Volume 3, Issue 6, June 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

# A Detection of DDoS Attacks Using the Distributed Architecture of Internet Prevention Systems(IPSs)

**Ms.Leena.C[*] , Mr.Daison Raj.A, Mr.BalaAnand.M,**

*Department of CSE ,*
*V.R.S.College of Engineering. & Technology,*
*Villupuram, TamilNadu, India*

***Abstract*— The nature of the threats posed by Distributed Denial of Service (DDoS) attacks on large networks, such as the Internet, demands effective detection and response methods. The power of a DDoS attack is amplified and the problem of defense is made more complicated. The impact of DDoS attacks can vary from minor inconvenience to users of a Web site to serious financial losses for companies that rely on their online availability to do business. .In this paper,we address the problem of DDoS attacks and present the theoretical foundation, architecture, and algorithms for the distributed Architecture and,this architecture consists of the multiple internet prevention Systems(IPSs) located at the internet Service Providers(ISPs). It forms the protection rings around the hosts to defend and collaborate by exchanging the selected traffic. The evaluation of distributed architecture using extensive simulations and a real dataset is presented, showing effectiveness and low overhead, as well as its support for incremental deployment in real networks.***

***Keywords*— Collaboration, detection, distributed de- nial-of-service (DDos), flooding, network security,Distributed Architecture**

## I. INTRODUCTION

All Internet Service Providers (ISPs) face the problem of increasing amounts of unwanted traffic. Unwanted traffic is the data packets which consume limited resources like bandwidth and decrease the performance of the network, thus lowering the service quality of the network. Unwanted traffic can be produced by user misbehavior or explicit attacks like flooding-based Distributed Denial of Service (DDoS). A flooding based DDoS attack is a very common way to attack a victim machine by sending a large amount of unwanted traffic. DDoS attacks still constitute a major concern[1] even though many work shave tried to addressthis issue in the past (ref. survey in[2]). As they evolved from relatively humble megabit beginning sin2000,the largest DDoS attacks have now grown a hundredfold to break the 100 Gb/s, for which the majority of ISPs today lack an appropriate infrastructure to mitigate them [1].

Most recent works aim at countering DDoS attacks by fighting the underlying vector, which is usually the use of botnets [3]. A botnet is a large network of compromised machines (bots) controlled by one entity (the master). The master can launch synchronized attacks, such as DDoS, by sending orders to the bots via a Command & Control channel. Hence, this paper focuses exclusively on flooding DDoS attacks.1 A single intrusion prevention system (IPS) or intrusion de- tection system (IDS) can hardly detect such DDoS attacks, un- less they are located very close to the victim. However, even in that latter case, the IDS/IPS may crash because it needs to deal with an overwhelming volume of packets (some flooding attacks reach 10–100 Gb/s). In addition, allowing such huge traffic to transit through the Internet and only detect/block it at the host IDS/IPS may severely strain Internet resources. Network level congestion control can successfully throttle peak traffic to protect the whole network. However, it cannot prevent the quality of service (QoS) for legitimate traffic from going down because of attacks. DDoS is one of the major threats for the current Internet because of its ability to create a huge volume of unwanted traffic [1]. The primary goal of these attacks is to prevent access to a particular resource like a Web site [4]. The first reported large-scale DDoS attack occurred in August, 1999, against the University of Minnesota [6 ]. This attack shut down the victim’s network for more than two days. In the year 2000, a DDoS attack stopped several major commercial Web sites, including Yahoo and CNN, from performing their normal activities [5]. The detection of the DDoS attack is very hard under this situation. There is a lack of an effective differentiation mechanism that results in minimal collateral damage for legitimate traffic. The second one is that the sources of DDoS attacks are hard to be found out in a distributed network. A DDoS attack is difficult to be stopped quickly and effectively.

This paper proceeds as follows. Section II describes the architecture and the global operation of A Distributed Architecture. The different leveraged metrics and components of the system are presented in Section III. Section IV presents A Distributed Architecture attack detection algorithm. Section V presents the simulations we conducted in order to evaluate A Distributed Architecture. The complexity of A Distributed Architecture is analysed in Section VI. Section VII summarizes related work. Finally, Section VIII concludes the paper and outlines future research directions.

