International Association of Scientific Innovation and Research (IASIR)
(An Association Unifying the Sciences, Engineering, and Applied Research)

ISSN (Print): 2279-0020
ISSN (Online): 2279-0039

International Journal of Engineering, Business and Enterprise Applications (IJEBEA)

www.iasir.net

Dual Mechanism to Detect DDOS Attack

Priyanka Dembla , Chander Diwaker[2]
[1]Research Scholar, [2] Assistant Professor
CSE Department, U.I.E.T., Kurukshetra University, Kurukshetra University, Haryana, INDIA

Abstract: *Cloud computing has emerged as computing paradigm which shares resources and services with its customers. In its early days we does not heard of the Denial of service attack. But since 2000, a series of DDOS attacks by multiple nodes is accomplished of blocking the services of cloud servers. The attack can be for many reasons. It became a major threat for cloud environment. Significant problem of DDOS attacks is that they are difficult to detect. The effects of various attacks can shut the organization off from network. The main goal of this attack is to make cloud services unavailable for the legitimate users. This paper aims at proposing an efficient method for security in cloud. We propose an algorithm which modifies the confidence Based Filtering method (CBF) by adding the IP Spoofing filtering method before applying CBF.*

Keywords: *Cloud computing, DDOS attack, Hop count, IP spoofing*

I. Introduction

Cloud computing is one of the most hyped information technology and it has become one of the fastest growing segments of IT. Costumers must only pay for the amount they are using and have not to pay for local resources such as storage or infrastructure. The cloud offers several benefits like fast deployment, pay-for- use, lower costs, scalability and flexibility. Resources such as hardware and software are liable to be outdated soon [1]. Therefore outsourcing of resources is the solution.

Cloud computing is basically consist of 4 deployment models and 3 service models. Deployment models are- Public Cloud is cloud model in which services are available for the public and payment is on the basis of pay per use. It is less secure model among all the models. Private Cloud provides services to the particular group of people which may belong to some organisation. So it becomes easy to manage them. Hybrid is an environment in which some of the resources are for private use such as in private cloud and rest is for public use. It is a combination of public and private cloud. Community Cloud model is shared by the organisation or people which have similar cloud requirement. These number of organisation are limited in nature moreover they are trusted ones [2].

Services of the cloud is provided on the basis as Software as a service, Infrastructure as a service and Platform as a service. A cloud application delivers Software as its Service over the internet, thus clients does not have to install the application on its system. Platform as a Service provide a computing platform. It has all the application typically required by the client deployed on it. In Infrastructure as a Service, the client need not purchase the required servers, data center or the network resources. As a result customers can achieve a much faster service delivery with less cost.

DDos attacker is one of the most common attacks in cloud computing. Attacker sends a huge amount of packets to a certain service. Each of these requests has to be processed by the server. This increases workload per attack request. This usually causes denial of service to the legitimate users also the performance of network reduces. This attack is also known as flooding attack. Denial of service does not modify data instead it crashes server and networks, making service unavailable to the legal users. DOS can be launched from either a single source or multiple sources. Multiple sources DOS attacks are Distributed denial of service (DDOS) [3]. DDOS is distributed, large scale coordinated attempt of flooding the network with large amount of packets which becomes difficult for victim network to handle and hence the victim sever becomes unable to provide the services to its legitimate user. Various resources such as bandwidth, memory, computing power get wasted in serving flooding packets. It makes services or resources unavailable for indefinite amount of time. The attacker usually spoofs IP address section of a packet header in order to hide their identity from their victim[4].

II. Related Work

Kumar *et al.* [4] presented an approach in which packets with the same hop count passes through the router are assigned some identification number. This number is the combination of the 32 bits of IP address of the router and encrypted value of hop count. The receiver of the packets matches this hop count with the already stored value. This PID is placed in the identification field of the IP header. When the router receives the packet, it checks packet ID number whether it is valid. The advantage of this approach is that if it filters the traffic after receiving just one packet. If it is not valid, it means that the packet is arrived from the sender host or from

attacker which sends packet with forged mark. After receiving this type of packet, router starts detection process. Attack graph is constructed to filter all packets coming out of the attack source. By the attack path construction, it is easy to identify the source of the attacker. This algorithm lowers false alarm and is executed close to attack source. Overhead of routers are reduced as compared to the IP traceback and packet marking approach.

