A collaborative approach for proactive detection of distributed denial of service attacks

Jérôme François, Adel El-Atawy, Ehab Al Shaer, Raouf Boutaba

A Collaborative Approach for Proactive Detection of Distributed Denial of Service Attacks

Jérôme François *‡, Adel El-Atawy†, Ehab Al-Shaer†‡, Raouf Boutaba‡

* MADYNES - INRIA Nancy-Grand Est, CNRS, Nancy-Université, France. Email: jerome.francois@loria.fr
† School of Computer Science, DePaul University, Chicago, IL 60604, USA. Email: {aelatawy,ehab}@cs.depaul.edu
‡ David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada. Email: rboutaba@uwaterloo.ca

***Abstract*—Distributed Denial of Service attacks (DDoS) are a major threat to the Internet and detecting this kind of attacks as far as possible from the victim and close as possible to its source is a real challenge. We propose a new framework named FireCollaborator to deal with this problem on the Internet Service Provider (ISP) level, based on collaborating Intrusion Prevention Systems (IPS). A potential victim asks and pays the ISP to be protected. The key point is to use compressed metrics (*i.e.*, frequency and entropy) based on the routing rules in order to extract suspected flows. The information and alerts are shared amongst the IPSs to enhance their believes about the network status and thus to counter the attacks far away from the victim and to save the network resources.**[1]

I. Introduction

Maintaining a secure network has become a necessity for the survival of many entities that depend on their Internet presence. Protection against network attacks is a must to stay competitive in today's global market. Thus, denial of service attacks (DoS) have been considered one of the main threats against computer networks. Normally, a huge set of machines (zombies) are used to launch a distributed denial of service (DDoS) attack [1] against a certain server or set of servers. The attack, originating from different sources, is very hard to detect via any single border firewall or IDS as each device has only a local view. Besides, attackers try to generate packets that look like normal traffic. On the other hand, protecting the server at the close vicinity of its network is also inefficient because it becomes overwhelming for a single device to perform all the packets classification of the huge concentrated amount of traffic that it receives. The proposed system (FireCollaborator) is a distributed detection and alert/information sharing system that allows several Intrusion Prevention Systems (IPSs) to collaborate in order to stop distributed attacks as far as possible from the victim. Contrary to common solutions which are deployed in the lowest level in the network, an ISP level solution should save a lot of resources because the attackers' locations can be determined with high precision. Therefore the countermeasures can be defined to suit well these locations and can be placed at the most effective points of the network. We propose using the system in Tier 3 networks where the clients that ask for protection will get this optional service from their direct Internet provider, who will charge them for it. Periodic subscription requests are used to update the IPSs involved in the protection structure, and to remove the overhead of protecting machines/servers that are down/inactive. This architecture based on a payable service is a real motivation for ISPs that could be reluctant to deploy a distributed defense mechanism in the core of their networks. For the customer, it is a real gain because they are not forced to deploy and configure the security equipments on their own networks. In the following sections, we describe the subscription protocol used by clients in Section II, and an overview of the system technique in Section III. In Section IV, the main components of the system will be described. A preliminary evaluation is presented in Section V before a section about related works and the conclusion.

[1]This research has been supported under the Natural Science and Engineering Research Council of Canada grant NSERC - STP - 322235-05

II. Subscription Protocol

The ISP needs to be aware of servers and clients that subscribe to the protection service it provides. Without a subscription protocol, the ISP will loose the monetary incentive as well as force his routers to build tables containing all of his client machines. The main tasks of the protocol; 1) build up-to-date tables of subscribers at each IPS (*i.e.*, router nodes), and 2) organize these IPSs into *virtual rings* customized for each client.

Our protocol described in Fig.1 uses a trusted server S in the ISP to play the role of the token issuer. When a client subscribes (as a subnet or a single machine) for the protection service, the trusted server will add an entry showing the subscribing IP (or subnet) of the Client C along with its subscription period and the client capacity (*i.e.*, the maximum packet rate the client can handle). Periodically, C requests a new token from S. Then, S checks its tables and issue a token, signed with its private key to the client C. The token contains the client ID (*i.e.*, the IP or subnet of the client), the expiry of the token, and any optional information. C will send a multi-cast subscription message (SMsg) to all routers in the ISP asking them to update their tables with his entry (*i.e.*, all participating routers are members of this multicast group). The TTL in the periodic request message will be used to identify the ring number. In this way, routers at the same hop-distance