## II. Distributed Architecture

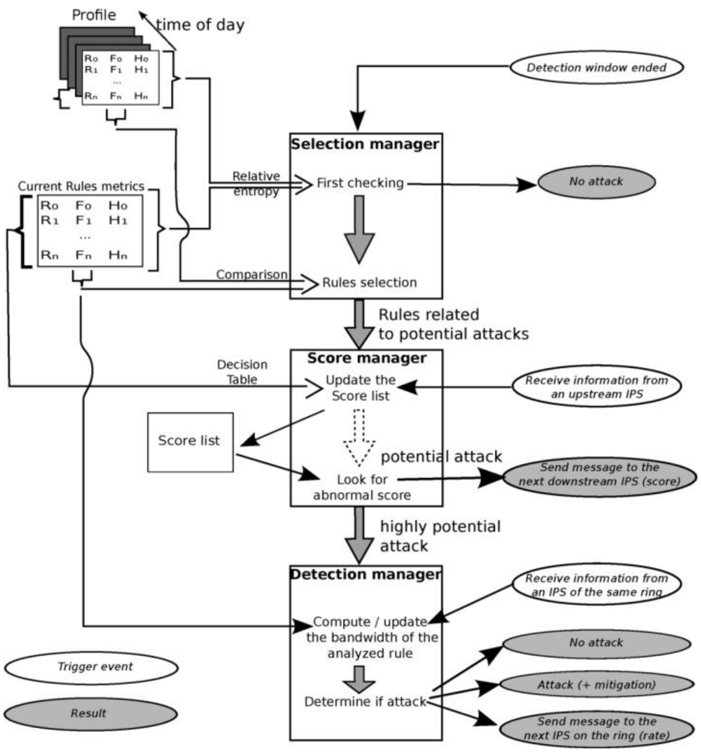


Fig.1.Distributed Architecture of Intrusion Prevention Systems

### A. Ring-Based Overlay Protection

A Distributed Architecture system(Fig.1) maintains virtual rings or shields of protection around registered customers. A Distributed Architecture IPS instance analyzes aggregated traffic within a configurable detection window. The metrics manager computes the frequencies and the entropies of each rule . A rule describes a specific traffic instance to monitor and is essentially a traffic filter, which can be based on IP addresses or ports. Following each detection window, the selection manager measures the deviation of the current traffic profile from the stored ones, selects out of profile rules, then forwards them to the score manager. Using a decision table, the score manager assigns a score to each selected rule based on the frequencies, the entropies, and the scores received from upstream IPSs (vertical collaboration/communication). Using a threshold, a quite low score is marked as a low potential attack and is communicated to the downstream IPS that will use to compute its own score. A quite high score on the other hand is marked as high potential attack and triggers ring-level (horizontal) communication in order to confirm or dismiss the attack based on the computation of the actual packet rate crossing the ring surpasses the known, or evaluated, customer capacity. As can be noticed, this detection mechanism inherently generates no false positives since each potential attack is checked. However, since the entire traffic cannot be possibly monitored, we promote the usage of multiple levels and collaborative filtering described previously for an efficient selection of rules, and so traffic, along the process. In brief, to save resources, the collaboration manager is only invoked for the few selected candidate rules based on resource-friendly metrics.