Yu *et al.* [5] proposed dynamic resource strategy for countering DDOS attack. It clones sufficient Intrusion Detection servers for the victim with the help of resources of other clouds. Before serviced by the server, packets have to pass through the queue and the IPS. The assumption of this paper that the users are stable and resources of clouds are sufficient to overcome DDOS attack. During the attack on any of the individual cloud server, number of attack packets generated by the botnets increases. This algorithm clones multiple IPS to maintain the quality of service. The number of IPS depends upon the volume of the attack. This method focuses on the resources management. The loop point of this approach is that if any of the data center runs out of resources during attack, this method will fails in this case.

Huang *et al.* [6] proposed low reflection ratio mitigation system against DDOS attack. System consists of source checking, counting, attack detection, turing test and question generation module. Turing test is conducted for the possible attackers detected at the detection module. This test can determine the incoming packet is initialized by Zombie host or Human. The packet first reaches checking and counting module. Attack detection module cooperates with source checking module to detect any DDOS attack. It tries to find malicious source and blocks it. Test based turing testing module randomly selects question from question generation module and waits for the requester to answer. Without getting correct answer to the question, it will not be allowed to reach server. The system has low reflection ratio with high efficiency.

Megha *et al.* [7] presented a mechanism to prevent DDos attacks and to improve resource availability of resources. The basic idea behind the proposed system is to isolate and protect the web server from huge volumes of DDos request when an attack occurs. In the proposed algorithm for user friendly in domain and the capacity to store user profiles and profiles and sending them to the server component aided by computer speed high memory capacity and accuracy. This have the advantage of differentiating the clients from the attackers those who tries to affect the server function by posting requests in a large amount for unwanted reasons. This can be used for creating defenses for attacks require monitoring dynamic network activities.

III. Proposed Approach

The main goal of this paper is to filter the packets received from various source on the basis of the IP spoofing by using TTL field in the packet and then allowing these filtered packets to go through CBF method. This method is based upon the correlation pattern stored in the packets. These patterns are mainly in network and transport layer.

DDOS attack is accompanied by IP spoofing. Attackers conceal their identity by changing the Source IP address field of the packet to make it as packet is coming from the legitimate user. But attacker can forge the Hop Count of the packet. This idea is used in this paper to filter the packets. Hop count and SYN flag of the packets detects whether the packet is spoofed one or legitimate. The spoofed packet is rejected and rest the packets which passed this test are collected under filtered list for further test. This filtering has reduced the numbers of packets on which further tests will be applied. Hence it reduced the overhead of applying CBF on all the packets. CBF consist of two concepts- Confidence and Score. Each packet from the filtered list is collected and the frequency of appearance of single attribute is calculated. This is the confidence of that attribute value. If the confidence of single attribute is greater than the minconf (pre defined) are selected to generate attribute value pairs. This step is essential because if the confidence of one attribute value in an attribute value pair is not greater than minconf, the confidence of the combination of this value pair will still not be greater than minconf. We again scan all the packets in the filtered list to count the frequency of appearances of attribute value pairs and count their confidence. Attribute values pairs whose confidence is greater than minconf will update the nominal profile. Nominal profile is a 3 dimensional array. The first dimension is for first attribute pair and the second dimension is for second attribute pair. The third dimension is the confidence value dimension. There is no need to update nominal profile if the confidence of attribute pairs less than predefined confidence value. Score is the weighted average of the confidence of the attribute value pairs in it.

