VOL. 10, NO. 18, OCTOBER 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

REAL TIME DELAY TOLERANT FEATURE APPROXIMATION TECHNIQUE FOR DDOS ATTACK DETECTION TO IMPROVE NETWORK PERFORMANCE

V. Shyamala Devi[1] and R. Umarani[2]
[1]Department of Computer Application, KSRCT,Tiruchengode, Tamilnadu, India
[2]Department of Computer Application, Saradha Womens College, Salem, Tamilnadu, India
E-Mail: shymalaphd@gmail.com

ABSTRACT

The performance of network highly depends on different metrics like throughput, latency, and security of packet delivered. All these three factors are interconnected and there exists various malicious threats for the packets, which pass through the network channels. Modern adversaries perform various threats like modification, spoofing, sinkholes and many more. To safeguard the network packets from these attacks, the researchers have proposed different approaches with different attributes of packets. Still there are many issues, which do not considered to overcome the problem of, network threats, For example the DDoS approach uses features like payload, TTL, Hop count, Hop addresses. They never consider about the delay that could occur in performing modification attacks. Similarly, there are many cases, which do not consider performing DDoS attack detection. This article aims at performing DDoS attack detection in different forms like Identifying botnet, Delay tolerant features based detection, Flow based approximation and so on. The proposed method uses various features in a different way. In botnet detection approach, the packet traversal patterns are identified and with the help of service access history the common nodes present in the traversal path is identified. Based on network topology and routes available, the presence of botnet controller and compromised nodes are identified to prevent the DDoS attack. In case of delay tolerant approach, the packet traversal path and hop count is used to compute the delay approximation according to available service access history to prevent DDoS attack. Similarly, the flow based approximation technique uses the service access history to compute the legitimate weight of packet being received to identify DDoS attack. All these approaches are interlinked to perform the DDoS attack so that the performance of the network could be improved.

Keywords: DDoS attack, performance metrics, delay tolerant feature approximation.

INTRODUCTION

The growth of internet technology paves the way for the user to access any service independent of location through internet. The requests are fled through the network in the form of data packets to reach the destination where the service is available. In general, the internet technology falls into the group of distributed computing where there exists N number of nodes, which have to perform routing the packets around the network. Any network has an entry point through which the nodes of the network or the service provider could be reached. The component, which are located at the entry point may be of router or gateway and performs routing of packets in both upstream and downstream.

In general, there are two different kinds of attacks present: one is connection based and the other packet based. In the former, there are few nodes which holds many number of connections without any data flow in the channel allocated which spoils the service performance of the service provider. In the later, there are few nodes that sends enormous amount of packets, which cannot be handled by the service providers. This also spoils the service quality of the service provider. More than the attacks there are other attacks like men in middle attack, modification attacks and many more. We concentrate on the denial of service attacks where the service access to the malicious node has to be stopped. To detect the malicious node there are many approaches available and performed in many ways. A simple approach is the host based method where there will be set of host names available with the router or any node. Based on the host name or address the packets comes from such hosts are dropped and the service will be denied and so on. Similarly, there are other approaches available in the literature and discussed later in this paper.

The performance of the network is high depending on the security measures enforced in the routing protocol or in the router or gateway. For example if any service provider wastes the time in verifying the packet signature then the throughput of the servicing node will be decreased. In addition, if it handles the malicious packets in most cases, the service rate and throughput also decrease and the genuine nodes or users will never get access. To make the service availability as higher there must be efficient methods to be enforced to identify the malicious node so that the packet requests will not approach the servicing nodes.

Network safety is input facilities, which provide sanctuary for not only the capital obtainable in a number of the nodes within the network but also provide security for the services obtainable in any of the node on or after the network. The presentation of the system is gauged by as

long as absolute and well-organized service. The performance of the system services are enormously precious by various attacks generate by hateful users from distant off location. Basically present are two dissimilar types of system attacks: packet base and connection base, where both of them decrease the overall performance of the network. In a packet-based attack generate malicious small package towards the repair point, reduces the throughput of the complex by increased treatment of hateful packets. Whereas in association base attacks the presentation of the member of staff serving at table is abridged by simply custody the connection

RELATED WORKS

For the distributed denial of service attacks there are various methodologies proposed by researchers and each have their own merits and demerits and we discuss a few of them here.