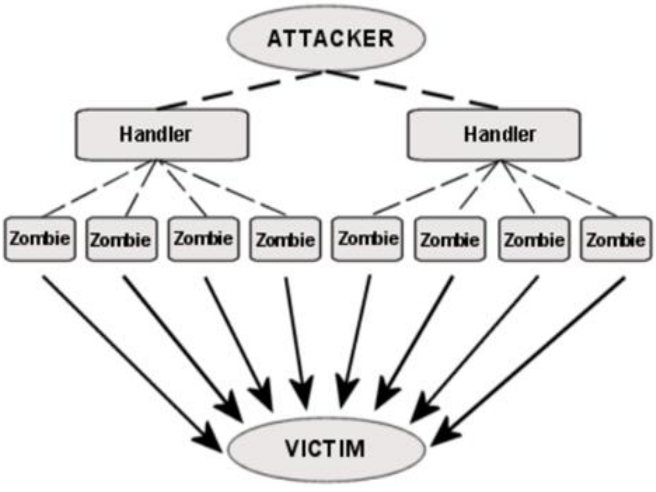


Fig.2.A Typical Architecture of DDoS Attack

### B. **Multiple Customers**

Because of their inherent complete independence, a Distributed Architecture allows the coexistence of multiple virtual protection rings for multiple customers across the same set of IPSs. Therefore, a single IPS may act at different levels with respect the customers it protects Although most of the figures in this paper represent overlay networks with a single route, from an ISP to a customer, this figure highlights that alternative paths are Possible.

However, as discussed in the previous section, the rings are dependent of the routing at a certain time, which is quite stable compared to the typical duration of flooding attacks, and soonly the current route is considered for building the rings .Fig. 2 illustrates the typical architecture of the model. One attacker sends control messages to the previously compromised agents through a number of handlers, instructing them to produce unwanted traffic and send it to the victim. The architecture of IRC-based model is not that much different than that of the agent handler model except that instead of communication between an attacker and agents based on handlers, an IRC communication channel is used to connect the attacker to agents .

## III. DISTRIBUTED ARCHITECTURE SYSTEM

### A. **Distributed Architecture Metrics**

With set of rules , A Distributed Architecture maintains the following frequency and entropy-based metrics.

1) Frequency: The frequency is the proportion of packets matching rule within a detection window

$$f_i = \frac{F_i}{\sum_{j=1}^{n} F_j} \tag{1}$$

where is the number of packets matched by rule during the detection window. Note that every customer rule set is complete, in the sense that every packet must match at least one rule. This is ensured by always having a de- fault rule matching all traffic not covered by the supplied rules.

2) Entropy: The entropy [(2)] measures the uniformity of distribution of rule frequencies. If all frequencies are equal (uniform distribution), the entropy is maximal, and the more skewed the frequencies are, the lower the entropy is

$$H = -E[\log_n f_i] = -\sum_{i=1}^{n} f_i \log_n(f_i). \tag{2}$$

.

3) Relative Entropy: The relative entropy metric [(4)](theKullback–Leiblerdistance)measuresthedissimilarity between two distributions ( and ). If the distributions are equivalent, the relative entropy is zero, and the more deviant the distributions are, the higher it become

$$\psi_i = \log \frac{f_i}{f_i'} \tag{3}$$

$$K(f, f') = \sum_{i=1}^{n} f_i \, \psi_i. \tag{4}$$

### B. **Distributed Architecture Components**

The A Distributed Architecture system is composed of several collaborating IPSs each enriched with the following components.

1) Packet Processor: The packet processor examines traffic and updates elementary metrics (counters and frequencies) whenever a rule is matched.

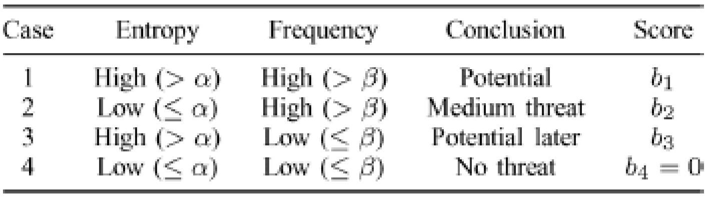
2) Metrics Manager: The metrics manager computes entropies [(2)] and relative entropies [(4)].

3) Selection Manager: The detection window ended event (Fig. 1) is processed by the selection manager, which checks whether the traffic during the elapsed detection window was within profile. It does so by checking whether the traffic distribution represented by frequencies follows the profile. This corresponds to check if [(4)], where is the current distribution of frequencies, is the stored distribution of the traffic profile, and the maximum admitted deviation from it.