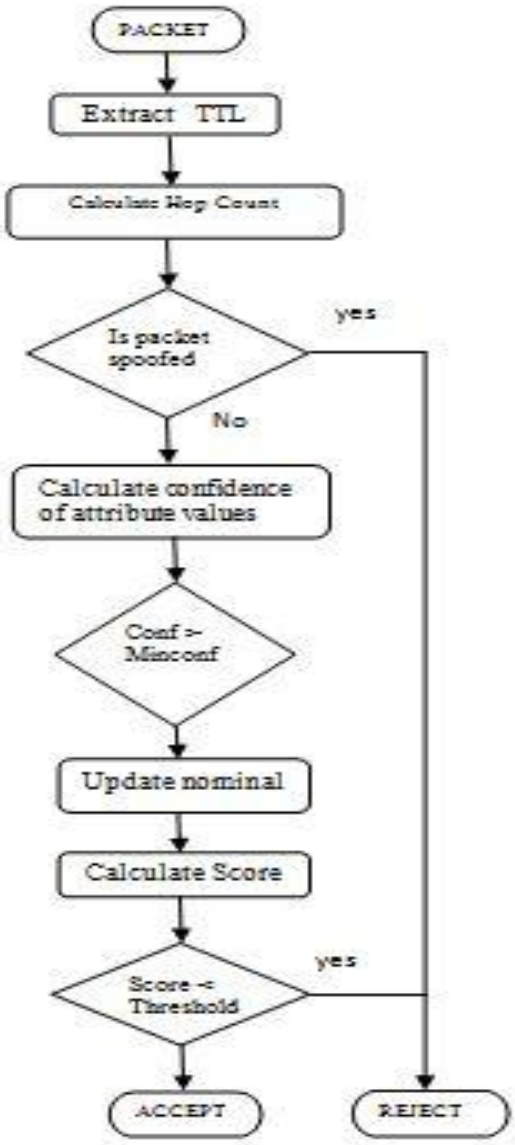
$$\text{Score} = \sum (\text{weight} * \text{confidence [attribute value pairs]}) / \sum \text{weight}$$

Weights of the attributes are adjusted on the basis of operating system, network structure and other elements. The patterns which are less copied by attackers are generally are given higher weight. This requires looking in the nominal profile for the confidence of the attribute pairs and applying some arithmetic operations. Attributes pairs whose confidence is not on the nominal profile, we will use minconf value instead when confidence values are used in calculating score. Score of the packets is generated by the above method. After calculating CBF

scores of the packets, we use it to distinguish attack packets from the legitimate ones. Method will only accept the packets with scores greater than discarding threshold. Discarding Threshold can be fixed depending upon the score distribution or dynamic like load shedding algorithm. In our paper we have used fixed discarding threshold.

IV. System Model

Figure 1 system model



V. Algorithm

nspoof==0;

count[attribute value]==0;

For each packet

Calculate hop count and SYN flag ;

hop count=Final TTL-Initial TTL;

if(packet is in table)

if(SYN==1)

compare hop count with stored hop count;

```
                if (same value)
                add it in table;
                else update hop count;
        else
                compare hop count with stored hop count;
                if (same value)
                allow packet;
                else
                remove packet;
                nspoof+1;
else                            // packet is not in table
        if (SYN==1)
                add packet in table;
        else
                remove packet;
                nspoof+1;
for each packet received after filtering spoofed
for each attribute value in packet
        count[attribute value] +1;
        calculate confidence[attribute value] = count[attribute value]/ no of packets;
        if (confidence[attribute value] > MinConf)
                calculate count[attribute value pairs];
                if (confidence[attribute value pairs]> MinConf)
                        update nominal profile;
                else do not update nominal profile;
for each value in nominal profile
        calculate score;
        score= ∑ (weight * confidence[attribute value pairs] )/ ∑ weight;
if (score< Threshold)
        reject packet ;
        else accept;
```

VI. Simulation conditions

The test environment is intel core i3 processor. The simulation programs is written in NetBeans. The window size is set to 10 packets, and the value of minconf is set to 0.13.Under this circumstance, the storage data at counting period are affordable in normal servers. Our method spends around 0.024 seconds to process data during each non attack phase. The weights in score calculation are set higher in the attribute pairs containing source IP address, TCP server port number or TTL value, and set lower in those only with TCP flag, IP protocol type and packet size. For the fast response at attack period, fixed discarding threshold is adopted. In implementation, discarding threshold is selected as 0.012. The six single attributes used are - total time, time to live, protocol type, source IP address, flag, Destination port number like those in CBF. The filtering is on the basis of the spoofed table which contains IP address and Hop count of all the packets entered in the system. SYN flag and TTL of each packet is extracted from the packet. Hop count is calculated from the TTL. Figure 2 is showing the spoof filtering result. Here, 10 packets are scanned in which 2 of them are found to be spoofed

