Machine Learning Based DDoS Attack Detection

Ajeetha G
Department of Computer Science and Engineering
Thiagarajar College of Engineering
Madurai, India
ajeetha.11@gmail.com

Madhu Priya G
Department of Computer Science and Engineering
Thiagarajar College of Engineering
Madurai, India
gmadhupriya@tce.edu

***Abstract*— Distributed denial of service attack has more risk especially in the field of cyber security. The DDoS attack usually arises from the application layer or the network layer where the victims system and the attackers system are interconnected in a network. The effects of these attacks may vary from causing significant failures at the aimed servers to causing inconvenience for users to use a particular service. The DDoS attack brings reputation damage, productivity loss, revenue loss, and even theft for huge business firms and also for banking sectors. Hence there is a need for a good distributed denial detection and prevention technique. The major goal is to deliver optimum solution for these problems using feature analysis. When a heavy traffic flow is experienced at the targeted server, it is important to classify them as an attack or legitimate access. Therefore a novel method has been proposed for the detection of Distributed denial of service attacks through the traces in the traffic flow. A confusion matrix has been generated from these traces. Two classifiers namely Naive Bayes and Random Forest are used to classify the traffic as abnormal or normal, using the normal and attack profile obtained from existing datasets. Naive Bayes algorithm gives better results than Random Forest algorithm.**

Keywords*— *DDoS attack, Naive Bayes, Machine Learning, Random Forest.

I. Introduction

To ensure network security is a necessary entity in every form of business, which includes banking sector, university e-services, social media, industries, E-mail services etc. Recently network services and web services have experienced intruder attacks. The hackers keep on creating new class of DDoS attack that work on both network layer as good as in application layer [1]. The threats mentioned in the above scenarios allow the hackers to prevent an access for a requested service and also slow down the access to a resource in a network.

During the configuration of an attack many infected computers that are under the control of the hackers are used either indirectly or directly to flood the targeted victim or the server using a large amount of information and choke it to prevent the access from legitimate users in using the resources. In most scenarios, the head of the bot computers might know that they are being used by the hackers. In some scenarios, it has only periodic flooding of web servers with heavy traffic to degrade the service, instead of breaking it down completely.

A. DDoS Execution

DDoS can be categorised into various types. But generally it is categorised as attacks on either application layer or network layer. In network layer and application layer, flaws in the protocols are being exploited by the attackers. Smurf attacks, SYN attacks, ping attacks are some major network layer attacks. HTTP GET attack is an application layer attack, which is more advanced than network layer attacks. Hence it is harder to detect attacks in application layer, which is more dangerous [2]. But it is easy to perform attacks in network layer since many online tools are found that are much sophisticated to launch a DDoS attack. These tools can be used to breakdown websites that lack proper configuration of firewalls.

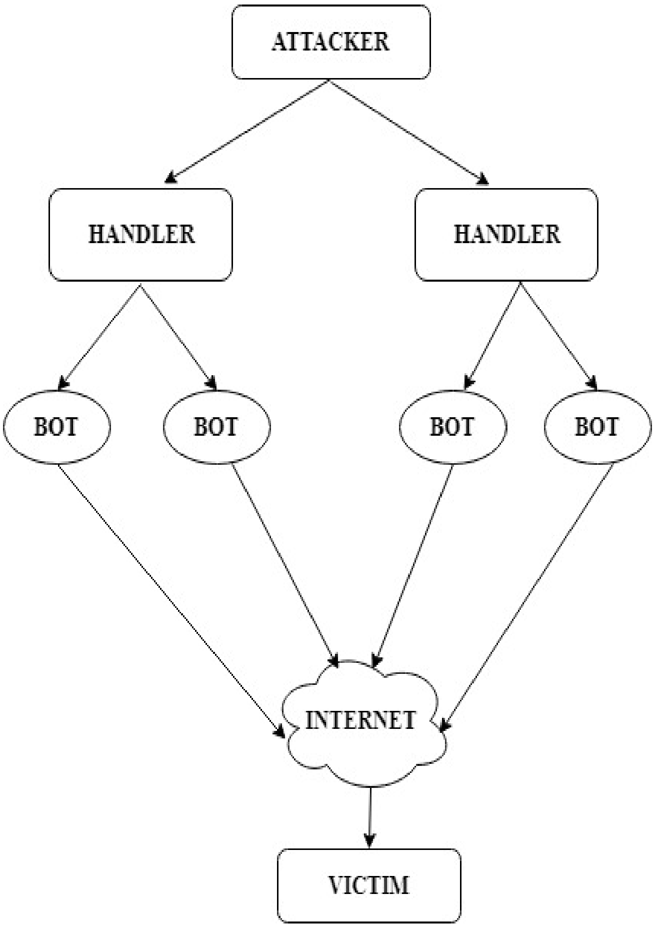


Fig.1. DDoS Attack Illustration.

