CHAPTER 5

IMPLEMENTATION OF FLOW BASED MULTI FEATURE INFERENCE MODEL FOR DETECTION OF DDOS ATTACKS IN NETWORK IMMUNE SYSTEM

5.1 PREVIEW

The process of distributed denial of service attack detection can be performed in many ways. The intrusion detection can be identified in a centralized or decentralized manner. The problem is how we are going to identify the denial of service attacks in a distributed manner. There are different methodologies exists to help the detection of DDOS attacks and each uses various methods. Network immune systems have been developed in many ways but differ with the feature set used and suffer with identifying network threats in an efficient manner. We propose a multi feature inference model, which uses various parameters of network flow. Unlike earlier approaches, the proposed method infers valuable knowledge from the packet flow and packet details to detect DDOS attacks. The proposed method uses, hop count, hop details, payload, Time to live with time variant information’s. The network packets are monitored about their traversal, through which they forwarded the packet to the destination. We consider the botnet attacks, which is supported by dedicated nodes distributed throughout the network. Whenever a new packet received form the network, various features are extracted, and we compute the probability of genuine value based on the features. The proposed immune system maintains packet trace for each of the
packet received at different time domains. At each time window, for each distinct traversal path, a probability value is computed using the features extracted from traffic trace. The inferred results are applied to do denial the service for the malicious nodes. The result will be inferred using computed probability value to allow or deny the packet in the network.

5.2 INTRODUCTION

The intrusion detection is the process of identifying malicious packets, which come from malicious users, and also we can define the connections and packets arrived from malicious users. The process of identifying malicious packets coming from various locations of the distributed network is called DDOS attack. The ultimate aim of DDOS detection is to deny the service provided by the network to the identified malicious node or user.

Now a day, the people are using the Internet to perform various activities like banking, shopping, money transfer or for some other works. The user submits valuable personal information through the web pages and navigates through the network. The user information is converted into underlying network packets and forward towards the destination. There are many kinds of attacks that occur in the network like Sybil attack, sink hole attack, modification attack and etc.. We consider the distributed denial of service attack only, and the aim is to identify the malicious packet and user, using that the service has to be denied to the particular user.

The growth of Internet technology increases the growth of network attacks, and also the type of threats is increasing every day. The network threats can be broadly classified as flow based or connection based threats. In flow based approach, the malicious user could generate an
enormous number of packets towards a service point, which is greater than the capacity of the servicing node. Whereas in case of connection based attack, malicious node may hold many numbers of connections without performing any data transfer on that. We focus on the previous one, also it becomes more complicated to distinguish the genuine packets from malicious one. The flow based approaches are available in the literature, from which we conclude that the existing methods use only the number of packets arrived at a time frame, but not considering other features to perform intrusion detection process. The other packet features also have to be used for the intrusion detection and we have to come up with more energetic solutions.

There are various approaches has been proposed in literature, which uses packet details like pay load, hop count and ttl values. Most of the methods suffer with identifying the host which generates malicious packet and will not be effective when there are dynamic addressing nodes present. The growth of Internet technology leads to support the malicious nodes to group compromised nodes to perform flow based attacks. The botnets are set of nodes to form a network with dedicated compromised nodes using which the nodes can generate malicious packets.

DDOS attacks, the more dominating kind of attack generated in networks with the intention to reduce the throughput of the network and reduce the service performance. The DDOS attacks are initiated from the various nodes within the network and send malicious packets towards the servicing node. Whenever a servicing node receives a malicious packet, it spends some time on that and drops the packet finally. This action makes the servicing node not spending time on genuine nodes and the frequency of getting service for the genuine node reduces. Such a malicious node and their packet have to be identified and dropped, and performing that process is
named as distributed denial of service. By adapting DDOS approach, the malicious nodes will not be provided service and whatever the packets come from the malicious nodes will be dropped soon.

Because of the presence of botnets, the network immune systems could not take any decision about the packet, and the decision making becomes more complicated. Even though, every packet approaches the network passes through a different traversal path, identifying them as malicious is impractical.