Host Based intrusion Detection [1], it presents intrusion detection system and informs the system administrator about potential intrusion incidence in a system. The designed architecture employee's statistical method of data evaluation, allows detection based on the knowledge of user activity deviation in the computer system from learned profile representing standard user behavior.

Network Intrusion Detection System [2], it is embedded in a NIDS Smart-sensor-inspired device under a Service-Oriented Architecture (SOA) approach. Using the embedded NIDS a node can act as an independent anomaly based NIDS. It combines the advantages of the smart sensor approach and the subsequent offering of the NIDS functionality as a service with the SOA use to achieve their integration with other DIDS components. It also addresses the construction of a physical sensor prototype. This prototype is used to carry out the tests, which have demonstrated the proposal's validity, providing detection.

An Activity Pattern Based Wireless Intrusion Detection System [3], it is intended for wireless set of connections. It exploits blueprint recognition technique to representation the practice pattern of genuine user and uses it to detect intrusion in wireless networks. User interest group is monitor and their discriminative skin tone is extracted to recognize intrusion in wireless networks. The discovery module uses PCA method to build up interested arithmetical variables and compare them by means of the threshold resulting from the user's action data. When the variables exceed the predictable threshold, an alarm is raise to alert about a likely intrusion in the system. The innovation of the planned scheme lies in its frivolous design, which require less dispensation and reminiscence resources plus it can be second-hand in real-time environment.

EAACK [4], it is proposed and implemented in a new intrusion-detection system named Enhanced Adaptive Acknowledgment (EAACK) specially designed for MANETs. EAACK consists of three major parts, namely, ACK, secure ACK (S-ACK), and misbehavior Report Authentication (MRA). In order to distinguish different packet types in different schemes, EAACK uses a 2-bit packet header in this scheme.

ANN Based Scheme to Predict Number of Zombies involved in a DDoS [5], it present a complete learn to show the hazard of Botnet-based DDoS attack on request layer, particularly on top of the Web member of staff serving at table. The greater than before incident of such attack have greater than before lately. Botnet-based DDoS attack incident and income losses of well-known company and administration websites are also describe. This provide better sympathetic of the difficulty, present explanation space, and prospect research range to defend alongside such attack professionally.

DDoS Attacks Detection by Means of Greedy Algorithms [6], it focuses on DDoS attack discovery by income of gluttonous algorithms. In this move toward, two algorithms have been projected that is to say corresponding and orthogonal corresponding pursuit. The technique represents the malicious group of students of facial appearance in the form of tree arrangement, which help discovery of refutation of service attack detection competently.

Botnet-based Distributed Denial of Service (DDoS) Attacks on mesh Servers: categorization and Art [7], present an all-inclusive study to give you an idea about the danger of Botnet-based DDoS attacks on submission layer, more than ever on the Web server in addition to the greater than before incidents of such attacks that have evidently greater than before recently. Botnet-based DDoS attack incident and revenue wounded of famous company and government websites are describe. This provides better sympathetic of the problem, current explanation space, and viewpoint research capacity to protect against such attack professionally.

Analyzing Feasibility for Deploying Very Fast Decision Tree for DDoS Attack Detection in Cloud-Assisted WBAN [8], propose classify data removal technique which uses, exceptionally Fast Decision Tree (VFDT) and well thought-out as the majority promising solution for concurrent data mining of elevated speed and non- stationary data stream gathered on or after WBAN sensors and consequently is selected, deliberate and explore for professionally analyze and detect DDoS assault in cloud-assisted WBAN surroundings.

A method of DDoS attack detection using HTTP packet pattern and rule engine in cloud computing environment [9], proposes a method of integration between HTTP GET flooding among Distributed Denial-of-Service attacks and Map Reduce (MR) processing for fast attack detection in a cloud computing environment. In addition, experiments on the processing time are conducted to compare the performance with a pattern detection of the attack features using Snort detection based on HTTP packet patterns and log data from a Web server. The experimental result shows that the proposed method is

better than Snort detection because the processing time of the former is shorter with increasing congestion.

