Abstract

Distributed Denial-of-Service (DDoS) attacks are a critical threat to the Internet. A distributed denial-of-service attack is one in which a multitude of compromised systems attack a single target, thereby causing denial of service for users of the targeted system. The flood of incoming messages to the target system essentially forces it to shut down, thereby denying service to the system to legitimate users. The proposed system suggests a mechanism based on entropy variations between normal and DDoS attack traffic. Entropy is an information theoretic concept, which is a measure of randomness. The proposed method employs entropy variation to measure changes of randomness of flows. The implementation of the proposed method brings no modifications on current routing software.

Keywords: Entropy variation; Denial of service

I. INTRODUCTION

In computing, a denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a computer or network resource unavailable to its intended users. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of the efforts of one or more people to temporarily or indefinitely interrupt or suspend services of a host connected to the Internet. Perpetrators of DoS attacks typically target sites or services hosted on high-profile web servers such as banks, credit card payment gateways, and even root name servers. The term is generally used relating to computer networks, but is not limited to this field; for example, it is also used in reference to CPU resource management. One common method of attack involves saturating the target machine with external communications requests, such that it cannot respond to legitimate traffic, or responds so slowly as to be rendered effectively unavailable. Such attacks usually lead to a server overload. In general terms, DoS attacks are implemented by either forcing the targeted computer(s) to reset, or consuming its resources so that it can no longer provide its intended service or obstructing the communication media between the intended users and the victim so that they can no longer communicate adequately.

Distributed Denial-of-Service (DDoS) attacks are a critical threat to the Internet. A distributed denial-of-service attack is one in which a multitude of compromised systems attack a single target, thereby causing denial of service for users of the targeted system[1]. The flood of incoming messages to the target system essentially forces it to shut down, thereby denying service to the system to legitimate users. To launch a DDoS attack, the attacker(s) first establishes a network of computers that will be used to generate the huge volume of traffic needed to deny services to legitimate users of the victim. To create this attack network, attackers discover vulnerable hosts on the network. Vulnerable hosts are those that are either running no antivirus or out-of-date antivirus software, or those that have not been properly patched. These are exploited by the attackers who use the vulnerability to gain access to these hosts. The next step for the attacker is to install new programs (known as attack tools) on the compromised hosts of the attack network. The hosts running these attack tools are known as zombies, and they can be used to carry out any attack under the control of the attacker. A number of IP traceback approaches have been suggested to identify attackers, and there are two major methods for IP traceback, the probabilistic packet marking and the deterministic packet marking. Both of these strategies require routers to inject marks into individual packets. However, the memoryless feature of the Internet routing mechanisms makes it extremely hard to trace back to the source of these attacks. As a result, there is no effective and efficient method to deal with this issue so far. The proposed method proposes a novel traceback method for DDoS attacks that is based on entropy variations between normal and DDoS attack traffic, which is fundamentally different from commonly used packet marking techniques. In comparison to the existing DDoS traceback methods, the proposed strategy possesses a number of advantages—it is memory nonintensive, efficiently scalable, robust against packet pollution, and independent of attack traffic patterns.

II. RELATED WORKS

Distributed Denial-of-Service (DDoS) attacks are a critical threat to the Internet. A number of IP traceback approaches have been suggested to identify attackers, and the following are two major methods for IP traceback node append and the probabilistic packet marking.
A. Node Append

The simplest marking algorithm [2], appending each node’s address to the end of the packet as it travels through the network from attacker to victim. Every packet received in the victim with a complete ordered list of the routers it traversed. It result in a built in attack path. The node append algorithm is both robust and extremely quick to converge (a single packet), however, it has several serious limitations. Principal among these is the infeasibly high router overhead incurred by appending data to packets in flight. Moreover, since the length of the path is not known a priori, it is impossible to ensure that there is sufficient unused space in the packet for the complete list.

Disadvantages:
1) The length of the path is not known prior. It is impossible to ensure that there is sufficient unused space in the packet for the complete list.
2) Packet pollution is another problem; attacker can change the markings.