Generally, the people of any organization have a specific behavior model in their access pattern. For example, if there exists a bank account which belongs to an organization and the account can be accessed by different persons such as the account manager, general manager and the people working in the accounts department. Everybody has rights to access the account, but there will be a different access behavior present in their access mode. The person working in an accounting department is responsible for clearing the cheque’s and has to monitor the credit of amounts and the payment should be done by other business partners and so on. So that, the account’s person will be checking the finance account frequently in his working hours, which will be more frequent in each day in his work.

Unlike an account’s person, the account manager is responsible for maintaining the accounts of all sections and the accounts of all business partners, so that he will be checking all the accounts once or twice a day to come up with the financial management. By monitoring the financial activities of the business partners, he will be able to take decisions about the financial behavior of partners before releasing any fund.
The general manager who is responsible for the whole unit of organization will be checking the financial account, company accounts every day and decides about the financial growth within the company. So he will be accessing the finance accounts once in a week or a day. By accessing this he can conclude about a business partner and generate orders to their subordinates about releasing funds or material to the organizations.

By identifying such behaviors and activities of different users, we can generate the activity patterns or behavior model which represents the behaviors of the user in accessing various services. The network is nothing but a collection of nodes between them in which some of them provide services. The service provided can be accessed by various users independent of their location, and we can monitor their behavior in accessing the services. The services cannot be accessed directly through the network.

5.3 RELATED WORKS

There are various approaches discussed in the literature, and we explore a few of them according to the problem

FireCol proposed by Francois (2012), is a collaborative forensic approach to detect flooding attacks. The method proposes a design and architecture for intrusion detection system. In this approach the firewall is located at the service provider to detect the attack. In this approach the intrusion detection is performed by exchanging the selective traffic between them.

Hyelim Koo (2012) gives a DDoS attack by flooding normal control messages in Kad P2P networks. It introduces a new DDoS attack by flooding control messages from normal users in Kad network, which is DHT-based P2P network. The proposed attack can make normal nodes
participated in the Kad network to act as if they were zombies to generate numerous control messages destined to a target system unintentionally. With the flooded control messages from those nodes, it can cause a DDoS effect to a certain system.

DDoS Attack Detection and Attacker Identification by Brajesh Kashyap et al (2012), describes about various denial of service attacks and proposes various solutions for these types of attacks. In this approach the original malicious user is identified by the spoofing attack or IP forgery. Also, it handles the flow rate to identify the intrusion detection and identifies the intrusion using proxy server based on entropy methods.

DDoS Detection Algorithm Based on Preprocessing Network Traffic Predicted Method and Chaos Theory proposed by Chen et al (2012), pre-process network traffic by cumulatively averaging it with a time range, and using the simple linear AR model, and then generate the prediction of network traffic. Secondly, assuming the prediction error behaves chaotically, we use chaos theory to analyze it and then propose a novel network anomaly detection algorithm (NADA) to detect the abnormal traffic. With this abnormal traffic, we lastly train a neural network to detect DDoS attacks.

Shalinie (2011) proposed a collaborative detection of DDOS attacks, discussed a distributed attack detector which is installed in different locations of the network. At each detector the traffic will be monitored for the flow rate and if there is any malicious flow pattern identified the it produces an alarm to the neighbor routers about the flow and inform about the node from where the flow is produced.
Botnet-based Distributed Denial of Service (DDoS) Attacks on Web Servers: Classification and Art (Esraa Alomari et al 2012), present a comprehensive study to show the danger of Botnet-based DDoS attacks on the application layer, especially on the Web server and the increased incidents of such attacks that has evidently increased recently. Botnet-based DDoS attack incidents and revenue losses of famous companies and government websites are also described. This provides a better understanding of the problem, current solution space, and future research scope to defend against such attacks efficiently.

Detecting and Analyzing DDoS Attack Using Map Reduce In Hadoop by Navale (2014), proposed a new approach to handle Big data. Hadoop technology takes cardinal part in analysis. They proposed detection of DDoS attack by using Counter based algorithm and Access Pattern algorithm which will be implemented in Hadoop framework. Along this we can provide future prediction functionality using analytics. The dashboard provides a visual view which will help to unveil the attacker and loyal user along with statistics.