Being explored the intrusion detection methods none of them is discussed for the reactive intrusion detection and proposed a novel approach for the mitigation of DDoS attack which is base on imprudent one.

PROPOSED METHOD

The proposed genuine time holdup tolerant estimate based DDoS detection approach has a variety of stage like Feature removal Botnet discovery, Delay tolerant DDoS discovery, and Flow estimate. Each stage has dissimilar scope and talk about in detail in this part.

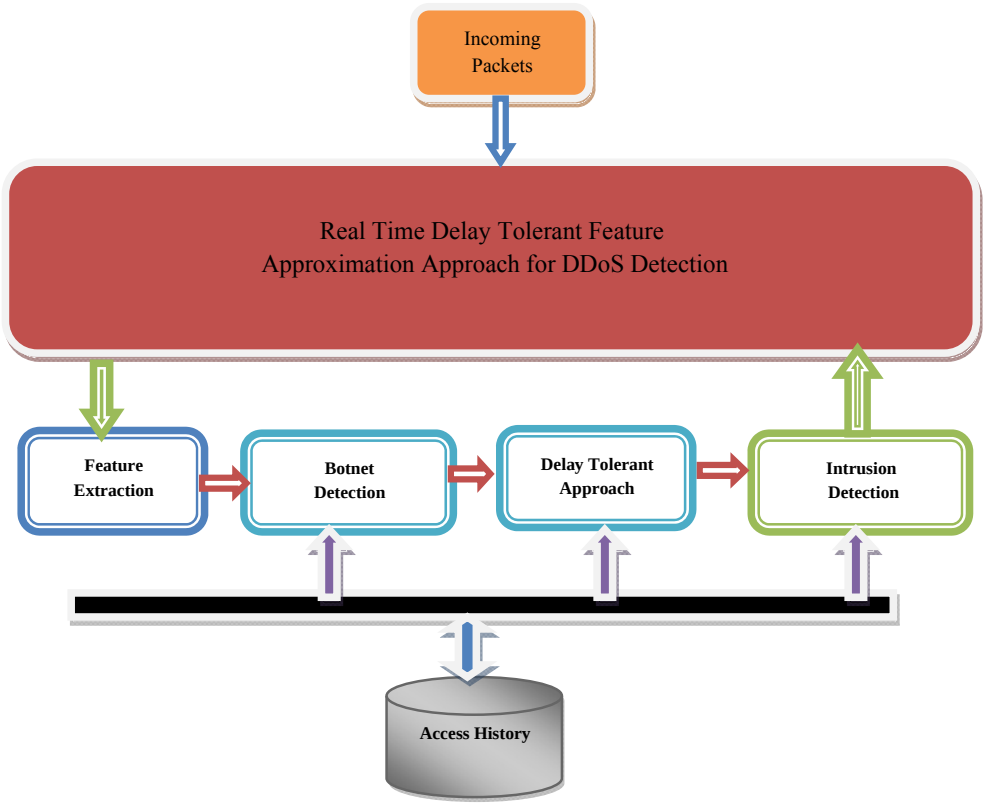

Figure-1. Proposed system architecture.

Feature extraction

The system data packet conventional at the system interface port is handling to the feature removal phase. The small package is rehabilitated into IP little wrap up and its skin like IP deal with and port payload particulars, Hop Count, Hop Details, TTL principles are extracted. From the extract values, it converts into a vector by addition those facial appearances in addition to give up to the Network forensics study stage.

Algorithm

Input: Raw packet- dp
Output: Feature vector Pv.
Step1: Start
Step 2: Read input packet dp.
Step 3: Convert raw packet into IP Packet ip.
Step 4: Read source IP address of packet ip.

Pip = $\int SIP \in IP$

Step5: Read Source port Sp of IP.