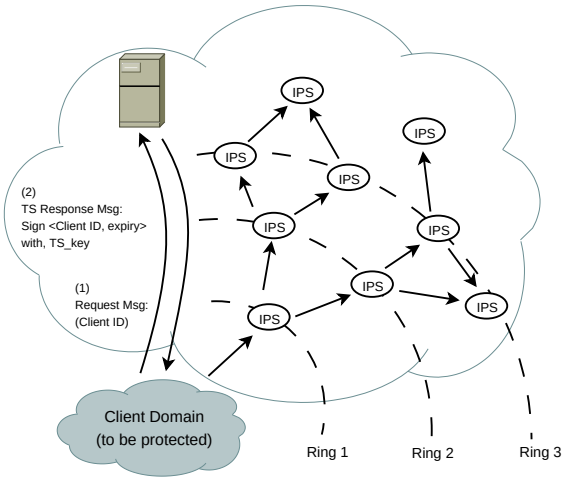


Fig. 1. The subscription protocol, showing the different communication steps needed to build the environment

from C will belong to the same ring (relative to this specific C). In response to the subscription message (SMsg), routers will send another set of multi-cast ring messages (RMsg) identifying themselves as ring members in the form of a membership phrase (*e.g.*, "IPS 10.12.76.25 belongs to ring 3 of client 10.120.87.*" = <10.12.76.25, 3, 10.120.87.*>). Also, the IPS can piggyback several membership phrases in the same RMsg to save bandwidth overhead. IPSs will use the information in RMsg to know its neighbors in a ring w.r.t. a specific C, by looking up the closest IPSs with the same ring level. Also, the IPS chooses the IPS that sent it the SMsg as the next in the path to the client (for vertical communication, as will be seen in Section IV).

III. Technique Overview

The goal is to detect the DDoS attack as far from the victim as possible. Thus the system aims to detect and counter the attacks at the Internet Service Provider Level, specifically tier three. This level is splitted into a number of rings. Intuitively, as we go closer to the source the traffic gets more concentrated. Then it is clear that it is easier to detect the attack at a low level (*i.e.*, at a ring near the victim) but the goal is to detect such flows as far as possible. Therefore, our approach uses a detection process that goes from outer to inner rings. Traditionally, an IPS has only rules (routing rules) but we assume that the IPS has - for a subscribing client - a rule that matches packets with the client's destination address. It is also possible that the client wants a defense depending on different packet properties. Therefore the rule can be defined by a kind of aggregation of common rules. The frequency of a rule is used in [2] to optimize the rule ordering of a firewall. Assume that f_i is the number of packets matching the rule R_i, the frequency of the rule is:

$$F_i = \frac{f_i}{\sum_{i=1}^{n} f_i} \quad (1)$$

The general scheme is:

Fig. 2. The different components and the detection process

- An IPS detects an abnormal value using the frequency and the entropy (details in the next section)
- The IPS cannot conclude directly and sends this information to next IPS on the path (vertical communication)
- At a lower level, an IPS receives different information and can detect that a traffic is suspect. An horizontal communication is initiated in order to determine the overall packets rate of this traffic. It is compared with the theoretic capacity of the destination host.

Generally, after the detection process, a response to counter the attack need to be provided. However, in this paper, our main goal is to provide a solution for the detection and this will be the main focus of the paper.

A. Entropy and relative entropy

The entropy for a set of rules is defined as.

$$H = -E[log f_i] = -\sum_{i=1}^{n} f(i) log_n(f_i)$$

where f_i is the frequency of the rule R_i

Entropy gives an idea of the traffic distribution. For example, if all rule frequencies of an IPS are equal (uniform distribution), the entropy is one. The more skewed the frequencies are the lower the entropy is. Relative entropy is another metric which is very useful for detecting changes in the distribution. We utilize the Kullback-Leibler distance that is defined as:

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

where $\psi_i = log\frac{f_i}{f'_i}$, f_i is the frequency of R_i in the second distribution, and f'_i is the frequency of R_i in the first distribution.

If the distributions are equivalent, the entropy is zero. The more different the distributions are the higher the value is. From an information theoretic point of view, ψ_i represents the drop in the information content of this individual flow.

IV. IPS Components

In this section and in the Figure 2, the different functionality components of each IPS are described.

A. Profile the traffic

Checking each rule for abnormalities can be a tedious task and will yield inaccurate results. A profile is very useful to detect abnormal values in rule frequencies. For each rule, the values of the frequency and the entropy are saved. It is possible to use multiple profiles: daily, weekly intervals or a simpler form by taking just the previous values.