The DDoS attack generally use layered structure, as shown in "Fig. 1", where the handlers are connected to the attackers system using client program. To perform DDoS attack, handlers system gives command to bot agents for the execution of an attack. The hackers capture the system using several mechanisms including malwares or Trojans. While launching an attack the attacker instructs the handler and in turn the handler commands the bot agents to consume all the resource of victim by flooding with enormous amount of traffics. Hence, the attacks on the DDoS brings several ill-effects into the human community [11]. Some common forms of DDoS attacks were explained in the following section:

B. Smurf Attack

The smurf attack is one amplification form of attack using an ICMP request which broadcast networks address. ICMP is most commonly used for information exchange and to determine the nodes operational status. The victim's IP address is spoofed by the attacker from ICMP request. Since handshaking protocol is not used by ICMP, the final node does not check whether the source node is legitimate or not. Once the request is received by the router, the request will be forwarded to all the connected devices in the network. Once these replies are received by the victims, the attackers ensure their success. A network packet which is spoofed by a smurf program has an ICMP Ping. The echo response and ping messages that are generated is sent towards the IP address of victims system.

C. SYN Flood Attack

As shown in "Fig. 2" the drawback of TCP three way handshaking protocols is exploited by the attacker. The host or victim which receives the SYN packet has invalid source address. In turn the acknowledgement is sent by the server to those invalid IP addresses. All the available servers will be saturated using the packets that have been sent by the attacker in which the legitimate users cannot access the server [3]. By continuously sending SYN requests the attacker is able to capture the targeted server machine.

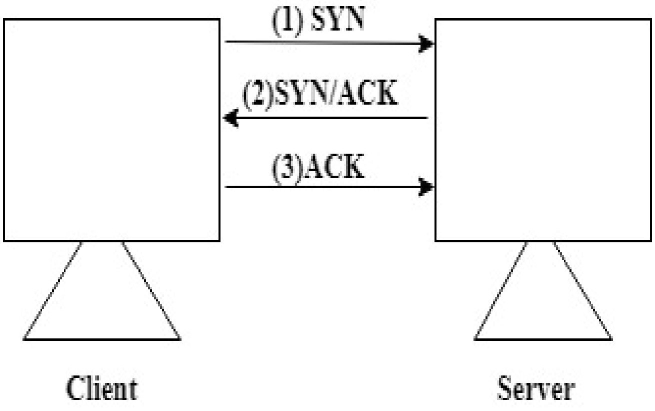


Fig.2. Three Way Handshake TCP.

D. Ping Flood Attack

To test the connectivity of two computers, ping requests are used by calculating the round trip time which is from the first ICMP echo request that has been sent to the reply received by this ICMP request. Usually the targeted network is overloaded with data packets during an attack. For the execution of ping flood the attackers must know the IP address of their targeted host. Based on how their IP address is resolved in its target, it can be classified into three categories.

E. HTTP GET Attack

This attack is also a type of DDoS attack that takes place in the application layer. HTTP POST or HTTP GET request to attack an application or a web server. HTTP attack requires only a less bandwidth to breakdown a targeted server when compare to other attacks. It does not require spoofing, malformed packets or a reflection technique to launch an attack, which makes the detection and blocking of this attack more complicated.

Since HTTP flood attack uses standard URL request it is very difficult to identify the valid traffic, which makes them one among the more advanced non vulnerability security challenge. The most effective mitigation mechanism is to use the combination of profiling methods including the employee progressive security challenge, keeping track of abnormal activity and identifying the reputation of an IP address

II. LITERATURE SURVEY

The literature survey [4] reveals many results the author has designed a mechanism called concurrent security monitoring mechanism, COMO, which is decoupled into two modules namely event collector and event analyzer. SIM frame work has been used to facilitate event collector that allows getting system events efficiently and securely. For facilitating event analyzer they combine virtualization and multicore technology and put them into a trusted execution environment. Lamport's buffer algorithm has been used to address the synchronisation issue in the modules.

