AN EFFECTIVE PREVENTION OF ATTACKS USING GI TIME FREQUENCY ALGORITHM UNDER DDOS

K.Kuppusamy [1] and S.Malathi [2]

[1] Department of Computer Science &Engineering, Alagappa University, Karaikudi
kkdiksamy@yahoo.com
[2] Research Scholar, Manonmaniam Sundaranar University, Tirunelveli
visitmalathi@gmail.com

ABSTRACT

With the tremendous growth of internet services, websites are becoming indispensable. When the number of users gets increased accessing the websites, the performance of the server gets down. Due to much burden on the server, the response time gets delayed. When the process becomes slow, the ratio of the users accessing to the site also goes down. Apart from this, it may also happen due to the attack of Hackers. We have implemented a special kind of technique to recognize the attack carried out by the hackers and block them from using the site. This is termed as Denial of Service and thus is carried out among the web users and is commonly referred to as Distributed Denial of Service (DDoS). To improve server performance and deny the accessibility permissions to the hackers are proposed in this paper.

KEYWORDS

Websites, Attack, Hacker, DDoS

1. INTRODUCTION

In this modern computerized world, large number of new technologies has been emerging. Websites are the common source through which they are made accessible to all.

Websites have the web server which processes the clients' request and send the response to them. The websites become popular either by most of the users access to this site or it may contain most useful information relevant to the users' needs. When the websites become accessible to large number of users it may sometimes lead to overload for the server. The result, the performance of the server goes down. When server performance is low, the response time for the client's request gets increased. So the accessibility of the website becomes reduced.

This is how the website competitors make the site less popular by making its performance very slow. It may also be done by other users to waste the server bandwidth. This kind of performance degradation is termed as Hackers or Intruders. Thus they make the website not to be used by the users. This may be carried out by one of the following ways:

- By sending the request continuously with less time intervals.
- By opening the website and refresh it unnecessarily.
- By using some automation protocol (QTP protocol), access the website to be processed automatically.

Thus by using any one of these ways mentioned above, the intruders will hack the performance of server. When these happen continuously, the users can't get better response time, since the server can't identify the right response from the right users. It just accepts all requests, stores it in queue and sends the response continuously. Thus the hackers will perform faster and thereby reducing the performance quality of the server.

DOI : 10.5121/ijnsa.2011.3619

Thus to tackle these problems, we have proposed a new technology of DDoS(i.e.) to deny the access of the intruders to the website, we have to implement the Distributed Denial of Service technology in a new manner.

A denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a computer resource unavailable to its hackers who prevent the efficient functioning of the sites.

Attempts are made to detect and prevent the intrusion of the hackers to the websites, the DDOS attack technique is carried out. This technique increases the server performance by preventing the intruders from making any intrusion.

2. RELATED WORK

The inter-domain packet filter (IDPF) to mitigate the level of IP spoofing on the internet was proposed in the paper [1]. IDPFs are constructed from the information implicit in BGP route updates and are deployed in network border routers. And also they proposed and studied an inter-domain packet filter (IDPF) architecture as an effective countermeasure to the IP spoofing-based DDoS attacks. IDPFs rely on BGP update messages exchanged between neighboring as is on the Internet to infer the validity of source address of a packet forwarded by a neighbor. It is stated that IDPFs can be easily deployed on the current BGP-based Internet routing architecture.

Distributed Denial of Service (DDoS) attacks pose an increasingly grave threat to the Internet, as evidenced by recent DDoS attacks mounted on both popular Internet sites [3] and the Internet infrastructure [2]. Alarmingly, DDoS attacks are observed on a daily basis on most of the large backbone networks [4].

One of the factors that complicates the mechanisms of policing such attacks is IP spoofing, the act of forging the source addresses in IP packets. By masquerading as a different host, an attacker can hide its actual identity and location, rendering source-based packet filtering less effective. It has been shown that a large part of the Internet is vulnerable to IP spoofing [5], [6].