Sp = $\int SPort \in IP$

Step6: Extract TTL from IP

TTL = $\int TTL \in IP$

Step7: Extract Hop ip addresses

HIP = $\int \sum_{i=1}^{N} HostAddress \in IP$

Step8: Extract final hop count from IP

Hc = $\int size(HIP)$

Step 9: Convert IP to network packet NP and extract payload details

PL = $\int Payload(NP)$

Step 10: Construct feature vector Pv.

Pv = {Pip, Sp, HIP, HC, TTL, Pl}.
Step 11: Stop.

Botnet detection

The process of botnet identification is performed to identify the botnet controller and a set of nodes, which are compromised to perform denial of service attacks. For any packet being received through the network, the features are being extracted and the extracted features have a hop address through which the packet being traversed. From the history of earlier packets being received and with the current feature vector a set of unique traversal path being identified. From identified set of paths, a set of common hops present in more frequent manner among malicious packets traversal path. In addition, with the help of network topology identifies the set of available paths to reach the service point. By using both available paths and unique traversal path and current host sequence, the packet is being identified as malicious or not. To declare the packet being malicious the packet payload is used.

Algorithm

Input: Access History Ah, Malicious History Mh, Feature Vector Fv.

Output: Packet Type PType.
Step 1: Initialize unique Traversal Path Set UTPS $\int_{i=1}^{k} UTPS \cup (Ah(k) \exists UTPS)$
Step 2: Identify the presence of current path in UTPS.
If UTPS $\exists FV(HostSequence)$ then
Compute Frequency of Current Path in Malicious History MFreq = $\int \frac{Count(\sum[Mh(k))]}{size(Mh)}$
Compute Frequency of Current path in Genuine Access is
GFreq = $\int (Count(Ah(k) - Count(\sum [Mh(k))])/(size(Ah$
If GFreq> MFreq Then
If FV (Payload)>PTh and GFreq(CTime)>GTh then
Drop packet and produce log in malicious history and access history.
Mh = $\sum Mh(i) + Fv$
Ah = $\sum Ah(i) + Fv$
End
Else
Allow the packet and produce log to Access History.
Ah = $\sum Ah(i) + Fv$
End.
Step 3: Stop.

Delay tolerant approach

The delay tolerant approach performs DDoS detection based on the time constraints. For any service-oriented architecture, the request produced by the user has to reach the destination on time with little tolerance. In this phase, the tolerance value is used to classify the packet as genuine or malicious and to identify whether the packet is being modified or reproduced by any of the malicious nodes present on the way. For each packet the traversal time and number of hops traversed to reach the destination is computed. Based on these values an average delay occurred at the network and current traffic condition to predict the packet nature. Based on computed approximated delay the packet is considered to access the service or dropped. (Please check-these lines-something is missing).

Algorithm
Input: Access History Ah, Feature Vector Fv
Output: Packet Status PStatus.
Step1: Identify the logs produced in current time window LCTW = $\int \sum Ah(k)@CTW$
Step2: Compute Average HopCount AHC = $\int \frac{\sum Hops(LCTW)}{size(LCTW)}$
Step3: Compute Average Delay ADelay = $\frac{\sum TT(LCTW)}{AHC}$
Step4: if FvDelay > ADelay then
Perform Botnet Detection.
Perform Flow Approximation.
Else
Ah = $\sum Ah(i) + Fv$
End.
Step 5: Stop.

Flow approximation

The flow approximation technique performs detection of malicious packets and nodes based on their Access history that is produced at different time of their access. The Access history specifies their behavior of access and how they support the service in accessing them and how well they complete the service and so on. Based on these details a legitimate weight will be computed to find out that whether the packet is being legitimate or malicious to provide access to the service and so on.

Algorithm
Input: Access History Ah.
Output: Packet Status Pstatus.

Step1: Find out service history at different Time window.

TAH= $\int_{i=1}^{N} \Sigma Ah \times Tw_i$

Step 2: Compute Frequency of access at each Time window Tw_i

FA = $\int_{i=1}^{K} \frac{\sum(Ah@Twk)}{Size(TAh)}$

Step 3: Compute Average frequency AF= $\frac{FA}{Number\ of\ Time\ Window}$

Step 4: if Current Frequency < Ah

Allow Packet.

Else

Drop Packet.

