IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 19, Issue 2, Ver. II (Mar.-Apr. 2017), PP 56-61
www.iosrjournals.org

Artificial Neural Network To Detect Know And Unknown DDOS Attack

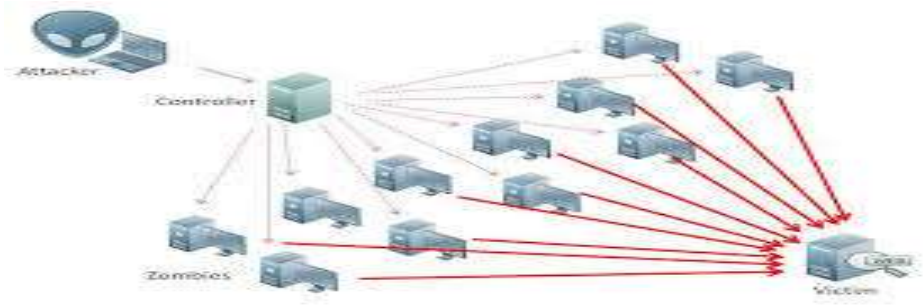
Dr. D. Karunkuzhali, G. Aishwarya, S. Eliza Paulin, A. Meena
(Information technology, Panimalar Engineering College, Chennai)

Abstract: *The key objective of Disturbed Denial of Service (DdoS) attack is to compile multiple systems across the internet with the infected zombies/agents and form the botnets of network. The infected systems are remotely controlled by an attacker that is programmed to launch packet flood. Within this context the purpose of this paper is to detect known and unknown DdoS attack. It can also detect DdoS attack from genuine traffic and also the encrypted packets are detected.*

Keywords: *AES-256 bit algorithm, IP traceback algorithm and Local Flow Monitoring algorithm.*

I. Introduction

The source of DDOS attack in the internet is usually hard. Normally attackers generate huge amount of request to victims through comprised zombies and entropy variations are identified from each packet. Recent survey shows that more than internet operators demonstrated that DdoS attack is strong. Here we have used the IP traceback means the capability of identifying the actual source of any packets across the internet. It is best methods of easily identifying the packet from which clients. Normally in networking means there performs the main role of routers. If once the attack has been launched IP Trace back has been launched IP Trace back has been used, and also the local flow monitoring has been for abnormal flow of packets. In DdoS attack the packet number distribution of packet loss, which are out of the control of attackers. For that entropy variations have been introduced. Here the packet that is through a router to flows was defined by the upstream router, where the packet comes from, and the destination address of the packet. The TCP, UDP, ICMP are the most used protocols to launch DDOS attack.



Here there are some steps to launch DDOS attack

- Attackers first establish a network which is responsible for huge volume of traffics to deny the series of normal users.
- Attackers then discover vulnerable host of the networks. Vulnerable host in the sense that the system running no anti viruses or out of date anti viruses software.
- It can be shown by the entropy growth from the point of attack.

Then encrypted packet has to be detected correctly with the help any one of the techniques.

II. Artificial Neural Network As Existing System

Artificial Neural Network (ANN) algorithm is used to detect DdoS attacks based on specific characteristic features (patterns) that separate DdoS attack traffic from genuine traffic. The patterns used for training purposes are instances of packet headers, which include source addresses, ID and sequence numbers coupled with source destination port numbers. The disadvantage of DdoS attack is that the ANN of the DdoS detector that was previously trained with old datasets fails to detect some known DdoS attacks. It is very difficult to detect DdoS attacks when the protocol headers are encrypted.

III. Related Work

Various methodologies and technologies for reducing the effects of DDOS attack in different network environments have been proposed and evaluated. Detect DdoS attack in mobile ad-hoc network and avoid change in threshold value for next flow many times by wasting resources. Compare the current probability distribution, cumulative distribution and best probability distribution to change the threshold value for next flow. Effective and efficient IP Traceback schema against DdoS attacks based on entropy variations. Dynamic value to determine the entropy rate which is based upon the packet size of the client. The decision rules are provided for both categorical continuous features. By combining the weak classifiers for continuous features and the weak classifiers features into a strong classifier, the relations between these two different types of features are handled naturally, without any forced conversions and categorical features. Adaptable initial weights and a simple strategy for avoiding over fitting are adopted to improve the performance of the algorithm. The performance of the bPDM is evaluated in three ways: first, synthetically- generated traffic provides for a controlled comparison of detection time as a function of the anomalous level of traffic. Second the approach is shown to be able to detect controlled artificial attacks over the USC campus network in varying real traffic mixes. Third the proposed algorithm achieves rapid detection of real denial of service attacks as determined by the replay of previously captured network traces. Compare 128 bits, 192 bit and 256bit AES techniques based on encryption and decryption time and throughput. The AES algorithm is symmetric block cipher that processes data blocks of 128 bits using a cipher key of length 128, 192, or 256 bits. Iterative looping approach with block and key size of 128 bits, lookup table implementation of S-box. The design has been coded by Very high speed integrated circuit Hardware Descriptive Language.