Recently, there is anecdotal evidence of attackers to stage attacks utilizing bot-nets1 [7]. In this case, since the attacks are carried out through intermediaries, i.e., the compromised .bots, it is tempting to believe that the use of IP spoofing is less of a factor than previously. However, recent studies present evidence to the contrary and show that IP spoofing is still a commonly observed phenomenon [8], [9].Man-in-the-middle attacks, such as variants of TCP hijack and DNS poisoning attacks [10], [11], are carried out by the attacker masquerading as the host at the other end of a valid transaction.

In [12], Li et al., described SAVE, a new protocol for networks to propagate valid network prefixes along the same paths that data packets will follow. Routers along the paths can thus construct the appropriate filters using the prefix and path information. Bremler-Barr and Levy proposed a spoofing prevention method (SPM) [13], where packets exchanged between members of the SPM scheme carry an authentication key associated with the source and destination AS domains.

The idea of IDPF is motivated by the work carried out by Park and Lee [14], which was the first effort to evaluate the relationship between topology and the effectiveness of route based packet filtering. The authors stated that packet filters that are constructed based on the global routing information can significantly limit IP spoofing when deployed in just a small number of ASes. In this work, they extend the idea and demonstrate that filters that are built based on local BGP updates can also be effective.

Unicast reverse path forwarding (uRPF) [15] requires that a packet is forwarded only when the interface that the packet arrives on is exactly the same used by the router to reach the source IP of the packet. If the interface does not match, the packet is dropped. While simple, the scheme is limited given that Internet routing is inherently asymmetric, i.e., the forward and reverse paths between a pair of hosts are often quite different. In Hop-Count Filtering (HCF) [16], each end system maintains a mapping between IP address aggregates and valid hop counts from the origin to the end system. Packets that arrive with a different hop count are suspicious and are therefore discarded or marked for further processing.

The Bogon Route Server Project [17] maintains a list of bogon network prefixes that are not routable on the public Internet. Recently IP trace-back mechanisms based on probabilistic packet marking (PPM) have been proposed for achieving trace-back of DoS attacks.

Effective mitigation of denial of service (DoS) attack is a pressing problem on the Internet. In many instances, DoS attacks can be prevented if the spoofed source IP address is traced back to its origin which allows assigning penalties to the offending party or isolating the compromised hosts and domains from the rest of the network. Recently IP trace-back mechanisms based on probabilistic packet marking (PPM) have been proposed for achieving trace-back of DoS attacks.

In the paper[18] shows that probabilistic packet marking—of interest due to its efficiency and implementability vis-`a-vis deterministic packet marking and logging or messaging based schemes—suffers under spoofing of the marking field in the IP header by the attacker which can impede trace back by the victim.

It shows there is a trade-off between the ability of the victim to localize the attacker and the severity of the DoS attack, which is represented as a function of the marking probability, path length, and traffic volume. The optimal decision problem—the victim can choose the marking probability whereas the attacker can choose the spoofed marking value, source address, and attack volume—can be expressed as a constrained mini-max optimization problem, where the victim chooses the marking probability such that the number of forgeable attack paths is minimized.

It also shows the attacker's ability to hide his location is curtailed by increasing the marking probability; however, the latter is upper-bounded due to sampling constraints. In typical IP internets, the attacker's address can be localized to within 2–5 equally likely sites which render PPM effective against single source attacks. Under distributed DoS attacks, the uncertainty achievable by the attacker can be amplified, which diminishes the effectiveness of PPM.

Denial of service (DoS) is a pressing problem on the Internet as evidenced by recent attacks on commercial servers and ISPs and their consequent disruption of services [19]. DoS attacks [20], [21], [22], [23], [24], [25] consume resources associated with various network elements—e.g., Through servers, routers, firewalls, and end hosts—which impedes the efficient functioning and provisioning of services in accordance with their intended purpose.