and hence filtered out. The remaining 8 packets are collected in the filtered list. We now apply second phase of testing which includes calculating confidence of each attribute pair and updating their value in nominal profiles. As we have 6 attributes on which we are working. So we will have total of 15 nominal profiles. The figure 3 is showing the result of proposed work. It is attack phase in which score is calculated and on the basis of discarding threshold , packets are dropped and accepted. This figure is also showing IP addresses of the packet which are discarded.

Figure 2 Spoof filtering

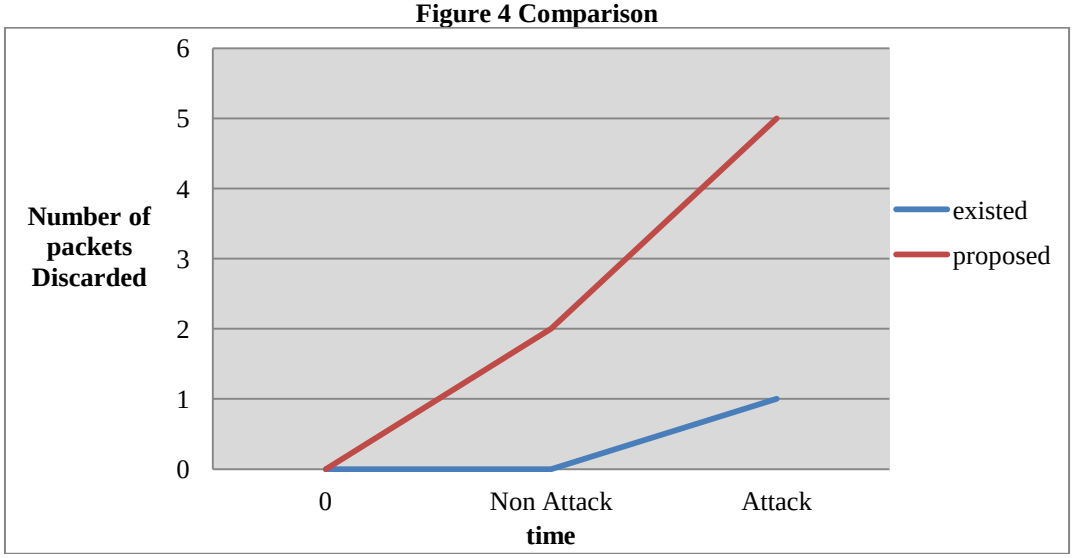
```
NON-ATTACK::Slot packets: 10
packets :[27 236 1 50856 75 8545]
ip: 198.168.176.204 syn: 1 hc :19
SIZE :1
packets :[97 119 2 50088 105 8159]
ip: 195.168.42.150 syn: 0 hc :9
SPOOFED 2
packets :[93 98 6 49833 26 8502]
ip: 194.169.79.64 syn: 1 hc :30
SIZE :2
packets :[79 118 1 50345 190 8094]
ip: 196.169.191.46 syn: 1 hc :10
SIZE :3
packets :[66 77 6 49833 219 8988]
ip: 194.169.58.191 syn: 1 hc :51
SIZE :4
packets :[38 175 6 50345 171 8685]
ip: 196.169.248.165 syn: 1 hc :80
SIZE :5
packets :[27 236 1 50856 107 8545]
ip: 198.168.176.204 syn: 1 hc :19
syn=1
added
packets :[97 119 2 50088 105 8159]
ip: 195.168.42.150 syn: 0 hc :9
SPOOFED 2
packets :[93 98 6 49833 26 8502]
ip: 194.169.79.64 syn: 1 hc :30
syn=1
added
packets :[79 118 1 50345 190 8094]
ip: 196.169.191.46 syn: 1 hc :10
syn=1
added
mcount: 2
NON-ATTACK::Filtered Size: 8
```