B. Selection manager

For scalability and real time constraints, all rules of each IPS can not be analyzed. So an IPS has to select the rules to analyze. The selection is based on the attack belief. The selection manager has to determine the rules for which something abnormal is observed, and assign a higher belief for such rules. It uses two metrics; the rule frequency and its entropy. The detection begins at the end of a time window when the IPS uses the profile to determine if the current distribution is different or consistent with history. The trigger of a possible attack is the relative entropy defined in III-A. So if the relative entropy is too low no rule will be selected because the traffic have not changed. This is the "First Selection Phase" on Figure 2. The process continues to the next step if; $K(f, f') > \omega$, where f is the current distribution and f' is the profile distribution.

This first checking is very useful in order to save resources. This trigger is a global trigger and we need to know what caused these changes and whether there is a potential attack. To save resources, again, only rules with a significant frequency change are checked out. Rule R_i will be selected for investigation if and only if at time t

$$\frac{f_i(t)}{f_i(profile)} > 1 + \gamma,\ 0 \leq \gamma$$

However, rules can have very high relative increases due to an initial low frequency (*e.g.*, at the beginning of a communication). Thus, only rules with a significant frequency will be selected; $f_i(t) > \epsilon$.

C. Score manager

After the initial selection, the potential attacks have to be extracted. To have an objective view, the knowledge of the "nature" of the traffic is needed. The entropy and the frequency metrics can help in this. Each metric can be "low" or "high" depending on a threshold α for the entropy and β for the frequency.

If the entropy is high that means that the traffic is well distributed, thus there are two cases:

- high frequency (case 1): an attack is potential because the traffic is well distributed and so all rules cannot be high, and hence, having this rule being very different than others is a good sign that it is a potential attack
- low frequency (case 2): so all the frequencies are about the same and it seems that is not an actual threat but maybe later because the frequency increased (first selection)

Considering the case of low entropy, there are also two cases but the conclusions are not so evident because in this case the frequencies are quite different:

- high frequency (case 3): in this case, the frequency increases and now is high but it can be high because there are a lot of different frequencies which can be lower. In this case, the attack is potential but not as much as when the entropy is high.

γ	0.4		b_2	0.65
ω	0.05		b_3	0.8
High entropy α	0.8		$score_{factor}$	0.5
High frequency β	0.4		ϵ	0.01
b_1	1		υ	0.05

TABLE I
THE VALUES OF THE DIFFERENT PARAMETERS

- low frequency (case 4): the frequencies are very different and so probably there are high frequencies and thus the current frequency is not a direct threat.

Only the first three cases seem to be used for the detection with a different potential degree of attack, the aggressiveness can be classified as follows: case 1 $>$ case 3 $>$ case 2. Therefore, three numerical levels b_1, b_2 and b_3 differentiate case 1, case 2 and case 3 and the rule score can be computed:

$$S_i = f_i * b_j \tag{2}$$

where b_j is the corresponding case value.

Each IPS has to maintain a scores list of threats (*i.e.*, a list with tuples like $< Rule,\ Score,\ Last\_update >$). This list has to be updated at the end of the detection window:

1) All scores are reduced by a factor $score_{factor}$, say 0.5 (*i.e.*, aging process). A score is removed if it is too low (*i.e.*, below a threshold υ).
2) The score indicates a belief and the IPS has the current scores list, the new scores list (previously calculated) and a list of the scores from other IPS. By combining the believes, the new current scores can be computed thanks to the Dempster's combination rule, please read [3] for more details about it.

From Figure 2, the next step is the detection of abnormal scores. Rules will be reclassified via another threshold; τ. Rules with $Score > \tau$ are considered as very potential attacks and will be managed by the detection manager (please see the next section). Otherwise, the IPS sends the tuple $< Rule,\ Score >$ to the next downstream IPS which will apply the belief combination function with this score.

D. Detection manager

There is a final step which aims to determine the packets rate in packets per second because a suspect communication can be only due to the increasing popularity of a service. First, for each very potential attacks the IPS calculates the packets rate from the frequency of the rule and the general used bandwidth (BW). If the rate is highest than the capacity of the destination, an attack is identified. Otherwise the IPS sends a message to the next sibling which computes the used bandwith of the two IPS, checks if the used bandwidth is too high (attack detected) or continues to forward the request to the next sibling. If the request goes back to the first IPS on the ring, there is no attack.