IV. Proposed System

The user needs to login to access the service. If the user fails to login in 3 attempts, that particular IP will be blocked for 12 hours. The server will wait for the incoming client calls, and periodically report its status to the The Local Flow monitoring algorithm (how many clients have been connected with the server and how many clients have been disconnected with the server). Once an incoming packet is detected and accepted, the server will create a separate thread to handle this client, it will therefore create as many separate sessions as there are incoming clients and it should be able to transfer files with any one of these clients. Once the server receives a Quit/quit message from one of these clients, it will shut down the connection with this particular client. If the server gets several requests for the same service, the IP tracking algorithm is used to track that particular IP address. The malicious MAC IP will be blocked. Encrypted headers can be detected using AES algorithm.

V. Modules

1. Network Assumptions: In this module, Multipath routing is considered an effective mechanism for fault and intrusion tolerance to improve data delivery in WSNs. The basic idea is that the probability of atleast one path reaching the sink node or base station increases as we have more paths doing data delivery. While most prior research focused on using multipath routing to improve reliability, some attention has been paid to using multipath routing to tolerate insider attacks. These studies, however, largely ignored the tradeoff between QoS gain vs. energy consumption which can adversely shorten the system lifetime.

2. User Interface: In this module, we have to create the user interface for establishing the connection between the sender and the receiver. Here the user has to prepare the data that has to send to the particular destination. For every transaction, user interface is the main part for establishing connection between the sender and the receiver. After establishing the connection, the sender has to prepare for the data, which he wants to send to the particular destination.

3. Path Selection: The splitted packets are sent to the path selector component, according to the path information, the path selector component will choose the path and send the packet through the network.

4. Authentication: MAC address is typically used as a unique identifier for all the nodes on the network. we have found that the distance between the centroid in signal space is a good test statistic for effective attack detection. All the Client nodes always login with our Specific IP and MAC address. Attackers can't easily forge their MAC address so they can avoid IP spoofing attacks

5. Spoofing Attacks: Due to the open-nature of the wireless medium, it is easy for adversaries to monitor communications to find the layer-2 Media Access Control (MAC) addresses of the other entities. Recall that the MAC address is typically used as a unique identifier for all the nodes 2 on the network. Further, for most commodity wireless devices, attackers can easily forge their MAC address in order to masquerade as another transmitter. As a result, these attackers appear to the network as if they are a different device. Such spoofing attacks can have a serious impact on the network performance as well as facilitate many forms of security weaknesses, such as attacks on access control mechanisms in access points , and denial-of-service through a deauthentication attack A broad survey of possible spoofing attacks can be found in . To address potential