Ah = $\sum Ah(i) + Fv$

End

Step 5: Stop.

RESULT AND DISCUSSIONS

The real time delay tolerant approach has been implemented and it identifies a set of common hop addresses from the history of traversal paths. The method identifies the presence of botnet and the controller who controls the botnet. The method also identifies the list of nodes, which are compromised with the botnet controller, by identifying the common hop names present in the traversal paths present in the history. The proposed delay tolerant approach has produced efficient results in all the factors of intrusion detection.

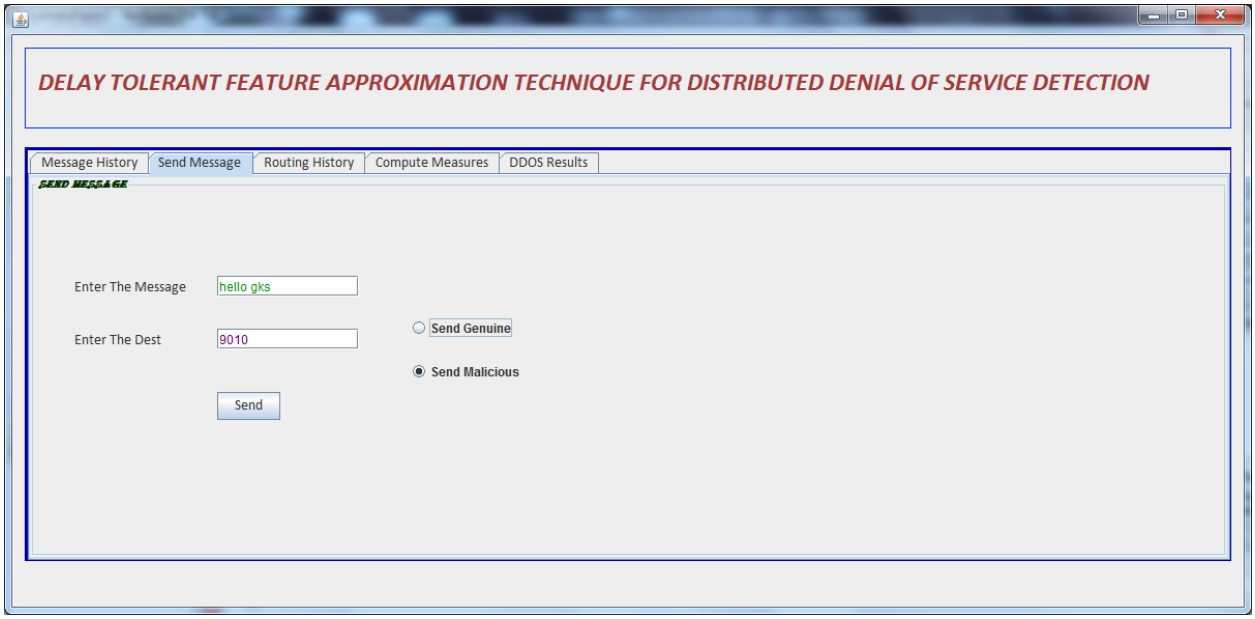

Figure-2. Generating packets.

The Figure-2 shows the generating packets towards particular destination. It is produced at the sender side and also the message has been targeted to the port 9010.