4) Score Manager: The score manager assigns a score to each of the selected rules depending on their frequencies and the entropy. The entropy and the frequency are considered high if they are respectively greater than a threshold and . The different cases are presented in Table I:

A) High entropy and High rule frequency: In this case, the traffic is well distributed, meaning that most rules have about the same frequency (they cannot be all high as the

TABLE I
THE DECISION TABLE

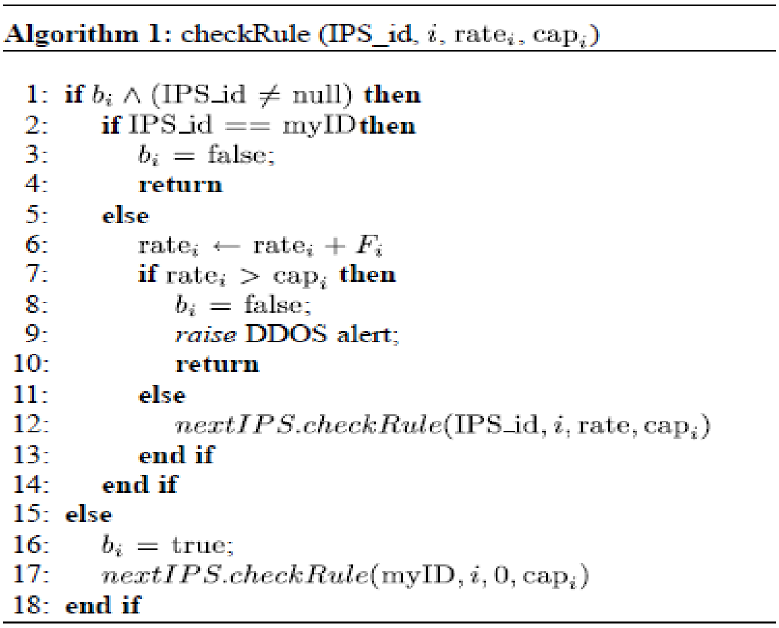


| Case | Entropy | Frequency | Conclusion | Score |
|---|---|---|---|---|
| 1 | High ($> \alpha$) | High ($> \beta$) | Potential | $b_1$ |
| 2 | Low ($\leq \alpha$) | High ($> \beta$) | Medium threat | $b_2$ |
| 3 | High ($> \alpha$) | Low ($\leq \beta$) | Potential later | $b_3$ |
| 4 | Low ($\leq \alpha$) | Low ($\leq \beta$) | No threat | $b_4 = 0$ |

sum is one). Hence, having one rule that is quite different from the others is a good sign that it is a potential attack.
B) Low entropy and High rule frequency: In this case, the at- tack is only potential, but not as much as when the entropy is high.
C)High entropy and Low rule frequency: This case represents a potential threat. Here,all frequencies are about the same, making it not a threat as the frequency is low. However, since it is increasing and deviates from the profile (first selection by the selection manager) [(5) and (6)], it may surpass other frequencies later on in time.
D) Low entropy and Low rule frequency: This case includes both high and low frequencies because of the low entropy.
E) Collaboration Manager: The collaboration manager is the last component in charge of confirming potential attacks. We claim that detecting a flooding attack can be confirmed only if the traffic it generates is higher than the customer's capacity. Hence, the IPS where the alert is triggered has to initiate a ring- level communication to calculate the average traffic throughput for subsequent comparison with the subscribers capacity.

## IV. DDoS ATTACK DETECTION ALGORITHM

For each selected , the collaboration manager computes the corresponding packet rate using rule frequencies and the overall bandwidth consumed during the last detection window. If the rate is higher than the rule capacity , an alert is raised. Otherwise, the computed rate is sent to the next IPS on the ring When an IPS receives a request to calculate the aggregate packet rate for a given rule, it first checks if it was the initiator. In this case, it deduces that the request has already made the round of the ring, and hence there is no potential attack. Otherwise, it calculates the new rate by adding in its own rate and checking if the maximum capacity is reached, in which case an alert is raised. Otherwise, the investigation is delegated to the next horizontal IPS on the ring. Algorithm 1 shows the details of this procedure. It is initially called with an empty . The first IPS fills it and sets the boolean to true (line 16). is reset after the computation finishes, i.e., when the request has made the round of the ring or when the alert is triggered. With simple adjustments, ring traversal overhead can further be reduced if several suspect rules are investigated in one pass. Rate computation can be performed based on the number of packets per second (pps) or bytes per second (bps). The first method is more suitable for detecting flooding DDoS attacks having a small packet pattern, such as SYN floods. Bytes-based method is better for detecting flooding attacks with large packet payloads. A Distributed Architecture customers can subscribe to either or both protection types.