B. Probabilistic Packet Marking Algorithm

Instead of recording the complete path information of a packet, here only records each edge traversed from the attacker to the victim site in a probabilistic fashion. Reserve two static address sized fields, start and end, in each packet to represent the routers at each end of a link, as well as an additional small field to represent the distance of an edge sample from the victim. When a router decides to mark a packet, it writes its own address into the start field and writes a zero into the distance field. Otherwise, if the distance field is already zero this indicates that the packet was marked by the previous router. In this case, the router writes its own address into the end field there by representing the edge between itself and the previous router and increments the distance field to one. Finally, if the router does not mark the packet, then it always increments the distance field. This somewhat baroque signaling mechanism allows edge sampling to be incrementally deployed-edges is constructed only between participating routers. The mandatory increment is critical to minimize spoofing by an attacker. When the packet arrives at the victim its distance field represents the number of hops traversed since the edge it contains was sampled. Any packets written by the attacker will necessarily have a distance greater or equal to the length of the true attack path. The router determines how the packet can be processed depending on the random number generated. If x is smaller than the predefined marking probability Pm, the router chooses to start encoding an edge. The router sets the start field of incoming packet to the routers address and resets the distance field to zero. If x is greater than predefined probability, Pm router chooses to end encoding an edge by setting the router address in the end field.

III. DETECTION METHOD

The proposed system suggests a novel mechanism for IP traceback using information theoretical parameters, and there is no packet marking in the proposed strategy; so it can avoid the inherited shortcomings of the packet marking mechanisms.

Entropy is an information theoretic concept, which is a measure of randomness. Detection method is based on the entropy variation between normal flow and attack flow. A flow is defined by a pair, source address and destination address.

A. Data set

In order to access the model datasets are used. Data sets are used for profiling the program behavior. Dataset mark the request to a web server for one minute. Each line in dataset represents each request to a web server. It consist of the following columns.
- Host: Host represents the host who make the request. To identify the host, hostname is used when possible otherwise internet address if the name could not be looked up.
- Timestamp: Time is in the following format date, month, year, hour, minute, second.
- Request: The request is given in quotes. It represents to which page the request is given.
- Reply: It represents connection established.
- Bytes in reply.
**B. Matrix Computation**
In Matrix computation, converting large data set into matrix format for computation purpose. In the matrix rows represent distinct users and columns represent time. The time is split into six intervals. So there are six columns. From the matrix it is easy to identify number of requests made by each user in a small time interval. So from this it is easy to identify the flow variations. Matrix computation helps to save memory and improve computation efficiency. Request rate variation can easily found by taking the difference between the numbers of request made in adjacent time intervals.

**C. Entropy Calculations**
Entropy [4] is an information theoretic concept, which is a measure of randomness.
1) Step I: Find the number of request made by each flow in time ‘t’.
2) Step II: Find the number of request made by each flow in time ‘t+Δt’.
3) Step III: Calculate the flow variation between adjacent time intervals.
4) Step IV: Find the probability of each flow over other flows.
5) Step V: Apply the probability derived from step IV in the below equation.

\[
H(F) = -\sum \text{probability of each user} \times \log \text{probability of each user}
\]  
(1)

**D. Detection**
Compare the computed entropy of each distinct flow with a predefined threshold. If computed entropy is less than threshold, then that flow is a normal flow. If computed entropy is greater than threshold, then that flow is an attack flow. The threshold can set by using a normal dataset. Normal dataset contain only normal flows. Based on that normal dataset, find the average entropy and standard deviation of all flows in that dataset. Threshold can be set by using following equation.

\[
\text{Threshold} = \text{average entropy} + \text{standard deviation}
\]  
(2)

**E. Filtering Dos attacker from flash crowd**
When a website is heavenly loaded, normal users cannot access to the website. Due to their impatience they make large number of request. Based on entropy they were also detected as attacker. In order to filter such normal users from attackers list, calculate the time gap between successive requests from each flow. Attackers generate large number of request in small time interval with the help of some attack tools. The time gap between successive requests is very small in attack case.

**IV. Conclusion**
Distributed Denial-of-Service (DDoS) attacks are a critical threat to the Internet. The memory less feature of the Internet routing mechanisms makes it extremely hard to trace back to the source of these attacks. As a result, there is no effective and efficient method to deal with this issue so far. The proposed method proposes a novel traceback method for DDoS attacks that is based on entropy variations between normal and DDoS attack traffic, which is fundamentally different from commonly used packet marking techniques. In comparison to the existing DDoS traceback methods, the proposed strategy possesses a number of advantages—it is memory nonintensive, efficiently scalable, robust against packet pollution, and independent of attack traffic patterns.
REFERENCES