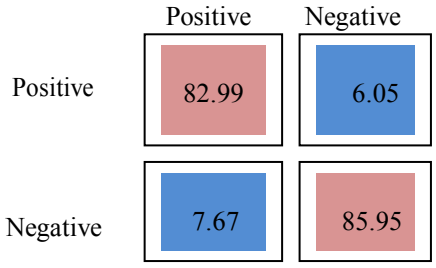
We evaluated the detection system by checking the following performance measures during the attack. To calculate the values of True Positive and False Negative, we compared the values of normal traffic with the injection of malicious traffic. The values of True Negative and False Positive were obtained comparing streams without the presence of attacks. To evaluate the confusion matrix, we calculated the following values:

1) True Negative (TN - True Negative) – A percentage of normal flows that are correctly classified.
2) True Positive (TP – True Positive) – A percentage of attack streams that are correctly classified.
3) False Positive (FP - False Positive) - The percentage of legitimate flows that are incorrectly classified such as attacks.
4) False Negative (FN - False Negative) - The percentage of attack streams that are incorrectly classified as legitimate.

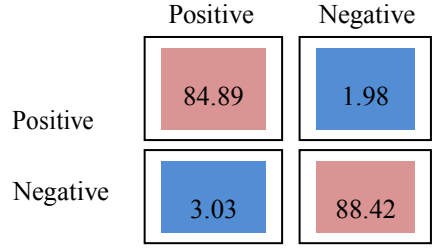
After the analysis with the continuous change in the network setup, the confusion matrix results are tabulated in Table1 and confusion matrix classifiers demonstrate in Fig 7 (a-c).

Table1: Average Results of confusion matrix (%) for classifiers

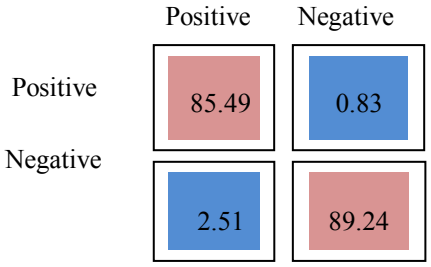
Algorithm	TP (%)	TN (%)	FP (%)	FN (%)
KNN	78.21	79.35	18.06	20.32
NB	81.78	86.05	15.54	18.99
SVM	82.03	87.17	11.67	14.76
SOM	84.33	88.45	6.97	5.55
KNN -SOM	82.99	85.95	7.67	6.05
NB -SOM	84.89	88.42	3.03	1.98
SVM -SOM	85.49	89.24	2.51	0.83



a)



b)



c)

Fig 7 a),b),c): Confusion Matrix classifier for KNN-SOM, NB-SOM, SVM-SOM

Accuracy is defined as the percentage of correctly classified attacks and is one of the widely used measures for the datasets, calculated as in Equation (5)

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} * 100 \quad (5)$$

Detection rate is the number of attacks correctly classified over all predicted attacks as in equation (6)

$$\text{DetectionRate(DR)} = \frac{TP}{FP+TP} * 100 \quad (6)$$

False alarm rate is a number of attacks wrongly classified and calculated as in equation (7)

$$\text{False Alarm Rate(FAR)} = \frac{FP}{FP + TN} * 100 \quad (7)$$

Table 2: Various classifiers performance (%) with feature selection

Algorithm	Accuracy	(DR)	(FAR)
KNN	81.4123	82.24026	18.540
NB	83.9363	84.03205	15.2967
SVM	86.4898	87.5453	11.806
SOM	93.2433	93.3658	7.301
KNN-SOM	92.4887	91.5398	8.192
NB-SOM	97.1904	96.5536	3.313
SVM-SOM	98.1243	97.1477	2.7140

Table 2: represents the various classifiers performance (%) with feature selection. When the basic supervised algorithms are combined with the unsupervised algorithm SOM, the observed results indicate that SVM-SOM combination is able to attain better performance level during the attack classification.

