CHAPTER 6

IMPLEMENTATION OF SERVICE ORIENTED MULTI-MODEL NETWORK INFERENCE FOR IDENTIFYING DDoS ATTACK IN NETWORK IMMUNE SYSTEM

6.1 PREVIEW

Identifying DDOS attacks has become a more important task in network management and security where the immune system has the responsibility of securing the services and data available in the network. We propose a service oriented multi model network inference model for identifying DDoS attacks, which consist of several phases. The Service Oriented network inference model provides the solution for a variety of services. Unlike other methodologies available, the proposed model uses various metrics at various levels and combines the earlier models at the required stages of the identification process. The proposed model maintains the log trace where the features of packets are kept stored, which are already received. The proposed approach extracts various features like ttl, payload, hop count, addresses, time, service details. Using those logs, the proposed approach splits them into time domain values for each of the distinct services available. We maintain the list of services available in the network and for each of them a set of access history being maintained. The service access history will be split based on time windows, from which the flow approximation is computed using previous access history. Based on the results of time orient and flow approximation, the proposed model infers the
packet signature. The proposed model produces efficient results and reduces the time complexity.

6.2 INTRODUCTION

Modern communication and information systems have become more sophisticated due to increased solution. The organizations maintain their valuable resources at different locations of their network and in many computers. Furthermore, there are service providers which provide services through which the user can access the required data to complete his task. Even though the service provider has security measures to access the service, but there are some genuine users initiates some malformed request to the service with the intention to degrade the service quality. Not only the service, but also the channel gets disturbed due to some malicious users.

Generally, there are a variety of network attacks, like flow based or connection based, where in the case of flow based attacks the malicious user tends to attack the channel capacity so that he floods enormous amount of packets to the channel. Similarly, in case of connection based attacks, the malicious user tends to attack the service throughput or network throughput. In the second case, a malicious user might hold many connections for a particular time without passing any information on that, and the connection will be present simply idle. In both the cases, the network and its services are getting affected hugely, which has to be avoided to improve the network performance. There are solutions for both, but focuses on any one of them, also few solutions are there to handle both but missed various features of the request.

The service oriented multi model focus on the security measures enforced on the service, and it also secure the service from a variety of attacks raised through the network. Generally, the DDoS attacks are focused at a
service point and has the intention to degrade the service quality. Our ultimate aim is to identify such attacks in a wide spectrum to maintain network performance.

There may be N number of services available in the network and for each service, the model maintains a set of security measures, which makes the service secure one. Generally, the request generated from a source node passes through the number of intermediate nodes, which may be vulnerable to different attacks. Whenever a malicious node identifies or learns something about the service parameter or a prototype of a service, then the malicious node is more capable of producing enormous amounts of the malicious requests with fake values or false prototypes. This spoils the service provided and the servicing node has to spend more time in verifying the service parameters which in turn reduces the service quality.

In any service orient architecture, the throughput of the system highly depends on the service quality and the performance of the system. In SOA, the throughput is gauged based upon the completion of service and their rate of completion. There may be N number of service request approaching the service point, but the throughput of the service is about how many of them get completed successfully.

In case of Denial of service attacks, the malicious node can learn from the packets captured about the service prototype, and they can produce an enormous amount of the service requests with fake values of parameters and make the service point to spend most time in verification of parameters. This makes the most service request gets discarded by the service point due to the wrong values of parameters. This kind of behavior or the service inputs makes the service quality poor, and the rate of service completion gets reduced. This ultimately reduces the throughput of the service, and the genuine user will never get a chance to access the service.
The affected service will lose the user opinion, and then after nobody looks for the service due to the service unavailability. Any service orient architecture has to provide and avail service to the requested user at the maximum constraint, otherwise user orientation could not be achieved. For example, if there are set of user who tries to access the service, but they did not get the service access due to the congestion generated by the malicious node for some time. Then the user will be searching for some other service provided by other service providers. This has to be considered while looking at service optimization and has to take care of all these issues.

How we can stop these issues of denial of service attack is, the users and their access have to be monitored in all the time. The security modules have to keep track of the information about their access about what service they access? And when they access? , how they finish the service? and so on. If we could track this information, then we can find out the method of access a genuine user performs on each service, and we can generate a pattern about them.

The same user may access the same service in a different manner which we call heterogeneity because they differ in the time. So all the user has a specific pattern of access at different time windows. For example, if there exist a service named check mail, then the person working for an organization may access the mail service during morning at arrival of office and spend some time in that. So the duration of accessing the mail service will be high. However, when he leaves the office, he may think to check for new updates in a hurry mood, so at that time the way of accessing the mail will be different. At the time of leaving, he may spend only little time and the access time of login and logout will be different.