V. EVALUATION

In this section, we discuss the effectiveness of our system. We built the system logic, and simulated it against healthy

traffic, and traffic with embedded attacks. The topology used is generated with different depths (*i.e.*, number of rings), and different connectivities among the routers. Classically there are 5 hosts to protect with 2 IPSs at the lowest level. The fanning out effect is taken in consideration (that is outer rings are larger in number than the inner rings as discussed before) with a factor of 1.5 between the number of routers of two consecutive layers.

A. Parameters

The main parameters of the simulation are defined with similar values for all routers as in Table I. These parameters were set by doing specific simulations where only one of them vary in a simulation. Then, we keep the value which allows to obtain the best results. All these simulations cannot be detailed in this paper due to the space limitation. So, fixing the right parameter needs a learning stage. Flow sizes are distributed according to a power law formula to follow the behavior of flow sizes and topology properties in the Internet [4] [5]. The main property in power law formulae is their scale invariance. As this property is preserved in exponential distribution, we can use the latter as an approximation (*i.e.*, to simplify implementation). Having limited range in the free variable ensures that the approximation is always acceptable. In other words, flows to host i have a relative size of $a \times e^{-b*i}$ with b chosen around 0.3 and a chosen to have the sum of relative sizes equal to 1. All routers are assumed to have the same capacity. The detection process is triggered every unit of time. For the simulations, a very high normal traffic is used (*i.e.*, the packet rates are close to an attack). The simulation is run over 100 units of simulation time (and detection periods) 2 kinds of attack are simulated:

- Attack 1: low attack against host 1, with a frequency less than 10% at each step, generated at the outer ring on about the half of the routers
- Attack 2: aggressive attack against host 3, with frequency greater than 30% at each step, generated at the outer ring on about the half of the routers

For the profile, because no periodic traffic is generated, an exponential moving average is used: $f_i(new\_profile) = a.f_i + f_i(profile).(1-a)$. The parameter a can be tuned and for the tests the value is 0.5.

B. Evaluation criteria

In the simulations, a router detects an attack if the score is abnormal. The last step for calculating the packets rate is not important for our simulation because the goal is to see if the selection process works well. Three different metrics are considered:

- The *detection rate* which is the number of attacks detected divided by the total number of attacks
- A false alert is defined by a router detecting an attack although if the rate is too low to be an attack. The number of *false alerts* metric is the total number of them for all routers and for all simulation steps. The false alerts rate is not useful as the detection process is always triggered at each simulation step.
- The *detection time* is the delay between the attack detection time and when the first attack packet is routed

C. Results

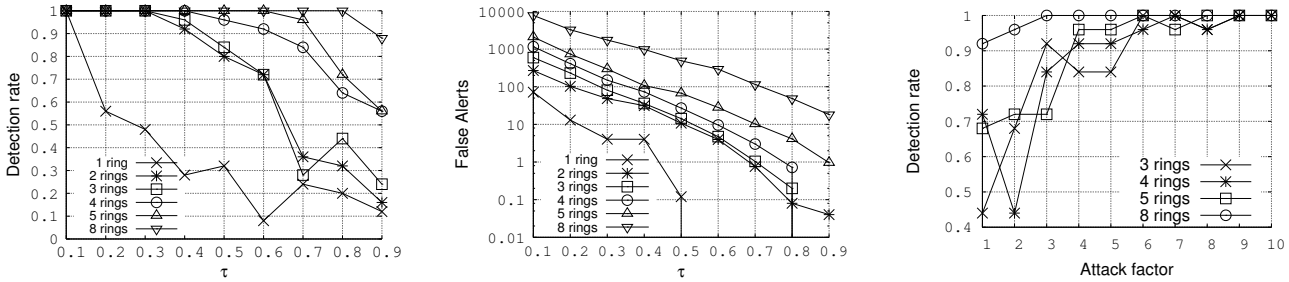
1) The score threshold parameter and the number of rings: The score threshold τ is still under investigation due to its important role. Thus, the first experiment aims to show the effect of this parameter and helps us to choose a value for other experiments. An attack of type 1 is generated at time 50 with a varying topology from 1 to 8 rings. Figure 3(a) shows the detection rate and Figure 3(b) shows the false alerts. An attack is detected if at least one router detects it and all the false alerts of the system are summed. For all experiments except if mentioned, each point in the graphs is an average of 25 independent simulation runs like. Logically, the lower the score threshold is, the higher the detection rate is.