**Algorithm 1:** checkRule (IPS_id, $i$, rate$_i$, cap$_i$)

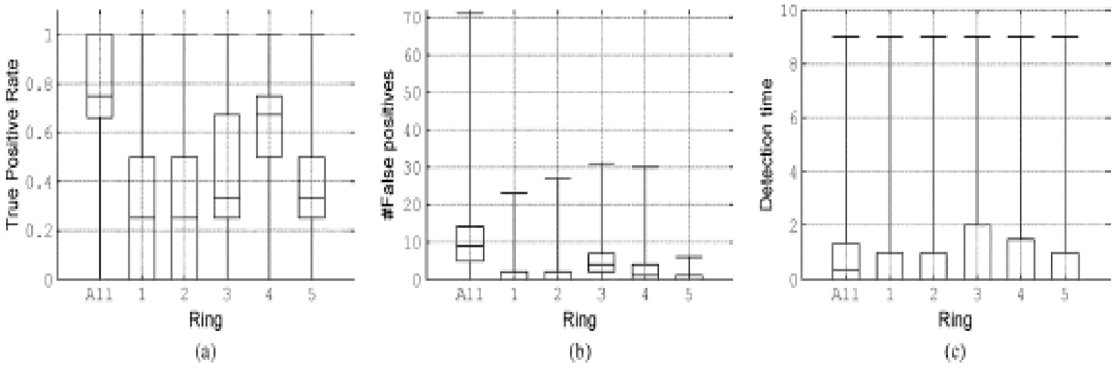
```
1:  if $b_i \wedge (\text{IPS\_id} \neq \text{null})$ then
2:      if IPS_id == myID then
3:          $b_i$ = false;
4:          return
5:      else
6:          $\text{rate}_i \leftarrow \text{rate}_i + F_i$
7:          if $\text{rate}_i > \text{cap}_i$ then
8:              $b_i$ = false;
9:              raise DDOS alert;
10:             return
11:         else
12:             nextIPS.checkRule(IPS_id, $i$, rate, $\text{cap}_i$)
13:         end if
14:     end if
15: else
16:     $b_i$ = true;
17:     nextIPS.checkRule(myID, $i$, 0, $\text{cap}_i$)
18: end if
```

## V. EVALUATION

### *A. Ring Efficiency*

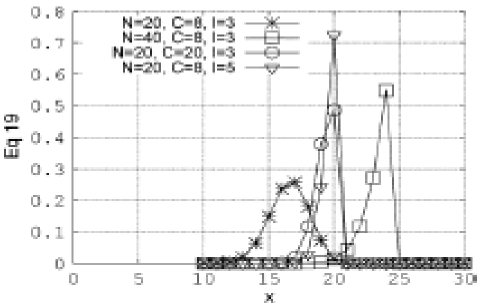
In this experiment, four attacks are generated on a five-rings topology with : two stealthy (frequency 10%) at times 40 and 50, and two aggressive (frequency 50%) at times 50 and 60. The 20th, 50th (median), and 80th percentiles and minimum and maximum values of 250 simulation runs are computed. The TPR is detailed for each ring, with the best ring being number 4 followed by ring 3 as shown in Fig. 11(a). In fact, 60% of the computed TPRs are within the 20th and 80th percentiles, which means that 60% of TPRs are between 0.5 and 0.75 for the ring 4. The fifth ring has a relatively low TPR close to 0.33 for 60% of simulations because it receives no information from upstream routers. This proves that the vertical exchange of scores between rings improves the accuracy. The TPRs of rings 1 and 2 are very low because the upper rings have already detected most attacks and hence no vertical communication is performed. A similar argument also explains why rings 1 and 2 have less false positives [Fig. 11(b)]. Fig. 11(c) shows the minimum, the 20th, 50th (median), and 80th percentile and the maximum detection delay. The median value is always 0 for all rings, and 0.33 by considering all of them. This means that the attacks are generally detected in the same window where they occur.