A big data based security analytics (BSDA) approach [6] has been proposed for the protection of virtualized infrastructures in cloud towards advanced attacks. This approach constitutes a three phase framework for real time attack detection. In the first phase the periodic data collected from guest virtual machines application and network logs and stored in database. In second phase the features of the attack are extracted from graph on the basis of event correlation and map reduce parser. In the final phase, two-step machine learning algorithm has been used to confirm the attack present. Then, the logistic regression method is involved to compute the conditional probability and using belief propagation method, the belief of attack present is calculated.

To discuss the issue in cloud security and data privacy in its application, the author [5] provides a secure framework in virtual environment. The virtual machines that are created are monitored by a hypervisor or virtual machine monitor or manager. The framework consists of two levels, Layer I is between the hypervisor and the virtual machine's hardware where, a cryptographic encrypted layer is presented. The complete layer is protected along with its hard drive's content. Without a decryption key, the hard drive's data cannot be retrieved and be secured. The layer II is created within the virtual machines and the hypervisor, which creates a Honeypots layer. These Honeypots are used for generating fake environment where the attackers penetrate into the real system but actually their operations will be recorded. Hence, even if the intruders penetrate through Layer I they will be trapped in second layer. The VM's still as the threat of being vulnerable, since the details of attacker are recorded and can be retraced it adds a secure mechanism to the infrastructure.

In this paper, [7] a stochastic model named SGN-based modelling methodology is proposed to quantitatively evaluate the virtualisation security risks for enterprise cloud services. First, the security risk factors for virtualisation in cloud services and the defects of existing evaluation methods were analysed. Second, the virtualisation security risk scenario was built and the SGN model for virtualisation security risk was established in cloud services. The parameterisation of the SGN model was given and the process of virtualisation security risk was simulated. Eventually, the virtualisation security risks in cloud services were evaluated based on the simulation result. This study illustrated that the SGN model can be used efficiently to simulate the dynamic and stochastic nature of the virtualisation security risk in cloud services. In addition, it also offered an evaluation system to assess several significant factors, such as difficulty of attack and economic loss. Based on a series of experiments, this research pointed out that the VM escape risk suffered more security pressures than other three risk factors with time. It also analysed the main risk sources and proposed corresponding measures to reduce or mitigate risks.

In this work, [9] it distinguishes between valid and invalid (malicious) virtual machines. Adopting the combined approach with security software provides required level of protection, immediate application of solutions and make sure that minimum level of security to all the virtual instances with no more overheads and problems. The machine where the Hypervisor is running is the one, who acts as central control point for the purpose of allocating the resources to the virtual instances created before processing starts and de-allocating the same resources from those instances after the processing is over, there by releases the virtual instances. Since it is in the position of creating, allocating and deallocating of resources, it may be vulnerable to attacks

III. EXISTING METHODOLOGIES

The existing methods available to detect the presence of an attack are limited owing to their ability to scale across multiple hosts and in real time threat detection. In BSDA approach [6] it works well when tested with known rootkit and attacks and malwares, but when the data set is evaluated against Livewire the performance [8] overhead incurred is less in case of detection through monitoring the guest virtual machines behaviour.

IV. PROBLEM STATEMENT

Detecting anomalies in network traffic, attaining high prediction accuracy is a major goal in building intrusion detection systems [13] and in designing machine learning algorithms. The most important network attack is DDoS attack which leads to unavailability of a service. Therefore there is a need for detecting DDoS attack intrusions in a network, this detection has to be a learning framework capable of detecting intrusions with trained knowledge. The objective of this work is to train the dataset with Naive Bayes and Random Forest algorithm with the DDoS attack intrusion in the network using trained algorithms.

V. PROPOSED APPROACH

In this proposed work, the DDoS attack has been detected through the application of Naive Bayes algorithm and the random forest algorithm. The Naive Bayes algorithm works by calculating the conditional probability using Gaussian probability density function, the class with maximum probability is chosen and Random Forest works by generating multiple decision trees, the output of this algorithm is the combination of these decision trees [14]. The malware ports and the legitimate ports are collected from the sans and iana websites which is used as the dataset in this work. Sans web portal has a collection of malware dataset and the iana web portal has a collection of legitimate dataset. The final dataset has 2818 observations of 4 variables namely service name, port number, transfer protocol and malware. The prediction is done using the malware variable which indicates 0 if it is a legitimate access and indicates 1 if it is an attack. The dataset needs to be classified as train data and test data. 70 percent of the dataset, which has 2113 observations, is given for training purpose and the remaining 30 percent, which has 705 observations of data is used for testing. Through the application of these algorithms a confusion matrix has been generated. Using this confusion matrix the accuracy has been calculated and a ROC curve graph has been plotted for the obtained accuracy. The ROC curve is plotted based on true positive and false positive rates. For the given dataset, by the application of these algorithms, the Naive Bayes algorithm has got more accuracy than the random forest algorithm.