A number of recent works have studied the problem of tracing the physical source of a DoS attack [23]. Several types of DoS attacks have been identified [19], [21], [23],[24] with the most basic DoS attack demanding more resources than the target system or network can supply. Resources may be network bandwidth, file system space, processes, or network connections [23]. While host-based DoS attacks are more easily traced and managed, network-based DoS attacks which exploit their accessibility of the TCP/IP protocol suite represent a more subtle and challenging threat [23]. Network-based DoS attacks, by default, employ spoofing to forge the source address of DoS packets to hide the identity of the physical source [25].

During a DoS attack, an attacker may try to gauge the impact of the attack using various service requests including them and ICMP echo requests. Thus, logging of such events and activities can disclose information about the attacker's source. The victim uses information inscribed in packets to trace the attack back to its source. In both methods, overhead in the form of variablelength marking fields that depend on path length or traffic overhead due to extra messaging packets are incurred.

Probabilistic packet marking [23] achieves the best of both worlds—space efficiency in the form of constant marking field and processing efficiency in the form of minimal router support—at the expense of introducing uncertainty due to probabilistic sampling of a flow's path. The latter has two important, and opposing, effects: (a) discovery of correct path information by sampling which aids the victim's objective of trace-back, and (b) injection of corrupted information by the attacker.

In the latter, with a certain probability a packet—however formatted by the attacker—will travel through untouched, and can impede the victim's ability to identify the true attack path. More generally, the number of forgeable paths that are from an information-theoretic point-of-view indistinguishable with respect to their validity from the true attack path can further render source identification difficult if their numbers are large.

Paper [18] shows the critical issue —the attacker's ability to inject misleading information—and give a comprehensive analysis of the effectiveness of PPM under single-source and distributed DoS attacks, complemented by numerical evaluations. They remark that PPM is not perfect and suffers under two additional they access (they are not unique to PPM, however, and are shared by the other approaches).

First, PPM is reactive in the sense that damage must occur before corrective actions— including source identification—can be undertaken by the victim. Second, PPM does not scale they all under distributed DoS (DDoS) attacks in the sense that the more hosts an attacker is able to compromise and use as a distributed attack site, the greater the effort needed (approximately proportional) to identify the attack sites.

Firewalls offer a protection for private networks against both internal and external attacks. However, configuring firewalls to ensure the protections is a difficult task. The main reason is the lack of methodology to analyze the security of firewall configurations. IP spoofing attack is an attack in which an attacker can impersonate another person towards a victim.

3. METHODOLOGY

3.1. Proposed Method

The aim of the proposed method is to develop an efficient method in order to deny the services to the hackers and improve the server performance using the DDoS technique.

This is summed up below: In order to detect the intruders, the entry of all users and their activities are maintained as history. The history also contains the information about the users with their corresponding entry time, date and their accessing site. Based on the history, we can identify all the users accessing the server.

Each user entering the internet is assigned a unique IP address. This IP address is also stored in the history along with the users' entry details. Based on this IP address, we can identify the particular user. This identification is successfully done by grouping the IP addresses from the history and count the number of occurrence of the same IP address under the same date.

If for example, the same IP address such as 192.323.2.3 is found occurring repeatedly under the same date, then their time of entry into the site is retrieved correspondingly and counts the number of occurrence. Thus we identify the user who utilizes the site for the maximum number of times on the same day.

The next step is to determine the time frequency of that user using our proposed algorithm named **GI (Group Intruders) Time Frequency Algorithm**. The time frequency is determined by calculating the time difference between the each entry time by using the relation,

$$\boldsymbol{T_{ij} = t_j - t_i} \qquad \rightarrow (1)$$

where,
T_{ij} is the difference between the time t_j and $t_{i.}$
t_j and t_i are the time in i^{th} and j^{th} entry of the user.