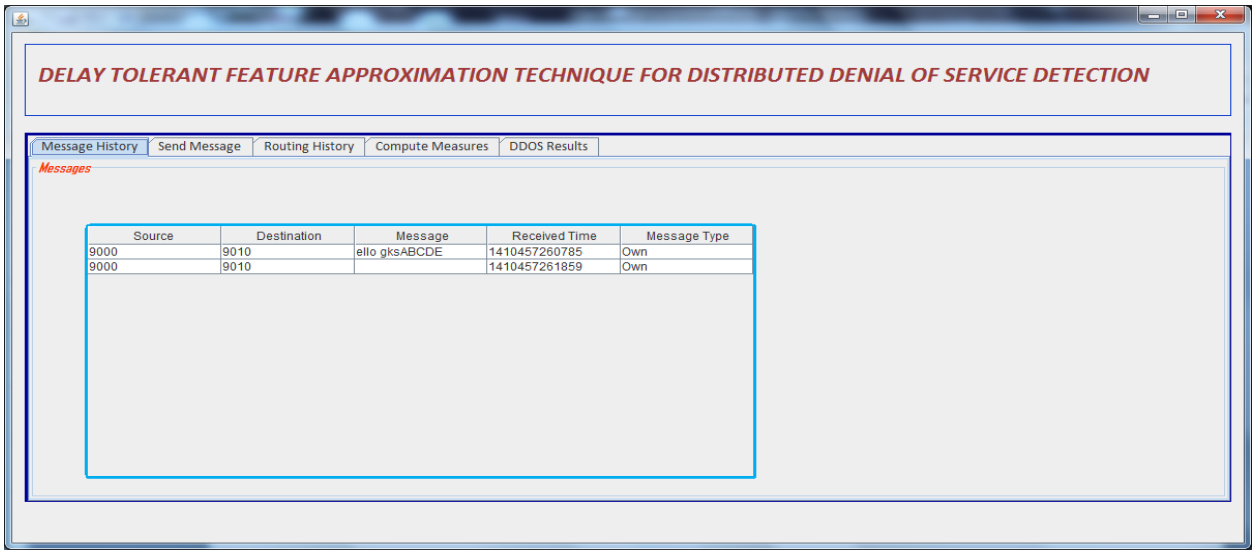

Figure-3. Packets received at the receiver side.

The Figure-3 shows the message details received at the server side or destination side where intrusion detection is performed. It shows that the packet has been originated at port 9000 and has been targeted to 9010. Also it shows the target node to which the message belongs to.

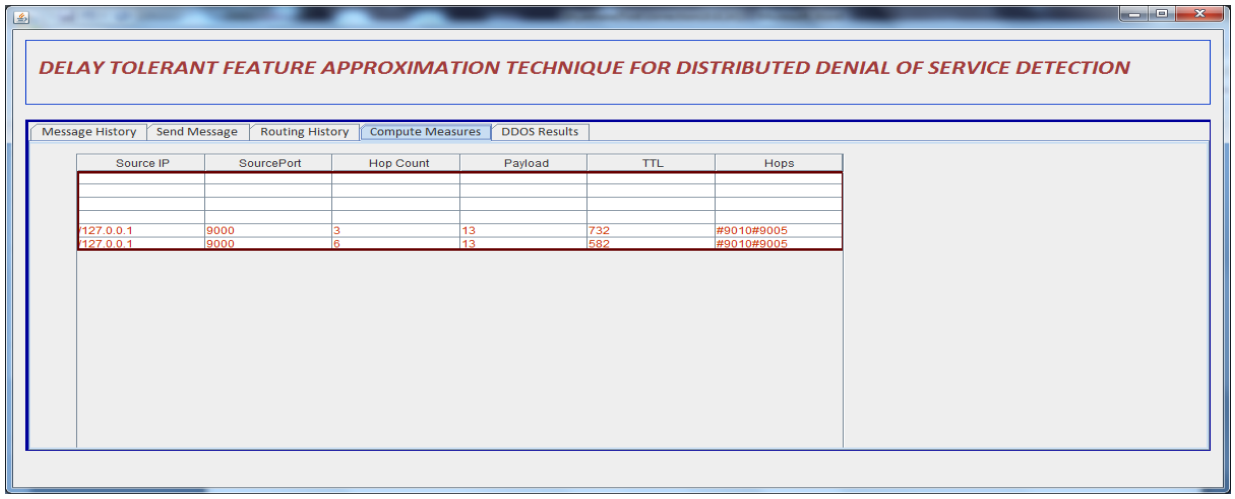

Figure-4. Extracted features.

The Figure-4 shows the features extracted at the receiving side. It shows the details of hop count, payload, TTL and hop values.