Poornima (2014) proposed an Intrusion detection using Flow Correlation Coefficient and Feedback. In this approach, initially the normal traffic is gauged and then it has been differentiated with the flash crowd, which is performed using a flash crowd detection algorithm. The flow correlation coefficient is used to detect the intrusion detection. A sequential approach has been discussed to perform intrusion detection and has produced efficient results in intrusion detection.

generation process to represent them in real time. Experimental results are provided to support the proposed mechanism.

All the above discussed approaches have used variations of DDOS detection approach, but suffer with the newly arriving attacks. We propose such a novel approach to handle future attacks and increase the throughput of the network.

5.4 PROPOSED METHOD

To support and provide a better solution with enhanced efficiency, we propose a flow based time variant multi feature enabled inference model. In this model the request produced by any user will be converted into network packets and forwarded through the number of nodes which are located in between the source and the destination where the service is available. So the packet has to travel through the number of intermediate nodes and the number of nodes it has to travel to reach the destination is depending on the routing protocol used.

The packets in the network have to be routed through a number of nodes, which is performed by a routing program hard coded in the router. The router has specific logic to route the packet and forward to a next node according to the logic enforced. The routing logic may be of the shortest path which is performed based on the number of nodes or hops it has to cross to reach the destination. Moreover, the network traffic or congestion also considered in selecting the next hop to forward the packet to reach destination. In this kind of routing the number of hops will not be considered and even a longer path also will be chosen to deliver the packet.
The longer path ultimately increases the time delay which increases the network latency. Furthermore, the energy is the most constraint in different networks like wireless sensor where the nodes are sustained by bounded energy. If the node involves a large number of packet transmissions, then the energy depletion occurred in the node will reduce the lifetime of the network. This problem also has to be considered while choosing the traversal path for any packet.

The service requesting a packet has a specific format and has space for the data to be submitted to the service point. This we called as payload, which has to be minimized so that the bandwidth occupation is reduced to increase the quality of service of the request and to increase the number of service request handling. The identification of malicious nodes can be done using the payload details of the request. Similarly, the TTL value of the packet also can be considered, which performing DDOS attacks, because whatever the condition may be the service, packet has to be delivered on time, if any packet, which is delivered after the TTL time can be considered as malicious packet. Because there may be middle nodes, which captures the packet and performs the modification of data which automatically introduces the delay in the delivery of packets.

The number of hops or hop counts is used to perform DDOS detection, because there may be a shortest path available in the network even at the more congested situation. However, still if the packet has been forwarded through a longer path, then by performing the traffic estimation of the network, the traversal path can be validated for its effectiveness to identify the denial of service attacks.

All the features like hop count, hop addresses, ttl value, payload data and their size can be used to perform the detection of denial of service
attacks. In this paper, we considered the hop count and their hop addresses to perform the detection of botnet attacks. In most networks, the source node will be a genuine one which does not know anything about the changes that occur in the request. However, the genuine node will never get correct access and result from the service point. The intermediate nodes which do not know where a specific service is available and what input has to be given to get the service, captures the packet and study about the pattern of the data. Using this information the attacker sends the packets. In Botnet based attacks, there will be a controller, which captures the network packets flows through that and identifies set of features like the service point details, input parameters and by using all these, the controller could predict the behavior of the service point.

Once the service points access mode has been identified it can generate attacks to degrade the service provided at the service point by anonymously producing enormous amounts of packets to the service point or producing an immoral pattern of input data. The controller will not stop and starts convinces set of neighbors of the network to perform the same. Ultimately, those nodes are also becoming compromised nodes and perform the same which in turn reduces the service quality and throughput of the network.

These kinds of network threats can be identified by using the host names and hop counts. How this can be performed is, in normal traffic pattern for any source to service point, there will be an average number of hops required. The average number of hops will vary in marginal level at high traffic conditions, but when there exists an attacking node in the network then it will be varying hugely. This action will increase the frequency of access by producing the request in higher rate in all the time window of the day through a longer path.
The selection of the longest path is performed to reduce the overall network throughput, because the most nodes will be involved in forwarding the malicious packet, and all of them will lose their energy in transmitting the malicious packets. So in this approach all these features can be used to identify the malicious packet, and identify set of all nodes involved in the denial of service attacks.