The block diagram of this approach is given in "Fig. 3". Through the application of these algorithms a confusion matrix has been generated. Using this confusion matrix the accuracy has been calculated and a graph has been plotted based on the obtained accuracy. The Naive Bayes algorithm has got more accuracy than the random forest algorithm.

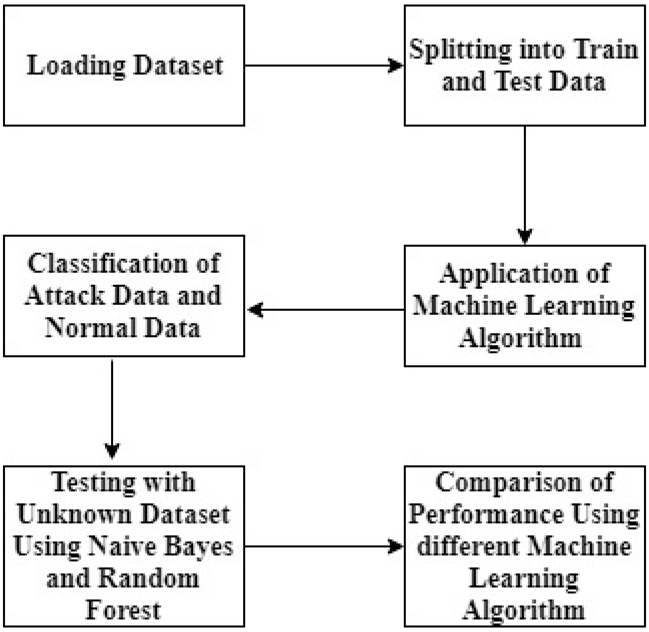

Loading Dataset
Splitting into Train and Test Data
Application of Machine Learning Algorithm
Classification of Attack Data and Normal Data
Testing with Unknown Dataset Using Naive Bayes and Random Forest
Comparison of Performance Using different Machine Learning Algorithm

Fig.3. Block Diagram

A. *Naive Bayes Algorithm*

INPUT:
Dataset D
Predictor Variable P

OUTPUT:
A group of dataset for testing.

STEPS:

1. Read the Dataset D.
2. Calculate the frequency for each attribute.
3. Calculate the conditional probability using Predictor variable P for every class
   r ← dependent class
   q← class variable

$$P\left(\frac{q}{r}\right) = \frac{P\left(\frac{r}{q}\right) \times P(q)}{P(r)}$$

4. Find the class with maximum probability.
5. Generate confusion matrix
6. Draw the ROC curve.

B. *Random Forest Algorithm*

INPUT: training dataset D := (a1, b1), . . . ,(an, bn), features S, and number of trees T.
OUTPUT: A group of data for testing.

STEPS:

```
1.  function RandomForest(D , S)
2.  M ← null
3.  for j ∈ 1,...T do
4.  D(j) ← A bootstrap from D
5.  Mj ← Randomised Tree (D(j), S)
6.  M ← M ∪ {mi}
7.  end for
8.  return M
9.  end function
10. function Randomised Tree (D , S)
11. At every node:
12. s ← small subset of S
13. Split the feature in s
14. return tree
15. end function
```

VI. PERFORMANCE ANALYSIS

Using the primary performance indicators obtained from the confusion matrix, the performance of random forest and Naive Bayes classifiers used in this work was evaluated. This confusion matrix holds the information about predicted and real classifications from the classification models. The performance metrics in these systems are usually evaluated by the information provided by the matrix. For determining four parameters two types of tests has been performed. They are true and false positives, true and false negative [19]. To determine the false positives the dataset is considered from the training phase as well as deployment phase, from this the abnormal number of windows are observed. To determine false negatives, the dataset is given in training phase, in the deployment phase, attack traffic is given as an input to the system and the number of windows classified as normal is observed.