Therefore filtering with an high threshold could discard real attacks. In the same way many false alerts are discarded and Figure 3(b) highlights the variation of the false alerts: most of the curves look to be logarithmic-linear (zero values are not displayed). For a topology of 5 rings, $\tau = 0.7$ seems to be a good choice to have a good detection rate with few false alerts. The healthy traffic can be considered as false alerts because our simulation limits the bandwith of its but not its variation. Obviously, increasing τ limits to select the healthy traffic rules and reduce the number of false alerts. The detection time is relatively low and is less than one detection window in the main cases and the highest observed value is 2.32.The very poor results for a one ring topology is a proof that the collaboration between different rings improves greatly the detection rate.

The second experiment is similar to the first one with only 3, 4, 5 or 8 rings and an attack of type 2 is triggered. This attack is more aggressive and the detection rate should be better. Indeed that is observed with a rate of 100% for all configurations with $\tau \leq 0.7$. The false alerts were counted and the graph is very close to the Figure 3(b). Thus, an attack does not affect the false alerts number. This is intuitively plausible, as false alerts are caused by cross-traffic that appears to the detection metrics to be in the form of an attack.

2) The attack aggressiveness: Different attacks are generated from the type 1 attack (low attack) multiplied by a factor (*i.e.*, rate changes to cover different aggressiveness levels). In order to see more visible variations, the score threshold is fixed to have a detection rate above 0.7 in the first simulation: 0.6 for 3 rings, 0.7 for 4 rings, 0.8 for 5 rings and 0.9 for 8 rings. Logically, the system detects more successfully the more aggressive attacks and the detection rate increases very slowly from an attack factor of 4 as can be seen in Figure 3(c) (for those score thresholds). Thus, after a certain level of aggressiveness, the system presents about the same satisfying performances.

3) The ring efficiency: Four attacks are generated in the next set of simulations. An attack of type 2 at time 40 and



(a) Score threshold parameter effect on the detection rate

(b) Score threshold parameter effect on the false alerts

(c) Attack aggressiveness effect

Fig. 3. Detection rate and false alerts

50 and an attack of type 1 at time 50 and 60 on a 5 rings topology with $\tau = 0.7$. The mean value of 250 simulation runs is computed. The detection rate is detailed for each ring and the best ring is 4 and after 3 in the Figure 4(a). The 5^{th} ring has a relatively low detection rate because it receives no information from upstream routers. Thus, the exchanges of scores between rings is a real factor to improve the detection efficiency. This communication is done only if needed and this is why the detection rate of rings 1 and 2 is very low because the previous rings have already detected the attacks and did not send the scores. As rings 1 and 2 do not often receive scores, many false alerts are not triggered like in Figure 4(b). The last one, Figure 4(c), highlights that the worst delay is observed for the ring 3 due to the same reasons.

To be brief, the core of the prevention system is located at rings 3, 4 and 5. This information is useful for a real implementation to help determining the routers which have to participate. Moreover it proves that the attack is detected early before reaching the final host, and very early into the attack duration. As the ring which detects the attacks will respond rapidly and be able to drop attack traffic early, all the downstream network overload should be quickly decreased.

4) The efficiency of our multi-level approach: The distributed approach efficiency were highlighted in the first simulations. Our framework is not only based on a scores exchanges but also on a multi-level process. First of all, the rules selection (selection manager) is very efficient because the number of false alerts is reduced by 50% or more. However the scoring process performed is also important because the number of false alerts decreases also. The variation seems to be lower (only 39% of false alerts) but an average of 49 false alerts are avoiding thanks to this stage for 5 rings.

To be brief, the multi-level approach reduces the number of false alerts. Moreover, less rules are selected and less score are exchanged. Thus the overhead of the system is greatly reduced thanks to this approach.

D. Communication and storage requirements

Regarding the previous results, an efficient system can be composed of three rings from level 3 to level 5. The simulation parameters implie respectively 4, 6 and 9 routers at respectively level 3, 4 and 5. A router at level i is connected to a router at level $i-1$ with a probability $1/i$. Therefore, the average value of needed connections is $\sum_{i=4}^{5} \frac{1}{i} \times \#routers_{level\ i} \times \#routers_{level\ i-1} = 16,8$ where $\#routers$ is the number of routers. Consequently, an attack provokes an overhead of about 17 messages in the network.