VIII. RESULTS AND DISCUSSION

Fig 8, 9 &10 shows the comparison of detection rate, accuracy and false alarm rate for the proposed simulation. In this simulation, several supervised classification performances of KNN, Naïve Bayes, and SVM were compared with powerful unsupervised classification method Self Organizing Map (SOM). Each supervised machine learning method is already trained with the labeled data. So detecting new kind of attacks will be difficult. In contrast, the unsupervised algorithms can detect the new attack but with an increase in the false alarm rate.

In this system, first the traffic is passed through the supervised models and the traffic is detected. If the new kind of attack is detected, the same modules of supervised learning method are passed through the unsupervised module to detect the traffic and terminate the connection. It is also observed that the performances and stability are good for the SVM – SOM examined cases. The highly flexible technique SVM-SOM[16] which is the simple, linear, discriminate analysis performs better than the non-linear techniques. Our proposed hybrid models provide high accuracy of 98.1423%, high detection rate of 97.1477% and low false alarm rate of 2.7140%.

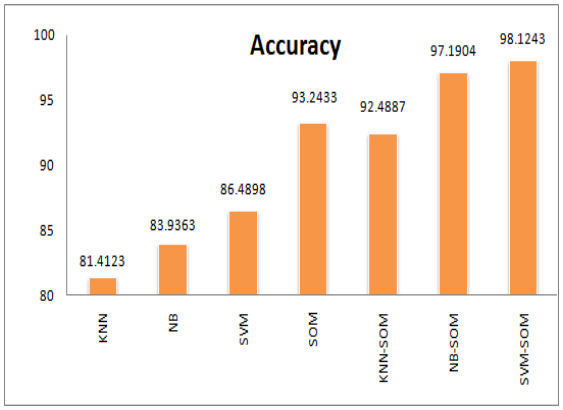


Fig 8: Comparison of Accuracy for KNN, NB, SVM, SOM, KNN-SOM, NB-SOM, SVM-SOM

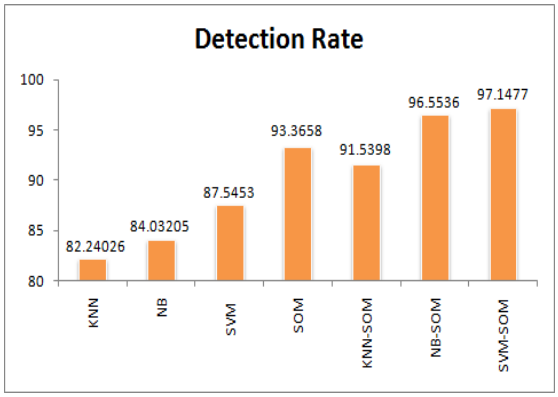


Fig 9: Comparison of Detection Rate for KNN, NB, SVM, SOM, KNN-SOM, NB-SOM,SVM-SOM

The comparative study of accuracy and detection rate are shown in Fig.8 and Fig.9.Our ensemble approach outperforms compared to the original KNN,NB, SVM, SOM classifiers. The SVM-SOM always accounts to be with the highest accuracy of 98.12% whereas the other combined classifiers are less than that of 2-6% and single classifiers are less than that of 5-18%. The detection rate of SVM-SOM is 97.14% which is almost greater than that of others about 4-15%.

Regarding False Alarm Rate, the rate of incorrect alarm generated by each classifier is around 3-18% but the SVM-SOM seems to be only 2.7%.

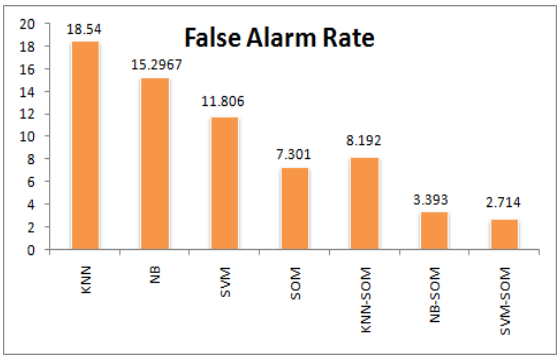


Fig 10: Comparison of False Alarm Rate for KNN, NB, SVM, SOM, KNN- SOM, NB-SOM, SVM-SOM