TABLE I. Accuracy of Algorithms

ALGORITHM	ACCURACY %
Naive Bayes	90.90
Random Forest	78.17

From the resultant of the confusion matrix, we have calculated the accuracy, recall and precision of the models, shown in table 1. The overall accuracy was **90.90** percent in case of Naive Bayes shown in “Fig. 4” and in case of random forest it has an accuracy of **78.17** percent shown in “Fig. 5”. Hence from the resultant of these confusion matrixes we could conclude that the Naïve Bayes algorithm proved to be a better classifier when compared with Random Forest Algorithm for DDoS attack detection.

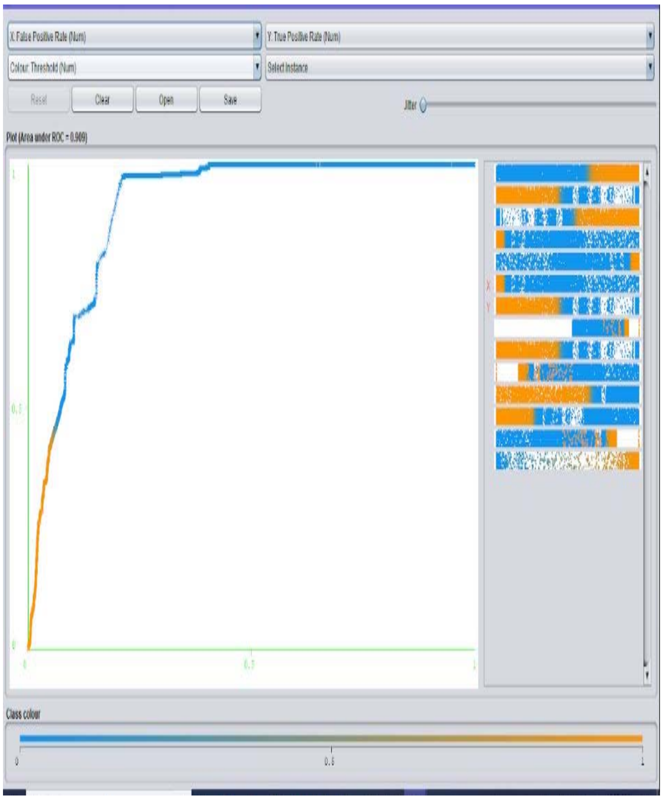
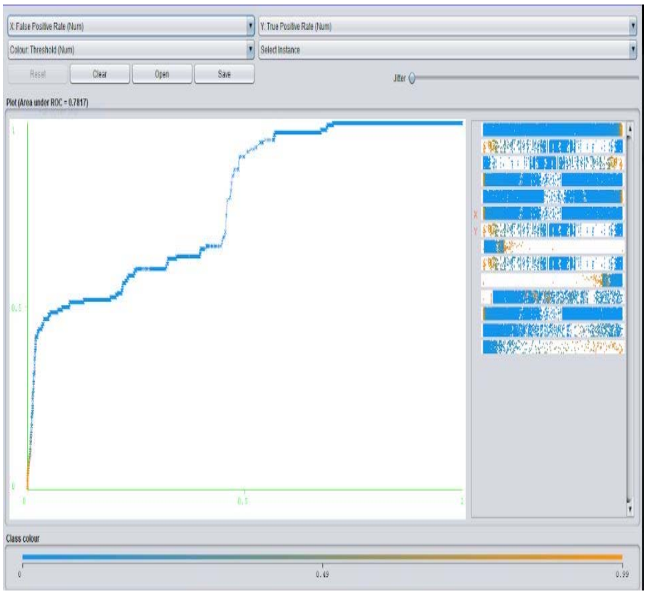


Fig 4: ROC Curve for Naive Bayes


X: False Positive Rate (Num)

Y: True Positive Rate (Num)

Colour: Threshold (Num)

Select Instance

Plot (Area under ROC = 0.7817)

Fig 5: ROC Curve for Random Forest

VII. CONCLUSION

In this paper, we propose a DDoS attack detection system based on machine learning to prevent attacks on the source side in the cloud. Our proposed system is able to detect attacks with high accuracy (90.90%). We applied the Naive Bayes and Random Forest techniques to detect abnormal traffic flow. A comparative analysis of these two machine learning strategies and algorithms was presented. From the obtained results, we concluded that our proposed approach, in which Naïve Bayes algorithm is adopted to detect DDoS attacks, gives more accurate results in comparison with other machine learning algorithm.

VIII. FUTURE WORK

Our Future works includes Monitoring the behaviour of guest virtual machines and collect data that can be used for real time threat detection and to analyze the security risks that have not been identified and provide an effective solution for the identified security risks.