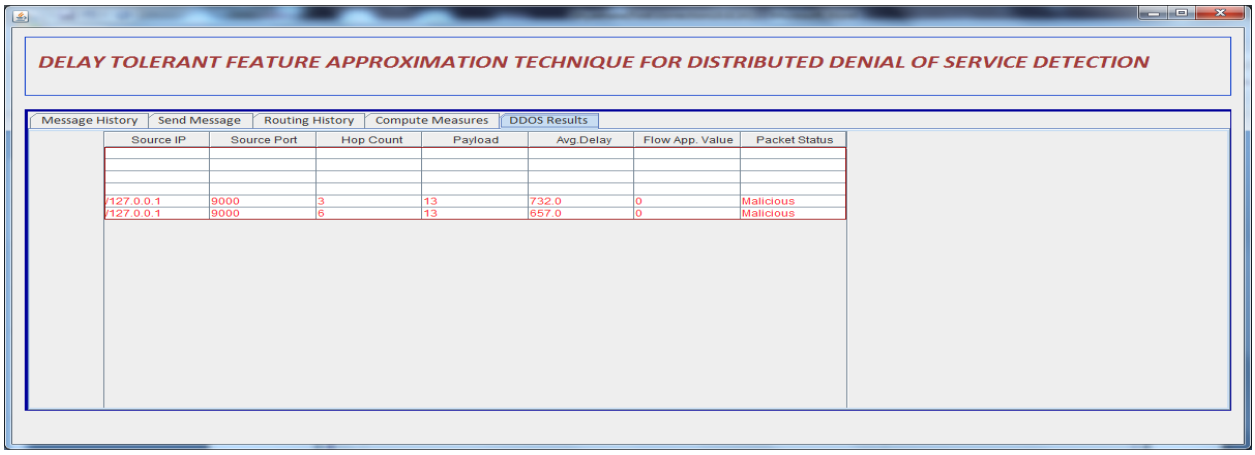

Figure-5. Result of intrusion detection.

The Figure-5 displays the result of intrusion detection, the average payload value, flow approximation results, and it shows the result of features being extracted using which the measures has been computed to decide the packet as legitimate or malicious. In Figure-6, Time complexity is an exact count of operations T (n) as a function of input size n. A measure of the amount of time required to execute an algorithm. It expresses relationship between the size of the input and the run time for the algorithm. Time complexity of computing different intrusion detection methods depends on the size of the traffic information received from the adjacent routers that is with in neighbor hops, and so it depends on the network topology and traffic volume on the network. The comparison of DDoS attack detection accuracy shows clearly that the proposed approach has produced efficient performance and results that are more accurate.

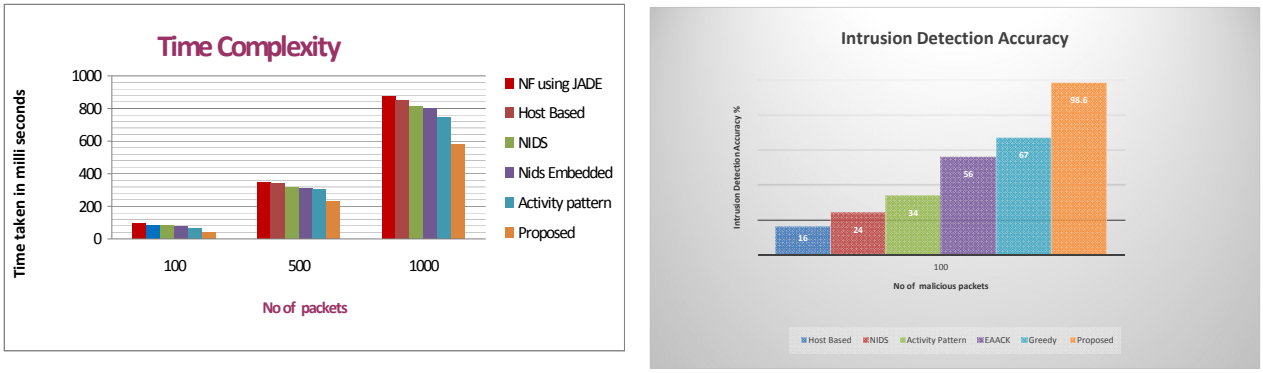

Figure-6. Time complexity and comparison of intrusion detection accuracy.