The detection delay is generally very low, and the worst case corresponds to the ring 3 where 80% of attacks are detected after two detection windows at most. Thus, it can also be observed that the core of the prevention system is located at rings



FI

Fig 3:Results of five rings topology with a mix of attacks. (a)Detection Accuracy.(b)False positives.(c)Detection Delay.



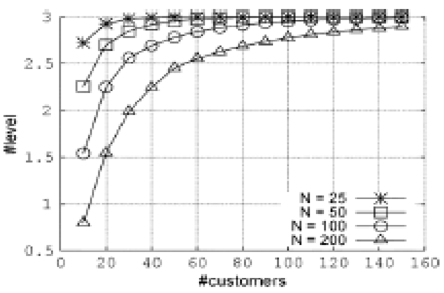


Fig 4:Probability Function

Fig 5:Average number of different ring levels for a single IPS

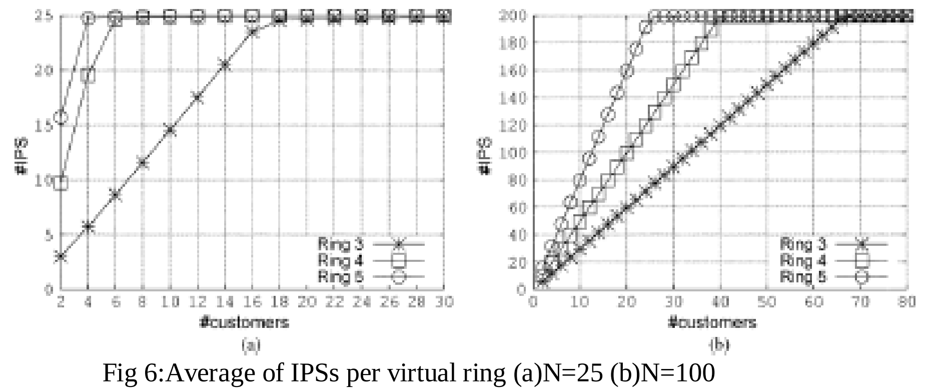


Fig 6:Average of IPSs per virtual ring (a)N=25 (b)N=100

B. ***Multiple Customers performance***

In practice, the *A Distributed Architecture* system is expected to simultaneously protect multiple customers. Assuming IPSs and customers to protect, the average number of IPSs at a certain ring level, , is computed. For this, we first compute the probability to have different IPSs at level .At level , there are at least different IPSs corresponding to the ring of a single customer [(12)]. Hence, the maximal number of IPSs for customers is (14) We then have (15) The number of IPSs at level , , is hence between and . Let denote the number of ways to define the customer–IPS relationships of the customers with at most different IPSs at level . Since for each customer, IPSs from among the are assigned, we have (16) Let be the number of ways to choose the different IPSs.

The total number of different IPSs has to be (and not ) for for (17) Therefore, the definition of for is (18) Fig. 6 plots the probability function of . When the number of customers increases, the observed peak is thinner, meaning that most IPSs act at the considered level because the load is shared. The peak highlights the most probable number of IPSs with the corresponding configuration. The same effect (for the same reasons) can be observed when the ring level increases because more IPSs are needed to provide protection to all clients. Finally, when the number of IPSs increases, the curve is shifted because more IPSs are available. Fig. 6(a) and (b) highlights the number of IPSs at a certain level with a fan-out effect of 1.5 and a three-rings configuration (from 3 to 5). Logically the curves tend to the total number of

IPSs in the system, where each IPS act atmost at each level. Moreover, the more IPSs there are, the less they participate Fig. 6. Average number of IPSs per virtual ring Fig. 6. Average number of different ring levels for a single IPS. pate into the rings because the responsibility of the protection of the different hosts is distributed among all IPSs, as illustrated in Fig. 6. This proves that the detection has to be distributed. Furthermore, Fig. 6(a) and (b) shows the worst case, i.e., the maximal number of IPSs, equivalent to having the maximal number of disjoint routes among customers. If they share more paths, the system can be better optimized by having more IPSs shared between multiple customers.