We can perform a set of inference from the number of request produced by an user at different time windows like the account’s person, account manager and the general manager. So they all have a different set of access behaviors, which can be considered by the flow of a request produced by them. Such flow will be varied according to the user and the time window. The same account person will have a different flow pattern in accessing his own bank account or mail. All these factors can be used in performing a flow based inference on identifying a malicious flow of request and perform denial of service attacks.

Flow based inference model, is the process of generating the probability value to infer some valuable knowledge using which the packet can be allowed or denied. The flow of packet per each time window can be identified and their payload, ttl values can be used to generate the probability of trustworthiness of the packet.

The proposed method has four functional modules namely:

- Packet Capturing
- Feature Extraction
- Log Generation
- Time variant multi Feature Inference Generation.

We explain in detail in the upcoming chapters about them.
5.4.1 Packet Capturing

The packet capturing is the process of monitoring and catching the incoming packet at the network interface or the router. The ports in the router have to be monitored and we monitor the incoming ports. The packet comes to the input port has to be captured to perform the subsequent operations. The captured packets will be compared with the Malicious list. If the address is in the malicious list, the packet is added to the malicious list otherwise it is given to the next process like feature extraction and intrusion detection.
Algorithm:

**Step1:** Initialize Malicious node list Ml.

**Step2:** Extract packet Address from P

**Step3:** Verify the presence of packet address in Ml.

**Step4:** If found then

    Add Packet to malicious list Ml

else

    continue;

end.

**Step5:** Stop.

5.4.2 Feature Extraction

The captured incoming packets become the input to this function module. This process has to extract various features which are necessary to perform intrusion detection. For each packet received the following features are extracted such as payload, sequence number, hop count, hop addresses, time to live. We cannot directly extract the fields of the packet, in order to extract the features, raw packet is converted into IP packets and to extract the hop count and hop details we convert them into communication header. The ttl value could be extracted from the IP packet. The ttl value was computed using the time available at the time slot of the packet. Extracted packet is converted into a feature vector which represents the packet information in computational form.
Algorithm:

Input: Packet P.
Output: Feature Vector V.

Step 1: initialize V.

Step 2: convert raw packet into IP Packet
Extract packet source Address Saddr.

Step 3: Extract Communication Header CMN_HDR.
HopCount = CMN_HDR(hcount).
TTL = CMN_HDR(ttl).
Hop Address Set Hs = CMN_HDR(ha).

Step 4: Extract payload from p.

Step 5: Construct feature vector using all the fields extracted as follows
V = {Saddr, Daddr, HopCount, TTL, payload, Hs}.

Step 6: Stop.

5.4.3 Log Generation

This function module performs the generation of network trace where the incoming and outgoing packet details has to be stored. The proposed method generates log for each packet received on the network interface of the node attached to the network. For each packet received, we generate a new trace for the packet with the following fields like time, source and destination IP and port, payload, hop count, ttl values, and hop addresses it followed.
Algorithm:

Input: Feature Vector V.
Output: Log L.

Step 1: Read network log.
Step 2: Initialize Log timer.
Step 3: Generate current time.
Step 4: Construct log with payload, hop details, hop count, ip details, port details, ttl value.
Step 5: Insert log into the Log list.
Step 6: Stop.

5.4.4 Time Variant Multi Feature Inference Model & DDOS Attack Detection

At this stage the logs from the trace were taken for processing. First, the traces will be split based on time window. The time will be divided into smaller time frames and the logs split will be named accordingly. For each log of the time window, we compute the common nodes present in the traversal path of a packet and then access rate will be computed. The packet will be identified only based on the network access rate and if it's greater than the threshold, then the particular node will be identified as malicious node. Further the packets received from such a node will be dropped.
Algorithm:

Input: Network Log L.

Output: Inference Value.

**Step1:** Initialize inference value, Network Access Rate, Malicious Node list.

**Step2:** For each time window

    Extract logs at the particular time window

    Extract common nodes present in the logs.