IX. CONCLUSION AND FURTHER WORK

In this paper, we have implemented ensemble machine learning models to detect the DDoS attack in SDN controllers by imposing the rules in the control plane. We analyzed proposed work based on the three performance metrics such as accuracy, detection rate and false alarm rate. Comparing with supervised algorithm, SOM works well for detection of attacks. But in proposed ensemble machine learning model, SVM-SOM achieved more accuracy, detection rate and low false alarm rate compared to simple machine learning model as well as other combined model. In future, we plan to deal with a new set of data to implement proper choice of classification method and to analyze the performance.

X. REFERENCE

[1] Nunes, Bruno Astuto A., et al. "A Survey of Software-definedNetworking: Past, present, and future of programmable networks."Communications Surveys Tutorials 16.3,2014, pp: 1617-1634.

[2] Kreutz, Diego, et al. "Software – defined Networking: A Comprehensive Survey." Proceedings of the IEEE 103.1 2015,pp: 14-76.

[3] Zargar, SamanTaghavi, James Joshi, and David Tipper. A. "ASurvey of Defense Mechanisms against Distributed Denial of service (DDoS) flooding attacks." IEEE communications surveys& tutorials 15.4 (2013), pp 2046-2069

[4] A.Saboor and B.Aslam, " Analyses of Flow Based Techniquesto Detect Distributed Denial of Service Attacks" In Proceedings of 12th International Bhurban Conference on Applied Sciences& Technology (IBCAST), 13th-17th Jan,2015. pp 354-362.

[5] Mabayoje M.A and Abdullateef Balogun, "A Software defect Prediction: effect of feature selection and ensemble methods", InFUW Trends in Science and Technology Journal, Oct,2018,pp 518-522.

[6] Yavuz,Canbay and Seref SAGIROGLU, "A Hybrid Methodfor Intrusion Detection",14th International Conferenceon MachineLearning and Applications(ICMLA),2015,pp 156- 164.

[7] Saurav Nanda, Faheem Zafari, CasimerDeCusatis, et al. "Predicting Network Attack Patterns in SDN using Machine Learning Approach", In IEEE Conference on Network Virtualization and Software Defined Networks (NFV-SDN),2016.

[8] Niyaz, Quamar, Weiqing Sun, et al . "A DeepLearning based DDoS detection system in Software-defined Networking (SDN)."Journal on European Digital Library, 28thDec, 2017. DOI:10.4108/eai.28-12-2017.153515

[9] Gisung Kim, Seungmin Lee, Sehun Kim "A novel Hybrid attackDetection method integrating Anomaly Detection with misuse Detection", - journal on Expert Systems with Applications, 41(4), March 2014, pp:1690-1700.

[10] Barki, Lohit, et al. "Detection of distributed denial of service attacks in software defined networks." International Conference on Advances in Computing, Communications and Informatics (ICACCI), Sep 2016, pp: 2576-2581.

[11] S. Shin et al., "Attacking Software-defined networks: a first feasibility study," in Proc. the second ACM SIGCOM workshop on Hot topics in software defined networking, 2013,pp 155-157.

[12] V.Bolon-Canedo, N.Sanchez-Marono, A.Alonso-Betanzos, "An ensemble of filters and classifiers for microarray data classification", journal of Pattern Recognition 45,2012, pp: 531–539.

[13] V.Deepa, K.MuthamilSudar, P.Deepalakshmi,"Detection of DDoS attack on SDN control plane using Hybrid Machine Learning Techniques", International conference on Smart System and Inventive Technologies(ICSSIT),ISBN:978-1-5386-5873 4, Dec, 2018.

[14] Mininet.http://mininet.org/.

[15] Pox.http://www.noxrepo.org/pox/about-pox/.

[16] CAIDA Dataset : http://www.caida.org/data/