## VI. Related Work

In this chapter, we compare and contrast our work with some related work. As we mentioned before that our proposed framework has three major components, the related work are divided based on the following three issues: DDoS detection, DDoS response, and DDoS defense framework. The other detection techniques mainly include IP attributes-based DDoS detection and trace volume-based DDoS detection. Current DDoS response techniques can mainly be divided into two types: packet filtering and rate limiting. We summarize the studies of the above two types and contrast the proposed distance based Max-Min fair share rate limit algorithm with other rate limit algorithms in Defense frameworks can be categorized into three types based on the location of the defense system in the network: victim-end defense, source-end defense, and distributed defenseIn [7], the approach is based on content-filtering. [8], a peer-to-peer approach is introduced, and in [9] mobile-agents are leveraged to exchange newly detected threats.

*Distributed architecture* provides a simpler solution in the sense that it uses simple metrics, while the former approaches can be costly in terms of resource consumption. Other approaches promoting the use of simple statistics are not distributed. Reference [10] uses a packet counter per flow, while [11] proposes entropy for better expressiveness. The authors in [12] use the conditional legitimate probability to determine the deviation from a defined profile. Mahajan *et al.* introduce in [13] a technique for detecting overloaded links based on traffic aggregation. Belief functions are also used by Peng *et al.* in [14] to detect DDoS attacks based on counting new IP addresses. These works are close but differ from *Distributed architecture*, in which detection is focused on the potential victim. The authors in [15] dealt with DoS-related overload issues by a cluster architecture to analyze firewall observations. In [16], a DoS resistant communication mechanism is proposed for end-hosts by using acknowledgments. Another solution [17] relies on tokens delivered to each new TCP flow. In [18], each router between the source and the destination marks the path to detect spoofed addresses. Detection of specific SYN flooding attacks at the router level is investigated in [19]. The authors in [20] also analyzed the correlation between the requests and replies to detect flooding attacks to limit overhead. The observation of past attacks or legitimate traffic in order to create a community-of-interest is another alternative [21]. Information sharing about DDoS attacks is also addressed in [22], but from a high-level perspective where atrusted network of partners (networks) is built

## VII. CONCLUSION AND FUTURE WORKS

Belief scores are shared within a ring-based overlay network of IPSs. It is performed as close to attack sources as possible, providing a protection to subscribed customers and saving valuable network resources. Experiments showed good performance and robustness of *Distributed architecture* and highlighted good practices for its configuration. Also, the analysis of Distributed architecture demonstrated its light computational as well as communication overhead. Being offered as an added value service to customers, the accounting for Distributed architecture is therefore facilitated, which represents a good incentive for its deployment by ISPs. As a future work, we plan to extend Distributed architecture to support different IPS rule structures.

## ACKNOWLEDGMENT

I would like to express my special thanks of gratitude to my guide Mrs.J.K.Jothi Kalpana, Mr.A.Parthasarathy HOD/CSE Dept, V.R.S.CET and Mrs.H.Ambika, Mrs.K.Nirmala, Ms.N.Kavitha Assistant Professors/CSE Dept, V.R.S.CET who gave me the golden opportunity to write this important paper, which also helped me in doing a lot of Research and I came to know about so many new things, I am really thankful to them

## References

[1] A. Networks, Arbor, Lexington,MA, "Worldwide ISP security report," Tech. Rep., 2010.

[2] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems," *Comput. Surv.*, vol. 39, Apr. 2007, Article 3.

[3] E. Cooke, F. Jahanian, and D. Mcpherson, "The zombie roundup: Understanding, detecting, and disrupting botnets," in *Proc. SRUTI*, Jun.2005, pp. 39–44.

[4] CERT Coordination Center, \Denial of service attacks."Available at http://www.cert.org/tech tips/denial of service.html, March 2007.

[5] L. Garber, \Denial-of-service attacks rip the Internet." *IEEE Computer*, vol. 33, no. 4, April 2000, pp. 12{17.

[6] R. R. Talpade, G. Kim, and S. Khurana, \NOMAD: tra±c based network monitoring framework for anomaly detection," in *the Fourth IEEE Symposium on Computers and Communications*, 1999, pp. 442{451.