There are various methods has been discussed and available, but most of them suffer with the efficiency of DDoS detection. Some models use
only the hop details to identify the malicious user; some may use packet information to decide the packet status. All of them have a problem with identifying the genuine packet from other packets. The inference model is one, which infers some conclusion using the available history of records, packet signature, service details.

6.3 METHODS EXPLORED

There exist various approaches for distributed denial of service detection and we discuss a few of them here with the problem identified.

An activity pattern for host based intrusion detection system (Haldar 2012), proposed a novel approach for intrusion detection. The method overrides the usage of pattern recognition technique by using usage patterns of different users. The method uses the activity patterns of the users which is produced from the access logs of the services. The method uses the principle component analysis to identify the activity patterns of the users. The result of PCA is applied to generate the threshold based on which the alarm is produced on finding malicious access.

Yamini (2014) has proposed a Detecting DDOS Attacks by Circular Protection Network, which is an intrusion detection system which is installed in the service providers. The method exchanges various specific traffic information between them and enables collaborative intrusion detection.

Genetic algorithm and artificial immune systems (Sridevi 2012), proposes a new kind of defense technology of the network security and used as a countermeasure to preserve data integrity and system availability during an intrusion. An ideal IDS system should be capable of evolving itself to identify not only known attacks but also unknown attacks. Algorithms based
on Genetic Engineering and Immune Systems are known to evolve and learn from small examples. They proposed to investigate the efficacy of genetic search methods for feature selection and Immune system to classify threats and non threats.

Network intrusion detection by an artificial immune system (Junyuan Shen 2011), proposed an intrusion detection method which performs intrusion detection based on the activities of the user. The method employs a genetic algorithm and computes the minkowski distance to perform intrusion detection. The proposed approach increases the detection rate and produces efficient results.

Detecting and Analyzing DDoS Attack Using Map Reduce In Hadoop by Navale (2014), proposed a new approach to perform intrusion detection where the method has to use large size data. The method uses different methods like access pattern and counter based methods. The method has produced efficient results and increases the efficiency of the overall network intrusion detection.

Intrusion detection using flow correlation and feedback has been discussed in Poornima (2014), where initially the method identifies the normal flow of traffic and its pattern. Then it identifies the abnormal traffic pattern using a flash crowd detection algorithm. The method uses the flow correlation coefficient algorithm and sequential pattern to detect the intrusion detection.

Live Baiting for Service-Level DOS Attackers by Khattab et al (2008), is discussed with the aim of identifying defective members of the group. This leverage allows live baiting to detect attackers using low state overhead without requiring models of legitimate requests nor anomalous behavior. The amount of state needed by live baiting is in the order of the
number of attackers not the number of clients. This saving allows live baiting to scale to large services with millions of clients.

On Detection of malicious Users Using Group Testing Techniques, Thai et al (2008), discusses a theoretical model to provide security over service orient architecture. The proposed size constraint group testing (SCGT) works based on the size of the network and they discusses various approaches for different network scenarios.

A method for DDoS attack detection using HTTP packet pattern and rule engine in a cloud computing environment Junho Choi (2014), proposes a method of integration between HTTP GET flooding among Distributed Denial-of-Service attacks and Map Reduce processing for fast attack detection in a cloud computing environment. In addition, experiments on the processing time were 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.

All the above discussed methods have the problem of time complexity and need proactive information and produce less efficiency in intrusion detection. Most of the methods use host names, set of rules, hop count and payload size to decide whether the packet is malicious. Here we propose a multi model approach for the detection of DDoS attacks to enhance the network security.

6.4 PROPOSED METHOD

We propose a service oriented multi model network inference system to identify the DDoS attacks. In this inference model, we capture the network packets which are approaching the network interface, and the captured packet features will be extracted. The extracted features are
generated as a feature vector or a log and will be stored within the database. Such a log is used to perform analysis of the service provision and used to control the access of the external and internal users on the network.

The method initially splits the whole service history into different groups according to the service id. The grouped service history is formed as another subset of history by splitting them according to the time window. Each history of records from each time window is analyzed and for each service request, we compute various measures like channel access rate, service access rate and hop similarity measure.