    For each distinct path with the common nodes

        Compute the network access rate.

        \[
        NAR = \frac{\text{Payload}}{\text{TTL}}
        \]

    end

End.

**Step3:** For each Time window \( w_i \)

    If the network access rate is greater than Threshold

        Infer the packet as malicious.

        Mark packet source address as malicious node.

        Add the node address to malicious list.

    End

End.

**Step4:** Stop.
5.5 EXPERIMENTAL RESULTS AND PERFORMANCE COMPARISON

The proposed flow based inference model has been tested using the SonaNet test bed with number of routers, server with large number of nodes in the network. The topology has number of networks interconnected to form a huge network. The packets are captured at different routers and the denial of service attacks being identified in various locations of routers.

5.5.1 Results of the Proposed Work

The Figure 5.2 shows the snapshot of user interface through which the user sends data packets.

![Snapshot of user interface of data transmission](image)

Figure 5.2 Snapshot of user interface of data transmission

The Figure 5.3 shows the result of packet captured and forwarded in the intermediate node of the framework where the packet has been forwarded towards the destination.
Figure 5.3 Snapshot of packets captured in intermediate node

The Figure 5.4 shows the list of messages received at the destination and it shows that the messages are belong to that.

Figure 5.4 Snapshot of the destination interface showing received messages

The Figure 5.5 shows the list of measures computed at the destination point and it has computed the results for hop count, payload, ttl values and shows the list of hops it followed.
Figure 5.5  Snapshot of result produced at the destination showing features

The Figure 5.6 shows the snapshot of the results produced and it shows the computed result of network access rate and the final decision about the packet. It shows that the packets has been classified as malicious.

Figure 5.6  Snapshot of final result of classification
The user have different access rate at different time window, which can be used to identify the malicious flow at any time window. The network access rate is computed based on the number of packets being sent and their pay load. The number of packets sent at any time window from any source node will be different which is used to identify the malicious flow even from legitimate user. We have used these metrics to identify the malicious flow and denial of service attacks.

All these extracted features will be used to perform flow inference by logging the packet information from the service log. The service log consists of more number of records which has the same format and set of information which are produced for each packet received from different nodes of the network. All this information can be used to perform the flow inference to identify the denial of service attacks.

5.5.2 Performance Analysis and Comparison with Existing Systems

The Figure 5.7 shows the time complexity generated by the proposed method, and the proposed method has produced less time complexity at different number of logs available. It shows that the proposed method produces less time for many numbers of logs.

![Figure 5.7 Time complexity of proposed method](image-url)
The Figure 5.8 shows the time complexity value generated for different number of logs available. It shows that the proposed method produces less space complexity even at the high number of logs.

The distribute denial of service attack detection accuracy is computed according to the number of malicious packets arrived at a particular time window and the number of packets analyzed as malicious.

Figure 5.8 Space complexity of proposed method

Figure 5.9 Accuracy of time values
The Figure 5.9, shows the accuracy of DDOS attack detection and it shows that the proposed approach has produced efficient results.

The graph shows that the accuracy of malicious packet detection has growing with the size of log. The detection accuracy of the proposed method varies according to the number of logs available and it increases with the size of trace. And similarly the detection accuracy is keep increasing with the number of logs and it shows the they are directly proportional. So that this becomes a learning model which learns from the number of logs and the efficiency is keep improving with the size of log available.

![](DDOS_Detection_Efficiency.png)

**Figure 5.10 Comparison of DDoS detection efficiency**

The Figure 5.10 shows the comparison of results on detection efficiency produced by different algorithms and it is clear that the proposed method has produced higher efficiency than previous approaches.
5.6 CHAPTER CONCLUSION

We proposed a flow based multi feature inference model for DDOS attack detection. The proposed method captures the packet, and extracts various features to generate the time variant log. For each time window, we identify the set of distinct traversal paths and compute the network access rate using the logs generated. The generated network access rate is used to compute the inference value, using which the packet will be allowed or denied from the network. The proposed method reduces the frequency of threats compared to other approaches and produces less time and space complexity values.