After calculating the time difference between each set, the average mean time difference is determined by using the relation,

$$\boldsymbol{T_m = \sum T_{ij}} \qquad \rightarrow (2)$$

where,
T_m is the mean time difference calculated from the sum of all time difference $T_{ij.}$

While calculating the mean time difference, the frequency is calculated by dividing the mean time by the number of times occurred. The relation is shown as below:

$$\boldsymbol{T_f = T_m / n} \qquad \rightarrow (3)$$

where,
T_f is the time frequency calculated.
T_m is the mean time difference found using the relation (2) n is the number of time the particular IP Address occurred.

A frequency limitation is set by us as **N** and now, the calculated time frequency is compared with this N frequency. When the calculated frequency is greater than the N frequency, then that IP address is treated as Hackers IP address and so the user is added to Intruders List to prevent their access further.

The IP address in the Intruders List is maintained permanently in order to check the upcoming user. If the user in the list tries to enter again, then the access permission is denied by not giving any response to that kind of users, using the DDoS mechanism. If other users enter into the site, the history is maintained in order to determine their performance.

The proposed method consists of a GI Time Frequency algorithm. All the required validation processes will be taken in consideration by the proposed method. The following provides the description about the proposed method.

3.2 GI Time Frequency Algorithm

```
Begin
Maintain the Intruders List, I
Maintain the History of the user, H
User Entered into the site, User.
Get the IP Address, Date, Time of the user and store the details in the history, H.
if (User.IP == I.IP)
{
        Type = "Existing Intruder"
        Print: Access Denied.
        Break
}
Else if (User.IP == H.IP)
```

```
{
        if (User. Date == H.Date)
        {
        Type = “New Intruder”
        }
}
if ( Type == “New Intruder”)


{
        Get the time from the history for the User.IP
        Calculate the time difference, $T_{ij} = t_j – t_i$
        Calculate the average mean time, $T_m = \sum T_{ij}$
        Find the number of occurrence, n.
        Calculate the time frequency, $T_f = T_m / n$
        Find the Maximum frequency, N.
        if ($T_f > N$)
        {
           Add the User.IP to the Intruder List, I
           Print : “Access Denied”
        }
}
Else
{
        Accept the request from the User.IP
        Send the response for the request.
}
End
```

3.3 Algorithm Explanation

The GI Time Frequency Algorithm is used to group the intruders under the Intruders list and thus prevent them from accessing the website. First step of the algorithm is to maintain the history of the user and the intruders list. When the user enters into the site, the details are collected and added in the history. Then the details are matched with the intruders list. If the match returns true value, then the user is treated as intruder and the access is denied. Otherwise, the details are matched with the history for finding the occurrence of the same user under the same date. If this returns true, then the time frequency is calculated. The time frequency is compared with the maximum frequency. If the calculated time frequency exceeds the maximum frequency, the user is added to the intruders list. Otherwise, their request is accepted and the response is provided to the user. Thus the GI Time Frequency provides a better method to block the intruders from accessing the web page.

3.4 Flow Chart

The diagrammatic representation of the flow of the GI Time Frequency algorithm is given as a flowchart below:

