Middle-East Journal of Scientific Research 20 (3): 353-358, 2014
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2014.20.03.11317

An Efficient Methods to Prevent Denial of Service Attack

S. Brintha Rajakumari

Department of CSE,
B. I. S. T. (Bharath University), Chennai, India

Abstract: The recent increase of their power and their use by organized criminal organizations make necessary to consider them as one of the major issues IT infrastructures will have to face in the next few years. Trying to defeat those attacks without understanding their technical aspects is illusory. In this paper, we propose to use this web site graph structure to mitigate flooding attacks on a website, using new web referral architecture for privileged service (WRAPS). WRAPS allows a legitimate client to obtain a privilege URL through a simple click on a referral hyperlink, from a website trusted by the target website. Using that URL, the client can get privileged access to the target website in a manner that is far less vulnerable to a distributed denial-of-service (DDoS) flooding attack than normal access would be. Our empirical study demonstrates that WRAPS enables legitimate clients to connect to a website smoothly in spite of a very intensive flooding attack, at the cost of small overheads on the website's ISP's edge routers. We discuss the security properties of WRAPS and a simple approach to encourage many small websites to help protect an important site during DoS attacks.

Key words: Dos Attack.

INTRODUCTION

Imagine an executive of a financial institution deprived of access to the stock market updates for several hours or even several minutes. In [1-10], the authors showed that whereas 50% of the attacks lasted less than ten minutes, unfortunately, 2% of them lasted greater than five hours and 1% lasted more than ten hours. There were dozens of attacks that spanned multiple days. Wide spectrum of motivation behind these DoS attacks exists. They range from political conflicts and economic benefits for competitors to just curiosity of some computer geeks. Furthermore, cyber terrorism may not be excluded in the future [10-15].

Attacks on the data forwarding process are of a more serious nature. These attacks inject traffic into the network with the intent to steal band- width or to cause QoS degradation for other flows. Since the differentiated services framework is based on aggregation of flows into service classes, legitimate customer traffic may experience degraded QoS as a result of the illegally injected traffic. Taken to an extreme, that excess traffic may result in a denial of service attack. This creates a need for developing an effective defence mechanism that automates the detection and reaction to attacks on the QoS-enabled networks [15-20].

In this paper, we first elaborate on the denial of service attacks and their potential threat on the system. We then classify the solutions proposed in the literature into two main categories: detection and prevention approaches. In addition, we propose network monitoring techniques to detect service violations and to infer DoS attacks. We believe that network monitoring has the potential to detect DoS attacks in early stages before they severely harm the victim [21-23].

DoS Attacks:

Detection and Prevention: In the literature, there are several approaches to deal with denial of service (DoS) attacks. In this section, we provide an approximate taxonomy of these approaches. In addition, we briefly describe the main features of each approach and highlight the strengths and weaknesses of it.

We divide the approaches for dealing with DoS attacks into two main categories: detection and prevention approaches. The detection approaches capitalize on the fact that appropriately punishing wrong doers (attackers) will deter them from re-attacking again

Corresponding Author: S. Brintha Rajakumari , Department of CSE, B. I. S. T. (Bharath University), Chennai, India.

and will scare others to do similar acts. The detection process has two phases: detecting the attack and identifying the attacker. To identify an attacker, several trace back methods can be used, as explained later in this section. The obvious way to detect an attack is just waiting till the system performance decreases sharply or even the whole system collapses. We propose a more effective method for detecting attacks before they severely harm the system. We propose to use monitoring for early detection of DoS attacks. The prevention approaches, on the other hand, try to thwart attacks before they harm the system. Filtering is the main strategy used in the prevention approaches [24].

DoS Attacks: The aim of a DoS attack is to consume the resources of a victim or the resources on the way to communicate with a victim. By wasting the victim's resources, the attacker disallows it from serving legitimate customers. A victim can be a host, server, router, or any computing entity connected to the network. Inevitable human errors during software development, configuration and installation open several unseen doors for these types of attacks.