[7] I. Yoo and U. Ultes-Nitsche, "Adaptive detection of worms/viruses in firewalls," in *Proc. CNIS*, Dec. 2003, pp. 10–12.

[8] R. Janakiraman, M. Waldvogel, and Q. Zhang, "Indra: A peer-to-peer approach to network intrusion detection and prevention," in *Proc. IEEE WETICE*, Jun. 2003, pp. 226–231.

[6] K. Deeter, K. Singh, S. Wilson, L. Filipozzi, and S. T. Vuong, "APHIDS: A mobile agent-based programmable hybrid intrusion detection system," in *Proc. MATA*, 2004, pp. 244–253.

[9] K. Hwang, S. Tanachaiwiwat, and P. Dave, "Proactive intrusion defense against DDoS flooding attacks," in *Proc. Int. Conf. Adv. Internet, Process., Syst., Interdiscipl. Res.*, 2003 [Online]. Available: http://gridsec.usc.edu/hwang/papers/IEEES&P414Final.pdf

[10] L. Feinstein, D. Schnackenberg, R. Balupari, and D. Kindred, "Statistical approaches to DDoS attack detection and response," in *Proc. DARPA Inf. Survivability Conf. Expos.*, 2003, pp. 303–314.

[11] Y. Kim, W. C. Lau, M. C. Chuah, and H. J. Chao, "PacketScore: A statistics-based packet filtering scheme against distributed denial-of service attacks," *IEEE Trans. Depend. Secure Comput.*, vol. 3, no. 2, pp. 141–155, Apr.–Jun. 2006.

[12] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, "Controlling high bandwidth aggregates in the network," *Comput. Commun. Rev.*, vol. 32, no. 3, pp. 62–73, 2002.

[13] T. Peng, C. Leckie, and K. Ramamohanarao, "Detecting distributed denial of service attacks by sharing distributed beliefs," in *Proc. 8th ACISP*, Wollongong, Australia, Jul. 2003, pp. 214–225.

[14] M. Vallentin, R. Sommer, J. Lee, C. Leres, V. Paxson, and B. Tierney, "The NIDS cluster: Scalable, stateful network intusion detection on commodity hardware," in *Proc. 10th RAID*, Sep. 2007, pp. 107–126.

[15] G. Badishi, A. Herzberg, and I. Keidar, "Keeping denial-of-service attackers in the dark," *IEEE Trans. Depend. Secure Comput.*, vol. 4, no. 3, pp. 191–204, Jul.–Sep. 2007.

[16] H. Farhat, "Protecting TCP services from denial of service attacks," in *Proc. ACM SIGCOMM LSAD*, 2006, pp. 155–160.

[17] A. Yaar, A. Perrig, and D. Song, "SIFF: A stateless internet flow filter to mitigate DDoS flooding attacks," in *Proc. IEEE Symp. Security Privacy*, May 2004, pp. 130–143.

[18] D.Nashat, X. Jiang, andS.Horiguchi, "Router based detection for lowrate agents of ddos attack," in *Proc. HSPR*, May 2008, pp. 177–182.

[19] H. Wang, D. Zhng, and K. Shin, "Change-point monitoring for the detection of DoS attacks," *IEEE Trans. Depend. Secure Comput.*, vol. 1, no. 4, pp. 193–208, Oct.–Dec. 2004.

[20] P. Verkaik O. Spatscheck, J. Van der Merwe, and A. C. Snoeren, "Primed: Community-of-interest-based DDoS mitigation," in *Proc.ACM SIGCOMM LSAD*, 2006, pp. 147–154.

[21] G. Koutepas, F. Stamatelopoulos, and B. Maglaris, "Distributed management architecture for cooperative detection and reaction to DDoS attacks," *J. Netw. Syst. Manage.*, vol. 12, pp. 73–94, Mar. 2004.

[22] I. B.Mopari, S. G. Pukale, and M. L. Dhore, "Detection of DDoS attack and defense against IP spoofing," in *Proc. ACM ICAC3*, 2009, pp. 489–493.

[22] A. El-Atawy, E. Al-Shaer, T. Tran, and R. Boutaba, "Adaptive early packet filtering for defending firewalls against DoS attacks ," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 2437–2445.