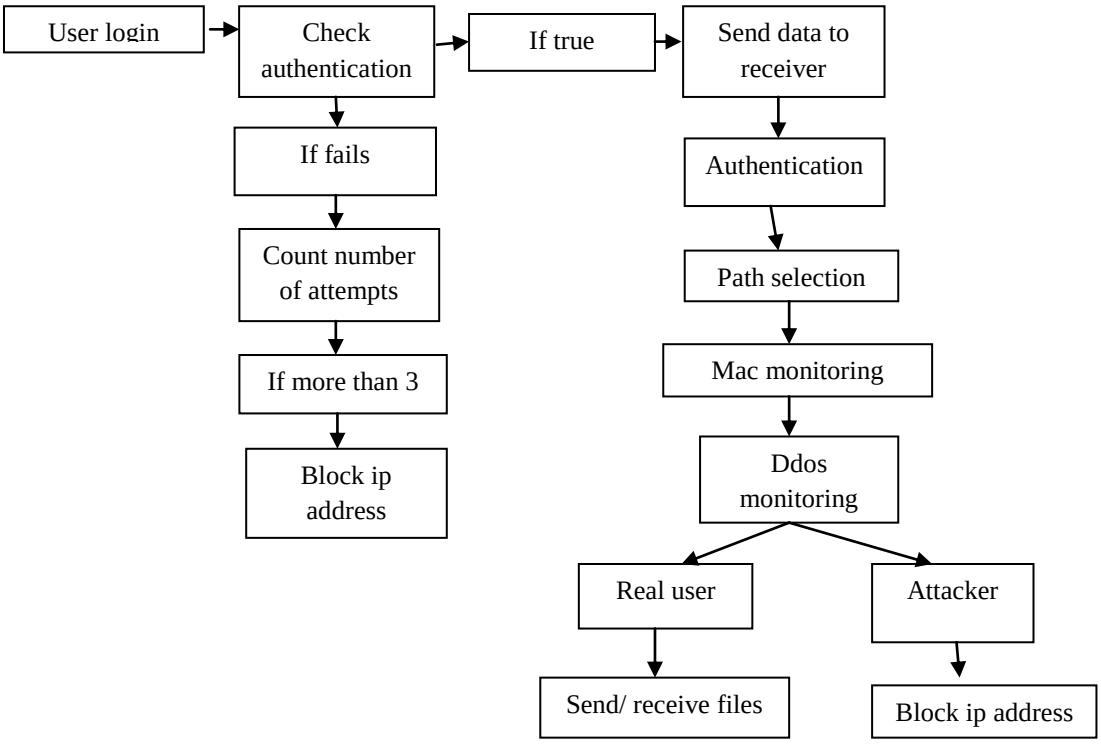
spoofing attacks, the conventional approach uses authentication. However, the application of authentication requires reliable key distribution, management, and maintenance mechanisms. It is not always desirable to apply authentication because of its infrastructural, computational, and management overhead. Further, cryptographic methods are susceptible to node compromise– a serious concern as most wireless nodes are easily accessible, allowing their memory to be easily scanned.

6. Flood Attack: In this project, A DoS attack involves sending large number of packets to a destination to prevent legitimate users from accessing information or services. Zombies are gathered to send useless service requests Continuously , packets at the same time. DdoS attacks are targeted at stealing, modifying or destroying information.

7. Server Monitoring: This module continuously monitoring the all request from the Client. When the request is coming, it identifies the IP address with MAC address and stored in cache and starts counting the request from the same IP address and also maintains the timer. More than 20 requests within one second from same IP address are considered as DDOS attack. Then the IP address is blocked for certain time periods (e.g. 12Hr).

- **Detection:** More than 20 requests within one second from same IP address is considered as DDOS attack.
- **Prevention:** The suspicious IP address is blocked for certain time periods (e.g. 12Hr)

VI. Architecture Diagram



VII. Proposed System Algorithm

7.1 Local flow monitoring

Initialization

1. Local threshold parameter C
2. Another local threshold parameter E.
3. Next local threshold parameter for sampling time period ΔT

Step1. Label first flow as f1 and set packet sixe as 100 bytes.

Step2. Execute first flow for first time ie, x1=1 (we have to execute first flow f1 (every flow) for 5 times.)

Step3. Wait till ΔT is over or not.

Step4. Once timer is equal to, then calculate probability distribution as follows.

$$P_i=x_i|\sum_{i=1}^{n} x|^{-1}$$

Step5. After getting probability distribution, calculate entropy distribution as follows...