TCP SYN flooding is an instance of the flooding attacks [22]. Under this attack, the victim is a host and usually runs a Web server. A regular client opens a connection with the server by sending a TCP SYN segment. The server allocates buffer for the expected connection and replies with a TCP ACK segment. The connection re- mains half-open (backlogged) till the client acknowledges the ACK of the server and moves the connection to the established state. If the client does not send the ACK, the buffer will be deal located after an expiration of a timer. The server can only have a specific number of half-open connections after which all requests will be refused. The attacker sends a TCP SYN segment pretending a desire to establish a connection and making the server reserves buffer for it. The attacker does not complete the connection. The tool watches for SYN segments coming from spoofed IP ad- dresses and sends TCP RST segments to the server. The RST segments terminate the half-open connections and free their associated buffers. Other types of flooding attacks include TCP ACK and RST flooding, ICMP and UDP echo-request flooding, a DNS request flooding [16, 24]. This list is by no means exhaustive.

A DoS attack can be more severe when an attacker uses multiple hosts over the Internet to storm a victim. To achieve this, the attacker compromises many hosts and deploys attacking agents on them. The attacker

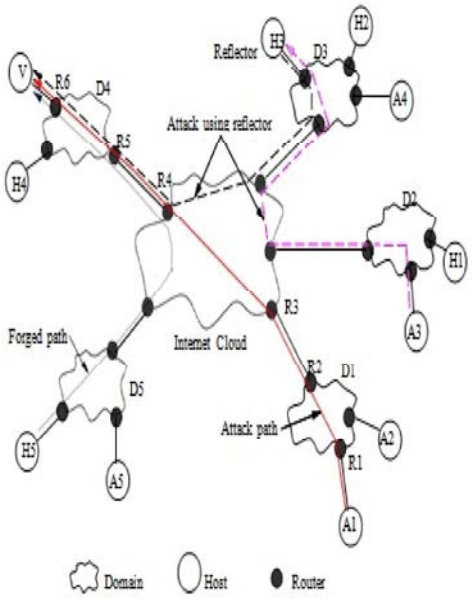

Fig. 1: Different scenarios for Dos attacks A1 launches an attack on the Victim V A1 spoofs IP address of host H5 from domain D5 Another attacker A3uses host H3 as a reflector attack V,

signals all agents to simultaneously launch an attack on a victim. Barros [1] shows that DDoS attack can reach a high level of sophistication by using reflectors. A reflector is like a mirror that reflects light. In the Internet, many hosts such as Web servers, DNS servers and routers can be used as reflectors because they always reply to (or reflect) specific type of packets. Web servers reply to SYN requests, DNS servers reply to queries and routers send ICMP packets (time exceeded or host unreachable) in response to particular IP packets [25]. The attackers can abuse these reflectors to launch DDoS attacks. For example, an attacking agent sends a SYN request to a reflector specifying the victim's IP address as the source address of the agent. The reflector will send a SYN ACK to the victim. There are mil- lions of reflectors in the Internet and the attacker can use these reflectors to flood the victim's network by sending a large amount of packets. Paxson [20] analyses several Internet protocols and applications and concludes that DNS servers, Gnutella servers and TCP-based servers are potential reflectors.

Detection Approaches: The detection approaches rely on finding the malicious party who launched a DoS attack and consequently hold him liable for the damage he has caused. However, pinning the real attacker down is not a

straightforward task. One reason is that the attacker spoofs the source IP ad- dress of the attacking packets. Another reason is that the Internet is stateless, which means, whenever a packet passes through a router, the router does not store any in- formation (or traces) about that packet. Therefore, mechanisms such as ICMP traceback and packet marking are devised to figure out the real attacker. In this subsection, we describe several techniques to identify the attacker after the attack took place [26].

Prevention Approaches: Preventive approaches try to stop a DoS attack by identifying the attack packets and discarding them before reaching the victim. We summarize several packet filtering techniques that achieve this goal.