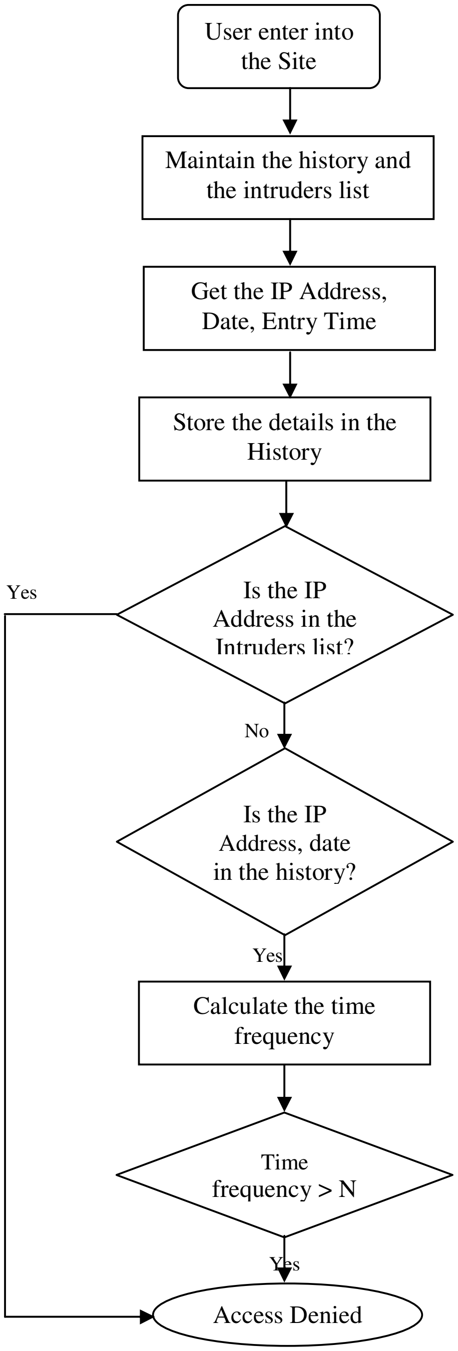
User enter into the Site
Maintain the history and the intruders list
Get the IP Address, Date, Entry Time
Store the details in the History
Is the IP Address in the Intruders list?
Yes
No
Is the IP Address, date in the history?
Yes
Calculate the time frequency
Time frequency > N
Yes
Access Denied

Figure-1 Process Flow of the Algorithm

4. EXPERIMENTAL RESULTS

The experimental results of this paper are carried out by taking a set of intruder list and the website. The browser maintains the history of the user and at the same time the details of the history are tabulated with the fields such as Date, Time, and IP Address. Based on the IP Address, each incoming user is analyzed.

When the new user enters into the site frequently, the algorithm is implemented to determine whether the user is intruder. If not, proper response is provided to the user.

The experimental setup is carried out with two different situations. At first, the experiment is carried out without any intrusion detection or any DDoS prevention. In that situation, normal performance of the web server is found and noted. When the intruders are allowed to access the site, the performance in this situation is also calculated and noted.

At the second part, the intruder list is maintained and checked the user with the list. If the intruders are found, the access is denied by implementing the GI Time Frequency Algorithm. In this situation, the web server performance is noted. And thus the comparison is made between the two experimental setups. This helps the users to determine the efficiency of our proposed algorithm named as **GI Time Frequency Algorithm.**

Thus the implementation of the DDoS to prevent the server from accessing the server and lower the performance of the server is meted out successfully in this system.

5. CONCLUSION

The aim of the paper is to propose a method to detect the intruders accessing the website unnecessarily minimizing the performance ratio of the server. Such intruders are detected using a special technique which is proposed in this paper, and their access is prevented by using the DDoS technique.

A special algorithm named GI Time Frequency Algorithm is implemented in this paper to group the detected intruders and prevent them from accessing to the website and thereby the quality of the server performance is maintained.

REFERENCES

[1] "Constructing Inter-Domain Packet Filters to Control IP Spoofing Based on BGP Updates" by Zhenhai Duan, Xin Yuan, Jaideep Chandrashekar.

[2] Massive DDoS attack hit DNS root servers. http://www. internetnews.com/ent-news/article.php/1486981, October 2002.

[3] Yahoo attributes a lengthy service failure to an attack. http://www.nytimes.com/library/tech/00/02/biztech/ articles/08yahoo.html%, February 2000.

[4] Craig Labovitz, Danny McPherson, and Farnam Jahanian. Infrastructure attack detection and mitigation. SIGCOMM 2005, August 2005. Tutorial.

[5] R. *Beverly. Spoofer project. http://momo.lcs.mit.edu/ spoofer*.

[6] R. Beverly and S. Bauer. The Spoofer Project: Inferring the extent of Internet source address _ltering on the internet. In Proceedings of Usenix Steps to Reducing Unwanted Traf_c on the Internet Workshop SRUTI'05, Cambridge, MA, July 2005.