REFERENCES

[1] Yi Xie , Shun-Zheng Yu, "Monitoring the Application-Layer DDoS Attacks for Popular Websites," IEEE/ACM Transactions on Networking Volume: 17 , Issue: 1 , Feb. 2009.

[2] Monowar H. Bhuyan, H. J. Kashyap, D. K. Bhattacharyya, J. K. Kalita, "Detecting Distributed Denial of Service Attacks: Methods, Tools and Future Directions," The Computer Journal Volume: 57 , Issue: 4 , Apr. 2014.

[3] Udaya Kumar N.L., Siddappa M., "Ensuring Security for Virtualization in Cloud services," in International Conference on Electrical, Electronics, Communication, Computer and Optimization Techniques (ICEEECCOT), 2016.

[4] Tian Donghai, JIA Xiaoqi, CHEN Junhua, HU Changzhen "A Concurrent Security Monitoring Method for Virtualized Environments," in IEEE Security Schemes And Solutions, Volume: 13, Issue: 1, Jan. 2016.

[5] Tianwei Zhang Ruby, B. Lee "Monitoring and attestation of Virtual Machine Security Health in cloud computing," in IEEE Computer society, 2016.

[6] Thu Yein Win, Huaglory Tianfield, Quentin Mair "Big Data Based Security Analytics for Protecting Virtualized Infrastructures in Cloud Computing," in IEEE Transactions on Big Data, Volume: 13, Issue: 4, Mar. 2018.

[7] Xiyanyang Luo, Lin Yang Ma, Shaming Chu Hao Dai "Virtualization Security Risk and Solution of Cloud Computing via Divide and Conquer Strategy," in Third International Conference on Multimedia Information Networking and Security, 2015.

[8] Leonardo Richter Bays, Rodrigo Ruas Oliveira, Marhino Pilla "Virtual Network Security: Threats, Countermeasures and Challenges," in Journal of Internet Services and Applications, 2015.

[9] Omnia Abdelrahem, Ayman M. Bahaa-Eldin, Ayman Taha "Virtualization Security: A Survey," in 11th International Conference on Computer Engineering and Systems (ICCES), 2016.

[10] Junnije Lv, Juling Rong "Virtualization Security risk assessment for enterprise cloud services based on Stochastic game nets model," in IET Journals of Information security, 2017.

[11] Aqeel Sahi , David Lai , Yan Li , Mohammed Diykh, "An Efficient DDoS TCP Flood Attack Detection and Prevention System in a Cloud Environment," IEEE Access , Volume: 5, April 2017 .

[12] Uygar Dincalp, Mehmet Serdar Güzel, Omer Sevine, Erkan Bostanci, Iman Askerzade, "Anomaly Based Distributed Denial of Service Attack Detection and Prevention with Machine Learning,", 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 2018.

[13] Ashley Chonka , Jaipal Singh , Wanlei Zhou, "Chaos theory based detection against network mimicking DDoS attacks," IEEE Communications Letters Volume: 13 , Issue: 9 , Sept. 2009 .

[14] Khundrakpam Johnson Singh , Khelchandra Thongam , Tanmay De, "Detection and differentiation of application layer DDoS attack from flash events using fuzzy-GA computation," IET Information Security Volume: 12 , Issue: 6 , November 2018 .

[15] Ademola P. Abidoye , Ibidun C. Obagbuwa, "DDoS attacks in WSNs: detection and countermeasures," IET Wireless Sensor Systems Volume: 8 , Issue: 2 , April 2018 .

[16] Y. Xiang ; Y. Lin , W.L. Lei , S.J. Huang "Detecting DDOS attack based on network self-similarity," IEE Proceedings - Communications Volume: 151 , Issue: 3 , June 2014.

[17] Kübra Kalkan , Gürkan Gür ; Fatih Alagöz, "Filtering-Based Defense Mechanisms Against DDoS Attacks: A Survey," IEEE Systems Journal Volume: 11 , Issue: 4 , Dec. 2017.

[18] Saurabh Bahulikar, "Security Measures for Big Data, Virtualization and the Cloud Infrastructure," in First India International Conference Information Processing (IICIP), 2016.

[19] Ashalatha R, Jayashree Agarkhed, and Siddarama Patil, "Network Virtualization system for Security in cloud Computing," in 11th International Conference on Intelligent System and control, 2017.