For each service accessed they identify the unique path followed by the request and from the set of unique paths available, we identify the common hops present in the path to compute the hop similarity. Similarly, we compute the service access rate using the number of times a service has been accessed, and the channel access rate is computed using how much data has been placed in the channel and so on.

By using all these measures, we compute a legitimate weight which represents the trustworthy of the request which decides whether the request has to be serviced.

The proposed approach has three functional modules namely:

- Packet Tracing
- Service Oriented Analysis
- DDoS attacks detection.
6.4.1 Packet Tracing

The packet enters the network is captured and the features of the packet are extracted. We extract the following features of the incoming packet like TTL, PayLoad, Time, Source Address, Destination Address, Source Port, Destination Port, Hop Count, Hop Addresses, service Name, Service Id. From extracted features, a vector is constructed to be stored in the log trace. The
proposed approach maintains two different logs as genuine and malicious. The extracted feature will be given to the SOA for performing DDoS attack.

**Algorithm:**

**Input:** Raw Packet Rp.

**Output:** Feature vector v.

**Step1:** Convert the Rp into IP packets.

**Step2:** Extract source IP, destination IP, source port, destination port, ttl values.

**Step3:** Extract hop count, hop addresses from communication header.

**Step4:** Identify service id, service name.

**Step5:** Construct feature vector \( v = \{ Sip, Dip, Sport, Dport, ttl, Haddrs, HopCount, ServiceID, Service Name \} \).

**Step6:** Stop.

### 6.4.2 Service Oriented Analysis

At this stage, the extracted features are used to identify whether the packet received is malicious or genuine. The SOA retrieves the previous logs generated by the system and splits them into different groups based on available services. There exist many services in the network and the logs of distinct services are grouped separately. The overall time window is split into number of windows of small size and based on the time window the logs of service groups are split into subgroups. Now we have a small set of service groups where each has logs of distinct service. First, we identify the service to which the packet is belongs and we separate the logs from others. Second, we perform time orient analysis on the logs of the specific service, with the flow
approximation. We compute the service access weight of the packet using which further DDoS detection will be performed.

**Algorithms:**

Input: Feature vector V, Network Trace Nl.

Output: Service Access Rate SAR, Channel Access Rate CAR.

**Step 1:** for each time window Tw_i

Compute Total Number of services Accessed.

end.

**Step 2:** Compute service access rate (SAR) using number of all services accessed from the servicing node and hop counts.

**Step 3:** For each time window Tw_i

Compute total number of channel access.

\[ Tca = \int_{i=0}^{i=N} \sum Nl(Tw_i, Haddress) \]

Compute channel access rate

\[ CAR = \frac{\sum Tca}{\sum Nl(Tw_i All Services) \times THC} \times THC \]

end.

**Step 4:** Stop.

The service oriented architecture based approach has compute the channel access rate, which specifies the frequency and depths of channel access based on the hop counts and addresses and the payload of each request.

Similarly the method computes the service access rate with the number of times a particular service has been accessed and the total number of service access present in the service access history. For each packet
received, the method calculates the service access rate, Channel access rate for the path that the packet has been travelling. Using these values, the method computes the access flag for the packet that is being captured.

6.4.3 Denial of Service Attack Detection

The DDoS attack detection is performed based on computed values of service access rate and channel access rate. With the malicious and genuine history, we compute the Service Denial factor which show the genuiness value, using which we can infer that the packet received is genuine or not. With the malicious history, we compare the pattern of feature and looks for the match to be found. We compute service denial factor for the packet received for the same source address. We collect all the malicious traces and identify the hop address and their pattern then match with the pattern followed by the captured packet to find the attacks. Based on that we compute the service denial factor and then based on pre computed access rates the final inference will be taken.

Algorithm:

Input: Malicious History Mh, Packet Feature V, SAR, CAR
Output: Boolean service-flag.

Step 1: for each history Hᵢ from Mh

Compute total Hops addresses matched

\[ \text{SHa} = \int_{j=0}^{N} \sum \text{HopAddr} (Hᵢ) \in v(Haddr) \]

end

Step 2: Compute Hop similarity value Hsv

\[ Hsv = \frac{\text{SHa}}{\text{Total Number of packets received from Hop address}} \]
**Step3:** Compute Access Rate (AR) using service access rate (SAR) and channel access rate (CAR)

**Step4:**

if AR < ATh and Hsv < HTh then

    return true.

else

    Generate log in the database.

    return false.

end

Here the denial of service has been detected in a combined manner using the access rate, hop similarity value. If the calculated value is and the set of thresholds has been used. By using all these values the method concludes the received packet as genuine one or malicious.