[7] Srikanth Kandula, Dina Katabi, Matthais Jacob, and Arthur Berger. Botz-4-Sale: Surviving Organized DDoS Attacks that Mimic Flash Crowds. In Second Symposium on Networked Systems Design and Implementation (NSDI'05)., 2005.

[8] D. Moore, G. Voelker, and S. Savage. Inferring internet Denial-of-Service activity. In Proceedings of 10th Usenix Security Symposium,August 2001.

[9] R. Pang, V. Yegneswaran, P. Barford, V. Paxson, and L. Peterson. Characteristics of internet background radiation. In Proceedings of ACM Internet Measurement Conference, October 2004.

[10] M. Dalal. Improving TCP's robustness to blind in-window attacks. Internet Draft, May 2005. Work in Progress.

[11] J. Stewart. DNS cache poisoning - the next generation. Technical report, LURHQ, January 2003.

[12] J. Li, J. Mirkovic, M. Wang, P. Reiher, and L. Zhang. SAVE: source address validity enforcement protocol. In INFOCOM, June 2002.

[13] Bremler-Barr and H. Levy. Spooling prevention method. In Proc. IEEE INFOCOM, Miami, FL, March 2005.

[14] K. Park and H. Lee. On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law internets. In Proc. ACM SIGCOMM, San Diego, CA, August 2001.

[15] F. Baker. Requirements for IP version 4 routers. RFC 1812, June 1995.

[16] C. Jin, H. Wang, and K. Shin. Hop-count filtering: an effective defense against spoofed ddos traffic. In Proceedings of the 10th ACM conference on Computer and communications security, October 2003.

[17] Team Cymru. The team cymru bogon route server project. http: //www.cymru.com/BGP/bogon-rs.html.

[18] "On the Effectiveness of Probabilistic Packet Marking for IP Traceback under Denial of Service Attack" by Kihong Park, Heejo Lee, Network Systems Lab, Department of Computer Sciences, Purdue University.

[19] Lee Garber, "Denial-of-service attacks rip the Internet," Computer, pp.12–17, Apr. 2000.

[20] John Elliott, "Distributed denial of service attack and the zombie ant effect," IT Professional, pp. 55–57, March/April 2000.

[21] Jari Hautio and Tom Weckstrom, "Denial of service attacks," Mar. 1999, http://www.hut.fi/u/tweckstr/hakkeri/DoS paper.html.

[22] John D. Howard, An Analysis of Security Incidents on the Internet, Ph.D. thesis, Carnegie Mellon University, Aug. 1998.

[23] Night Axis and Rain Forest Puppy, "Purgatory 101: Learning to cope with the SYNs of the Internet," 2000, some practical approaches to introducing accountability and responsibility on publicinternet,http://packetstorm.securify.com/papers/contest/RFP.doc.

[24] Computer Emergency Response Team, "Denial of service," Feb. 1999, Tech Tips, http://www.cert.org/tech tips/denial of service.html.

[25] Computer Emergency Response Team (CERT), "CERT Advisory CA-2000-01 Denial-of-service developments," Jan. 2000, http://www.cert.org/advisories/CA-2000-01.html.

Authors

1. **Dr.K.Kuppusamy** is working as an Associate Professor in the Department of Computer Science and Engineering, Alagappa University, Karaikukdi, Tamilnadu, India. He received his Ph.D in Computer Science and Engineering from Alagappa University, Karaikudi, Tamilnadu in the year 2007. He has 23 years of teaching experience at PG level in the field of Computer Science. He has published many papers in International & National Journals and presented in National and International conferences. His areas of research interests include Information/Network Security, Algorithms, Neural Networks, Fault Tolerant Computing, Software Engineering and Optimization Techniques.

2. **Mrs.S.Malathi** is working as a Lecturer and Head in the Department of Computer Science, Rabiammal Ahamed Maideen College, Tiruvarur, Tamilnadu, India. She has 12 years of teaching experience in the field of Computer Science. She has guided around 10 M.Phil., scholars. She has published one book and one research paper. Her area of interest is Network Security.