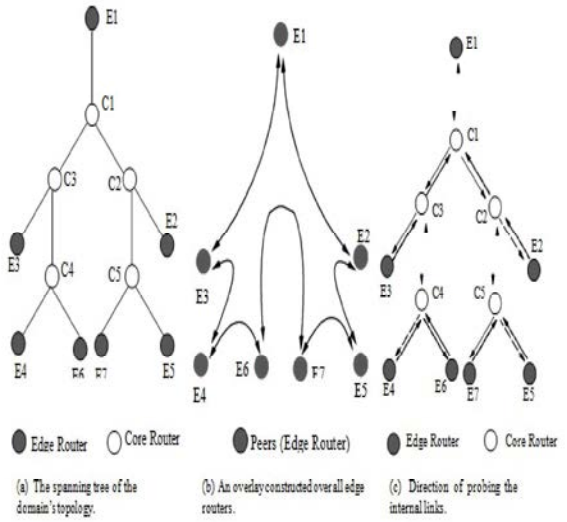

Fig. 2: Missing

Monitoring to Detect Service Violations and Dos Attacks: In this section, we show how network monitoring techniques can be used to detect service violations and to infer DoS attacks. We believe that network monitoring has the potential to detect DoS attacks in early stages before they severely harm the victim. Our conjecture is that a DoS attack injects a huge amount of traffic into the network, which may alter the internal characteristics (e.g. delay and loss ratio) of the network. Monitoring watches for these changes and our proposed techniques can identify the congested links and the points that are feeding them.

We describe the monitoring schemes in the context of a QoS-enabled network, which provides different classes of service for different costs. The schemes are also applicable to best effort (BE) networks to infer DoS attacks, but not to detect service violations because there is no notion of service differentiation in BE networks.

Table 1: Symbols used in the comparison and their values.

Symbol	Description	Values used in comparison
P_{sch}	Processing overhead for scheme sch	–
C_{sch}	Communication overhead for scheme sch	–
M	Number of edge routers	[10 – 20]
N	Number of core routers	12
F	Number of flows entering through each edge router	100,000
P	Number of packets per flow	10
p	Probability to mark a packet	[0 – 0.20]
θ	Percentage of misbehaving flows	[0 – 20%]
h	Path length inside a domain or hop count	4, 6
s	Length of a stripe	3
f_s	Frequency of stripe per unit time in stripe-based monitoring	20
f_d	Frequency of probes per unit time in distributed monitoring	30
α_1	Processing overhead for filtering	–
α_2	Processing overhead for marking	–
α_3	Processing overhead for monitoring	–

To monitor a domain, we measure three parameters: delay, packet loss ratio and throughput. We refer to these parameters collectively as the service level agreement (SLA) parameters, since they indicate whether a user is achieving the QoS requirements contracted with the net- work provider. In our discussion, delay is the end-to-end latency; packet loss ratio is defined as the ratio of number of dropped packets from a flow2 to the total number of packets of the same flow entered the domain; and through- put is the total bandwidth consumed by a flow inside the domain. Delay and loss ratio are good indicators for the current status of the domain. This is because, if the do- main is properly provisioned and no user is misbehaving, the flows traversing through the domain should not experience high delay or loss ratio inside that domain. It is worth mentioning that delay jitter, i.e. delay variation, is another important SLA parameter. However, it is flow- specific and therefore, is not suitable to use in network monitoring [27].

The SLA parameters can be estimated with the involvement of internal (core) routers in a network domain or can be inferred without their help. We describe both core- assisted monitoring and edge-based (without involvement of core routers) monitoring in the following subsections.

Core-Based Monitoring: A core-based monitoring scheme for QoS-enabled net- work is studied in [11]. In this scheme, the delay is measured by having the ingress routers randomly copy the header of some of the incoming packets. The copying depends on a

pre-configured probability parameter. The ingress router forms a probe packet with the same header as the data traffic, which means that the probe packet will likely follow the same path as the data packet. The egress router recognizes these probe packets and computes the delay.

This monitoring scheme measures the loss ratio by collecting packet drop counts from core routers. It then contacts the ingress routers to get the total number of packets for each flow. The loss ratio is computed from these two numbers. To measure the throughput, the scheme polls the egress routers. The egress routers can provide this information because they already maintain this information for each flow.

Edge-Based Monitoring: We describe two edge-based monitoring schemes: stripe-based and distributed. Both schemes measure delay and throughput using the same techniques as the previous core-based scheme. They differ, however, in measuring the packet loss ratio.

When delay goes high, the SLA monitor triggers agents at different edge routers to probe for loss. Each edge router probes its neighbours. Let Xñ be a Boolean random variable that represents the output of probe ñ. Xñ takes on value 1 if the measured loss exceeds the threshold in any link throughout the probe path and takes on 0 otherwise [28].