Figure 3 Discarded IP addresses

```
test  Packets : 8
selected packet: 1 0.028666666706299616
ATTACK::Conf Packets: 1
Attack end....
End time: 1404777438899
Discarded:
194.169.58.191
195.168.42.150
195.168.176.165
197.168.181.238
Proposed: true
Total Time taken(millisec):: 20113
Total Time taken(sec):: 20.113
Total Time Non-Attack Profile Add(millisec):: 24
Total Time Non-Attack Profile Add (sec):: 0.024
Total Time Attack Profile Test(millisec):: 0
Total Time Attack Profile Test(sec):: 0.0
Non-Attack              Attack
Packets Spoofed         Packets Spoofed Conf
-------------------------------------------------------------
10      2               10      2       1
-------------------------------------------------------------
```

Figure 4 shows the comparative analysis of packet discarded over time. Here x axis represents the time (Attack and Non-Attack phase) and y axis represents the number of packets discarded. As it can be seen after implementation the proposed approach, the packet discarded over time increased.

Figure 4 Comparison



VII. Conclusion and future scope

Cloud computing is one of the most hyped information technology and it has become one of the fastest growing segments of IT. The most serious threat to cloud computing is DDOS attack. It caused a lot of damage to many organizations. Attacker shut down the servers for a period of time. The site became non functional for some time. Dual mechanism approach is used to prevent attack. This method is about to improve the CBF method which is based on the correlation patterns. Our analysis has provided a tool to prevent from attack by using IP Spoofing and correlation pattern among attributes of packet .DDOS attack is mainly associated with spoofed packets. The spoofed packets are dropped in the initial phase so reducing the overhead in calculating confidence and score of the all packets. The simulation result showed that 90 % of the DDOS attack can be dropped.
The proposed system can be enhanced in future by other researchers in the following ways:

- More flexible strategy for choosing weights for each attribute pair
- Discarding threshold can be chosen dynamically based on the load balancing or other factors
- Question generated module can be added which ask some questions by possible attackers before discarding packet. These questions can be easily solved by human but not program run by zombies.

References

[1] Farzad Sabahi, "Cloud Computing Security Threats and Responses", IEEE 3rd International Conference Comunication Software and Networks,2011, pp.245-249.

[2] Ziyuan, Wang, "Security and privacy issues within Cloud Computing", IEEE International Conference on Computational and Information Sciences, 2011,pp.175-178.

[3] Bansidhar Joshi, A. Santhana Vijayan and Bineet Kumar Joshi, "Securing Cloud Computing Environment Against DDos Attacks" IEEE International Conference on Computer communication & Informatics , 2012, pp. 1-5.

[4] Bharathi Krishna Kumar , P. Krishna Kumar, and R. Sukanesh, "Hop count based packet processing approach to counter DDoS attacks", IEEE International Conference on Recent Trends in Information, Telecommunication and Computing, 2010, pp. 271-273.

[5] Yu, Shui, Yonghong Tian, Song Guo, and D. Wu, "Can we beat ddos attacks in clouds?", IEEE International Conference on Transactions on Parallel and Distributed Systems, 2013, pp.1-11.

[6] Vincent Shi-Ming Huang., Robert Huang, and Ming Chiang,"A DDoS Mitigation System with Multi-stage Detection and Text-Based Turing Testing in Cloud Computing", 27th IEEE International Conference on Advanced Information Networking and Applications Workshops,2013, pp. 655-662.

[7] Patel Megha, Arvind Meniya, "Prevent DDOS Attack Using Intrusion Detection System in cloud", International Journal of Computer Application ,Vol. 2 Issue 3 , 2013 , pp. 95-104.

[8] Haining Wang , Cheng Jin, and Kang G. Shin," Defense Against Spoofed IP Traffic Using Hop-Count Filtering",IEEE/ACM Transactions on Networking, 2007, pp.40-53.

[9] Weili Huang and Jian Yang, "New Network Security Based On Cloud Computing" ,IEEE Second International Workshop on Education Technology and Computer Science, 2010, pp.604-609

[10] Xue Jing and Zhang Jian-jun, "A Brief Survey on the Security Model of Cloud Computing" IEEE Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science, 2010 ,pp.475-478.