CONCLUSIONS

Propose a new network forensics approach to find the malicious packet and DDoS mitigation attacks. (Check this something is missing) The proposed approach combines three different approaches to perform denial of service attack detection. The delay tolerant approach utilizes the botnet based detection and flow approximation techniques to perform detection of malicious packets. Please check-these lines-something is missing)

The botnet based approach identifies not only the origin of threat and identifies a set of compromised nodes which supports the DDoS attacks and the flow based approximation techniques identifies the threat based on the illegal flow of packets and request from various sources of network. Finally, the delay tolerant approach uses both the techniques to perform complete detection of DDoS attacks and produce efficient results. The combination of these approaches helps in identifying the malicious nodes and requests in efficient manner, which increases the throughput of the overall network.

REFERENCES

[1] Vokorokos L. 2010. Host Based Intrusion Detection System. Intelligent Engineering Systems (INES). pp. 43-47.

[2] Maciá-Pérez F. 2012. Network Intrusion Detection System Embedded on a Smart Sensor, Industrial Electronics. IEEE Transactions on. 58(3): 722-732.

[3] Haldar N. A. H. 2012. An Activity Pattern Based Wireless Intrusion Detection System Information Technology: pp. 846-847.

[4] Elhadi M. Shakshuki. 2013. EAACK-A Secure Intrusion-Detection System for MANETs. IEEE transactions on industrial electronics. 60(3).

[5] B. B. Gupta, R. C. Joshi, Manoj Misra. 2012. ANN Based Scheme to Predict Number of Zombies involved in a DDoS Attack. International Journal of Network Security (IJNS). 14(1): 36-45.

[6] Tomasz Andrysiak, Łukasz Saganowski, Michał Choraś. 2013. DDoS Attacks Detection by Means of Greedy Algorithms, Image Processing and Communications Challenges 4, Advances in Intelligent Systems and Computing. 184: 303-310.

[7] Esraa Alomari, Selvakumar Manickam, B B Gupta, Shankar Karuppayah and Rafeef Alfaris. 2012. Article: Botnet-based Distributed Denial of Service (DDoS) Attacks on Web Servers: Classification and Art. International Journal of Computer Applications. 49(7): 24-32.

[8] Rabia Latif, Haider Abbas, Saïd Assar, Seemab Latif. 2014. Analyzing Feasibility for Deploying Very Fast Decision Tree for DDoS Attack Detection in Cloud-Assisted WBAN, Springer. Intelligent Computing theory. 8588: 507-519.

[9] Junho Choi, Chang Choi, Byeongkyu Ko, Pankoo Kim. 2014. A method of DDoS attack detection using HTTP packet pattern and rule engine in cloud computing environment, Springer. Soft computing. 18(9): 1697-1703.

[10] R. H. Akbani, S. Patel, and D. C. Jinwala, "DoS attacks in mobile ad hoc networks: A survey," in Proc. 2nd Int. Meeting ACCT, Rohtak, Haryana, India. pp. 535-541.

[11] N. Kang, E. Shakshuki and T. Sheltami. 2011. Detecting forged acknowledgements in MANETs. In Proc. IEEE 25th Int. Conf. AINA, Biopolis. Singapore, 22-25, pp. 488-94.

[12] B. B. Gupta, R. C. Joshi, Manoj Misra. 2012. ANN Based Scheme to Predict Number of Zombies involved in a DDoS Attack. International Journal of Network Security (IJNS). 14(1): 36-45.

[13] Katkamwar N. S., Puranik A. G., Deshpande P. 2012. Securing Cloud Servers against Flooding Based DDoS Attacks. International Journal of Application or Innovation in Engineering and Management. 1(3).

[14] Lonea A. M., Popescu D.E., Tianfield H. 2013. Detecting DDoS Attacks in Cloud Computing Environment. Int. J. Comput. Commun. 8(1): 70-78.

[15] R. Akbani, T. Korkmaz and G. V. S. Raju. 2012. Mobile Ad hoc Network Security. In Lecture Notes in Electrical Engineering, vol. 127. New York: Springer-Verlag, 2012, pp. 659–666.