Violation and DoS Detection: In both the stripe-based and distributed-based monitoring schemes, when delay, loss and bandwidth consumption exceed the pre-defined thresholds, the monitor decides on possible SLA violation. The monitor knows the existing traffic classes and the acceptable SLA parameters per class. High delay is an indication of abnormal behaviour inside the domain. If there is any loss for the guar- anteed traffic class and if the loss ratios of other traffic classes exceed certain levels, an SLA violation is flagged. This loss can be caused by some flows consuming band- width beyond their SLA. Bandwidth theft is checked by comparing the total bandwidth achieved by a user against the user's SLA for bandwidth. The misbehaving flows are controlled at the ingress routers.

Future Work in Emergency Calling: The emergency call delay is bigger than the non-emergency one because it includes also a location parsing and database query for the nearest PSAP URI. In the near future we are considering analysing the performances of the Emergency Branch of Open IMS Core in order to find out what could be the possible optimisations of the message processing during an emergency call [28].

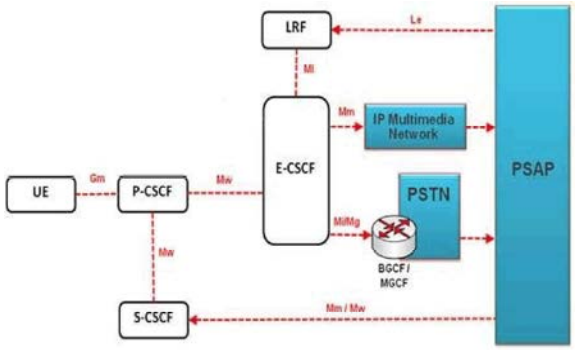

Fig. 3: Emergency support Framework for IMS

CONCLUSION

In this work presented a simple but effective solution to defend special kinds of distributed Denial of Service attacks on SIP based networks, particularly IMS. The solution adds a marginally longer delay for regular users, but keeps the proxy entity, in this case the P-CSCF, safe from overhead traffic. One major problem encountered is the low performance of the used Linux iptables firewall, which is neither capable of handling dynamic rule modification nor scales well beyond 20000 rules in real-time scenarios.

The presented comparative study showed several issues. First, it showed that while marking imposes less overhead than filtering, it is only a forensic method. Second, the core- based monitoring scheme has a high deployment cost be- cause it needs to update all edge as well as core routers. However, the core-based scheme has less processing over- head than the stripe-based scheme because it aggregates flow information when it reports to the monitor. Third, the stripe-based monitoring scheme has lower communication overhead than the core-based scheme for relatively small size domains [24-33]. For large domains, however, core- based may impose less communication overhead depending on the attack intensity. Fourth, he distributed scheme outperforms the other monitoring schemes in terms of deployment cost and overhead in many of the cases.

REFERENCES

1. http://www.research.att.com/lists/ietfitrace/2000/09/msg00044.html,.
2. Bellovin, S.M., 2000. ICMP traceback messages. Internet draft: draft-bellovin-itrace-00.txt, Mar.
3. Blake, S., D. Black, M. Carlson, E. Davies Z. Wang and W. Weiss. An architecture for Differentiated Services. RFC 2475.