### 6.5 EXPERIMENTAL RESULTS AND PERFORMANCE COMPARISON

The proposed service oriented Multi model network inference model has produced good results. Unlike other immune system, the proposed system addresses both connection and packet based attacks using a service oriented approach. We compute the service access rate and channel access rate for the packet being received and its hop from where the packet has been originated. Also, we compute the hop similarity value for the packet being received with the trace list and finally we conclude the packet based on the thresholds used for access rate and hop similarity. So the proposed approach has good frequency in finding malicious packets and reducing the attacking rate.
6.5.1 Results of the Proposed Work

Figure 6.2 shows the packet details extracted from the received packets through the gateway of the network. It is clearly visible that all the features mentioned in the paper have been extracted for the detection of the genuineness of the packet. The details displayed in the form has various parameters and the protocol name shows the service accessed and the payload details show the payload details of the packet received. The source IP and destination IP values in the form show the source address and destination address of the packet received.

Figure 6.2 The packet features extracted from the received packet
Figure 6.3 The result of proposed approach

The Figure 6.3 shows the details of packets concluded as malicious packets. We have computed the inference weight for all the distinct nodes of the network and the details are furnished clearly.

The figure also depicts the information about the source IP, Source Port, Destination IP, Destination Port, average number of packets, average number of hops, the average payload of the packets received from specific source IP, port to destination IP, port. Also the figure displays the inference weight computed using the above mentioned average values. Finally, with the computed value, the packet has been classified as genuine of malicious.
6.5.2 Performance Analysis of Proposed System

The Figure 6.4 shows the comparison of throughput achieved by the proposed method according to the number of logs available in the trace or service access history.

![Throughput Ratio Graph](image)

**Figure 6.4 Comparison of throughput achieved**

The size of the service access history plays the vital role in achieving throughput and the ratio of throughput is directly proportional to the number of trace it has. If the size of the trace is higher, then the system could achieve higher throughput. So to achieve better throughput more traces has to be maintained in the service access history.

Similarly, the number of trace available also affects the service availability of the network. If there exist an N number of services in the network, then the service availability also affects the quality of service. If the system has more service availability then we can say that the system provides
more quality service and the quality of service of the network is also becoming higher.

Figure 6.5  Comparisons of service availability

The Figure 6.5 shows the comparison of service available according to the number of records available in the service access history. The service availability is also depended on the size of the service access history.

The service availability will be higher, if we maintain more traces of the service access history. With the traces in the service access history it is possible to detect the malicious request accurately, so that the service will be provided to the genuine user. Increase in the service availability, will automatically increase the throughput ratio of the network.

6.5.3  Comparative Analysis of the Proposed Models

Figure 6.6 shows the result of the proposed system in finding malicious packet and if there are 100 packets which are malicious arrived on time, then the graph shows the frequency of detection of malicious packet. It
is very clear that the proposed system identifies the more malicious packet compared to other host based and activity pattern based intrusion detection systems.

**Figure 6.6** The frequency of detection of malicious packet

**Figure 6.7** The time complexity of the proposed system

Figure 6.7 shows the time complexity of the proposed system compared to other methodologies. It shows clearly that the proposed system takes only a little time compared to other methods for different number of
packets. The other methods take more time compare to the proposed system to analyze and detect the intrusion for number of packets.

The Figure 6.8 shows the comparison of false alarm ratio produced by different methods. It shows that the proposed method has less false alarm ratio than the others.

The false alarm is the ratio which represent the frequency of producing false detection like false positive results and false negative results. But both affects the throughput and performance of the whole system which has to be considered well.

The Figure 6.8 displays that the earlier approaches like Host based methods, Network intrusion detection systems have more false alarm ratio. But compare to others the later methods like activity pattern, heterogeneous pattern and service orient methods has less false positive ratio.

**Figure 6.8 Comparison of false alarm ratio**

The Figure 6.8 shows the comparison of false alarm ratio produced by different methods. It shows that the proposed method has less false alarm ratio than the others.
6.6 CHAPTER CONCLUSION

The proposed service oriented Multi model network inference has produced good results. Unlike other immune system, the proposed system addresses both connection and packet based attacks using a service oriented approach. We compute the service access rate and channel access rate for the packet being received and its hop from where the packet has been originated. Also we compute the hop similarity value for the packet being received with the trace list and finally we conclude the packet based on the thresholds used for access rate and hop similarity. So the proposed approach has good frequency in finding malicious packets and reduce the attacking rate.