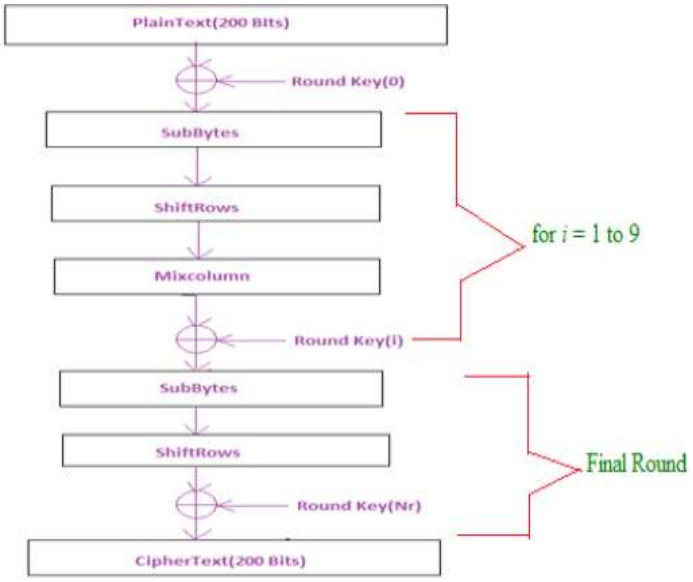
$H(F)=-\sum p_i \log p_i$
Where i is 1, 2, 3, 4, 5 times execution of each flow.
Step6. Then save H (F) for 5 times execution of each flow. So H (F) is for each flow.
Step7. Check weather absolute value of | H(F)-C| of current flow is less than or equal to local threshold parameter.
7.1 Calculate mean as follows
$C[t]=\sum a_i C[t-1]$, $\sum a_i=1$
7.2 Calculate standard deviation as follows.
$E[t]=\sum \beta_i \Delta[t-1]$, $\sum \beta_i=1$
Step8. Calculate direct probability distribution between current and previous flow.
$E_1[t]=E_1[t]-E_1[t-1]$
Step9. Calculate cumulative probability distribution of all previous flows.
$E_2[t]=\sum_{f=1}^{n} E[f]$

Step10. Calculate recommended probability distribution by comparing all flows.
$RecE_2[t]=max(Ef)$
Where f=1, 2 ,3, 4 ,---n
Step11. Compare E_1 and E_2 and $RecE_2$. Choose best sigma as final probability distribution.
Where t=1, 2,3,4,5 to number flows
Step12. Go for next flow.

7.2 IP Tracking Algorithm:

1. Initialize a set A= null and obtain the local parameter of C and D
2. Let U= {ui}, I belongs to I be a set of the upstream routers, D={di }, I belongs to I be a set of the destinations of the packets and V be the victim.
3. Define attack flows, fi =<u,v>, i= 1,2,,3….n, ui belongs to U and sort the attack flows in the descent order and we have f1, f2, …..fn.
4. For i=1 to n
{
Calculate H (F\fi1)
if(|H(F)-C|>D) then append the responding upstream router of fi to set A
else break;
end if;
end for;
5. Submit traceback request to the routers in set A respectively and deliver the confirmed zombies information set A to the victim.

7.3 Encryption Algorithm of AES



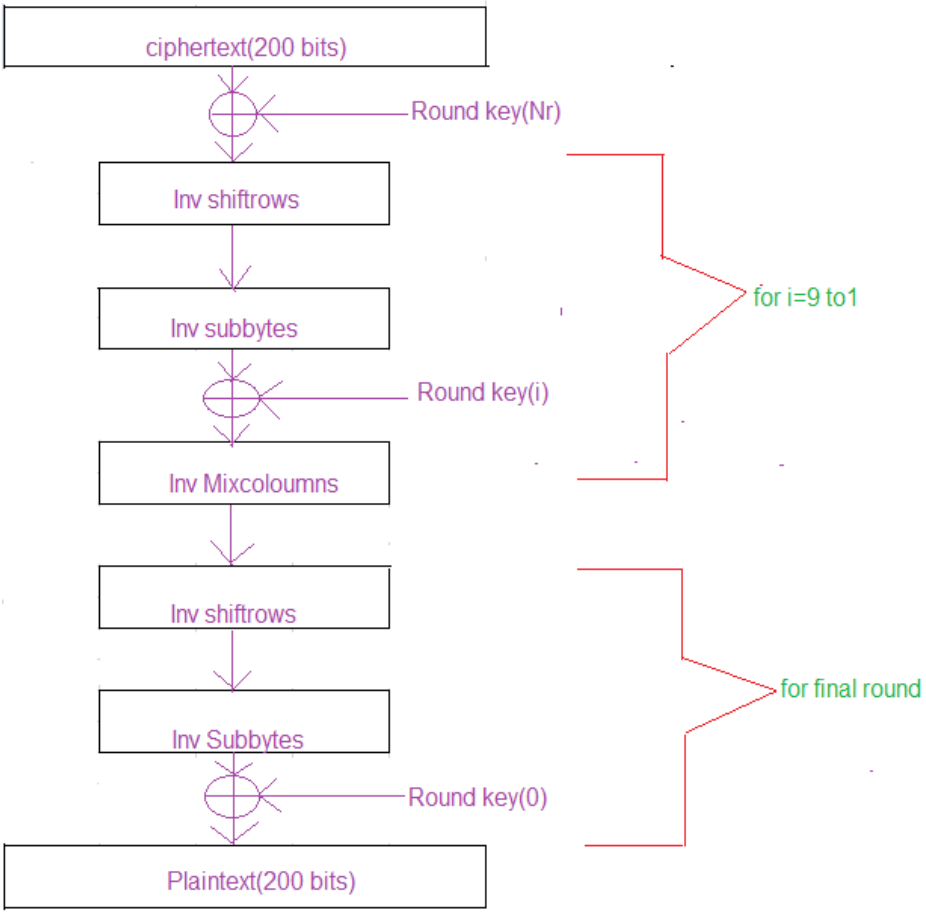
Step1: SubByte Transformation: In this transformation, each of the byte in the state matrix is replaced with another byte as per the S-box (Substitution Box). The S-box is generated by firstly calculating the respective reciprocal of that byte in GF (2^8) and then affine transform is applied