4. Breitbart, Y., C.Y. Chan, M. Garofalakis, R. Rastogi and A. Silberschatz, 2001. Efficiently monitoring bandwidth and latency in IP networks. In Proc. IEEE INFOCOM, Anchor- age, AK.
5. Burch, H. and H. Cheswick, 2010. Tracing anonymous packets to their approximate source. In Proc. USENIX LISA, pp: 319- 327. New Orleans, LA, Dec.
6. R. Cáceres, N., G. Duffi eld, J. Horowitz and D. Towsley, Multicast-based inference of network-internal loss characteristics. IEEE Transactions on Information Theory.
7. Chan, M.C., Y.J. Lin and X. Wang, 2000. A scalable monitoring approach for service level agreements validation. In Pro. International Conference on Network Protocols (ICNP), pp: 37- 48, Osaka, Japan.
8. Dilman, M. and D. Raz, 2001. Efficient reactive monitoring. In Proc. IEEE INFOCOM, Anchorage, AK.
9. Duffi eld, N.G., F.L. Presti, V. Paxson and D. Towsley, 2001. Inferring link loss using striped unicast probes. In Proc. IEEE INFOCOM, Anchorage, AK.
10. Ferguson, P. and D. Senie, 2010.Network ingress fi ltering: Defeating denial of service attacks which employ IP source address spoofi ng agreements performance monitor- ing. RFC 2827.
11. Habib, A., S. Fahmy, S.R. Avasarala, V. Prabhakar and B. Bhargava, 2010. On detecting service violations and band- width theft in QoS network domains. Journal of Computer Communicatons (to appear).
12. Habib, A. M. Khan and B. Bhargava, 2002. Edge-to-edge measurement-based distributed network monitoring. Technical report, CSD-TR-02-019, Purdue University, Sept.
13. Institute, S. Egress filtering http://www. sans. org/y2k/ egress.htm,.
14. Garber, L. Denial of Service attacks rip the Internet. IEEE Computer, 33(4): 12- 17.
15. Mahajan, M., S.M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson and S. Shenker, Controlling high bandwidth aggre- gates in the network. ACM Computer Communication Re- View, 32(3): 62- 73.
16. Moore, D., G.M. Voelker and S. Savage, Inferring Inter- net denial-of-service activity. In Proc. USENIX Security Symposium, Washington D.C.
17. Park, K. and H. Lee, On the effectiveness of probabilistic packet marking for IP traceback under Denial of Service attack. In Proc. IEEE INFOCOM, Anchorage, AK, Apr.
18. Park, K. and H. Lee, 2001. A proactive approach to distributed DoS attack prevention using route-based packet fi ltering. In Proc. ACM SIGCOMM, San Diego, CA, Aug.
19. Paxson, V., 1997. End-to-end internet packet dynamics. In Proc. SIGCOMM '97, Cannes, France.
20. Paxson, V., An analysis of using reflectors for distributed denial-of-service attacks. ACM Computer Communication Review.
21. Savage, S., D. Wetherall, A. Karlin and T. Anderson, 2001. Network support for IP traceback. IEEE/ACM Transaction on Networking, 9(3): 226- 237.
22. Schuba, C.L., I.V. Krsul, M.G. Kuhn, E.H. Spafford, A. Sundaram and D. Zamboni, 1997. Analysis of a denial of service attack on tcp. In Proc. IEEE Symposium on Secu- rity and Privacy, Oakland, CA.
23. Snoeren, A., C. Partridge, L. Sanchez, W. Strayer, C. Jones and F. Tchakountio, Hashed-based IP traceback. In Proc. ACM SIGCOMM, San Diego, CA.
24. Kerana, Hanirex D. and K.P. Kaliyamurthie, 2013. Multi-classification approach for detecting thyroid attacks, International Journal of Pharma and Bio Sciences, 4(3): B1246-B1251.
25. Khanaa, V., K. Mohanta and T. Saravanan, 2013. Comparative study of uwb communications over fiber using direct and external modulations, Indian Journal of Science and Technology, 6(6): 4845-4847.
26. Kumar Giri, R. and M. Saikia, 2013. Multipath routing for admission control and load balancing in wireless mesh networks, International Review on Computers and Software, 8(3): 779-785.
27. Kumarave, A. and K. Rangarajan, 2013. Routing alogrithm over semi-regular tessellations, 2013 IEEE Conference on Information and Communication Technologies, ICT 2013.
28. Kumarave, A. and K. Rangarajan, 2013. Algorithm for automaton specification for exploring dynamic labyrinths, Indian Journal of Science and Technology, 6(6).
29. Shafaq Sherazi and Habib Ahmad, 2014. Volatility of Stock Market and Capital Flow Middle-East Journal of Scientific Research, 19(5): 688-692.
30. Kishwar Sultana, Najm ul Hassan Khan and Khadija Shahid, 2013. Efficient Solvent Free Synthesis and X Ray Crystal Structure of Some Cyclic Moieties Containing N-Aryl Imide and Amide,Middle-East Journal of Scientific Research, 18(4): 438-443.
31. Pattanayak, Monalisa. and P.L. Nayak, 2013. Green Synthesis of Gold Nanoparticles Using Elettaria cardamomum (ELAICHI) Aqueous Extract World Journal of Nano Science and Technology, 2(1): 01-05.

32. Chahataray, Rajashree. and P.L. Nayak, 2013. Synthesis and Characterization of Conducting Polymers Multi Walled Carbon Nanotube-Chitosan Composites Coupled with Poly (P-Aminophenol) World Journal of Nano Science and Technology, 2(1): 18-25.

33. Parida, Umesh Kumar, S.K. Biswal, P.L. Nayak and B.K. Bindhani, 2013. Gold Nano Particles for Biomedical Applications World Journal of Nano Science and Technology, 2(1): 47-57.