Step2: ShiftRows Transformation: In this transformation, the bytes in the first row of the State do not change. The second, third, fourth and fifth rows shift cyclically to the left by one byte, two bytes, three bytes and four bytes respectively

Step3: MixColumns Transformation: It is the operation that mixes the bytes in each column by the multiplication of the state with a fixed polynomial matrix . It completely changes the scenario of the cipher even if the all bytes look very similar. The Inverse Polynomial Matrix does exist in order to reverse the mix column transformation.

Step4: AddRoundKeyTransformation: InAddRoundKey transformation, a roundkey is added to the State by bitwise Exclusive-OR (XOR) operation.

Decryption Algorithm for AES



Decryption is the process of extracting the plaintext from cipher text. The Decryption structure of proposed algorithm as shown in figure.5 is obtained by inverting the encryption structure which is shown above. Corresponding to the transformations in the encryption, InvSubBytes, InvShiftRows, InvMixColumns, and AddRoundKey are the transformations used in the decryption as shown in Fig. below. The roundkeys are the same as those in encryption generated by Key Expansion, but are used in reverse order

Step1: InvSubBytes Transformation: InvSubBytes is the inverse transformation of SubBytes, in which the inverse *S*-box is applied to individual bytes in the State. The inverse *S*-box is constructed by first applying the inverse of the affine transformation in , then computing the multiplicative inverse

Step2: InvShiftRows Transformation: InvShiftRows is the inverse transformation of ShiftRows. In this transformation, the bytes in the first row of the State do not change; the second, third, and fourth and fifth rows are shifted cyclically by one byte, two bytes, three bytes and four bytes to the right respectively

Step3: InvMixColumns Transformation: InvMixColumns is the inverse transformation of MixColumns. This is a complex procedure as it involves severely the byte multiplication.

VIII. Conclusion

We have used local flow monitoring algorithm to monitor the server and IP Traceback Algorithm to detect DdoS Attack that separates genuine traffic from DdoS attacks. Different DdoS attacks were then launched while normal traffic was going through the network. We can even detect unknown DdoS attack. Detecting DdoS attack when the protocol headers are encrypted with any encryption algorithm can be detected using AES 256 bits algorithm.

References

[1]. Mr. Sandeep Shinde, Dr. J. W. Bakal "Traceback Mechanism for DDoS Attacks Using Local Flow Monitoring in MANET" Shinde, Sandeep; Bakal, J W. International Journal of Science, Engineering and Computer Technology; Hisar 5.8 (Aug 2015): 301-303.
[2]. Mr.T.Bharath Manohar, Mrs.E.V.N.Jyothi, Mrs.B.Rajani , Mr.I.Rajesh Kumar "A novel entropy based detection of DDoS attack." International Journal of Emerging Trends & Technology in Computer Science, Volume 1, Issue 2, July – August 2012
[3]. "Efficient Implementation of AES "(2013)
[4]. "Implementation of AES Algorithm Based on FPGA "(2014)
[5]. W Hu, W Hu, S Maybank "AdaBoost-Based Algorithm for Network Intrusion Detection", IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) (Volume: 38, Issue: 2, April 2008)
[6]. G Thatte, U Mitra, J Heidemann "Parametric Methods for Anomaly Detection in Aggregate Traffic", IEEE/ACM Transactions on Networking (Volume: 19, Issue: 2, April 2011)