VI. Related work

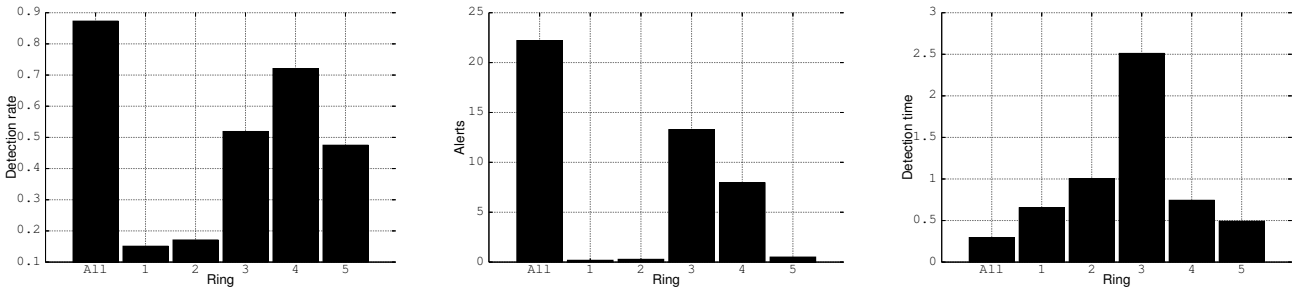
The collaboration between IPSs or firewalls is a topic for which a lot of papers can be found. In [6] the author proposes to use the distributed firewalls to counter efficiently the attacks. The author collaborates with other to propose an implementation in [7] but in fact only the rules to enforce are exchanged. So each firewall has to detect the attacks alone, the collaboration's goal is to counter the attack at each firewall to avoid traffic congestion. Our work goal is to use the collaboration to detect the attacks.

In [8] a peer-to-peer approach is introduced and in [9] a mobile-agent solution is proposed but in these 2 cases, the hosts communicate to exchange the new detected threats. Furthermore, they propose to detect another attack by using what happen on different hosts. The difference with our proposition is that the score is a compressed metric contrary to the previous solutions which need to inspect users and I/O actions.

In [10], an intelligent firewall is described in order to detect attacks proactively. The idea is to use the different characteristics of the emails to check a virus is attached. In fact it's filtering content which needs a lot of resources. So at an ISP level, it needs too much resources and it is too complicated contrary to our framework which uses only the frequency and the entropy computed easily.

Other authors propose to use simple statistics but the metrics used are not distributed over several IPS or firewalls. The authors in [11] use the conditional legitimate probability to determine the deviation toward a profile.

The paper [12] aggregates the traffic in the network in order to detect overloading links and to limit the rates for avoiding it. The aggregation can be seen as a more general rule with wildcards. The authors in [13] proposes to use belief function to detect distributed denial of service attacks too but only based on the number new source IP addresses. Our work focuses more on the potential victims *i.e.*, the mechanism



(a) Detection accuracy (b) False alerts in the system (c) Detection delay

Fig. 4. Results of a 5 rings topology with a mix of attacks

observes the traffic to a certain destination according to a specific rule. Thus it is a more practicable business solution as an additional service offered by ISP to their clients who are charged for this protection.

VII. Conclusion and future works

In this paper, we propose a new framework to improve the security at a higher network level that common solutions which are applied at low levels in the networks because they are not fitted for the ISPs level or because the ISPs have no benefit to deploy them and prefer to leave the DDoS problems to their clients. The key point is that IPSs can communicate to each other in order to exchange valuable information but not too much in order to avoid an attack whose the goal would be to overload this system (denial of service on the security infrastructure). Thus, we propose a system where clients subscribe to the distributed protection service with minimal communication overhead while avoiding the service to be used by unsubscribing clients. IPSs in the Tier 3 network will be arranged in virtual rings around each client. Each IPS is responsible for selecting the rules for which an attack is very potential. The collaboration takes place through vertical communication to help to make this choice and through horizontal communication to get a complete view of the traffic going into the client. The alert information technique guarantees the absence of false positives, using the overall rate compared with the host capacity as a sure sign of the existence of an attack. The simulation proves that a good accuracy can be obtained and that our framework presents a real benefit because the multi-level process to detect attacks filter a lot of packets and extract only interesting rules. Thus the number of false alerts is decreased and the IPS can inspect in more details the suspect rules. Due to these facts we are convinced that our framework is scalable and can provide a real alternative to classical IPSs or other security equipments by distributing the detection.

The plan for the future is to use real traces to test our solutions and deal with other problems like; security for the IPS communication, the effects of compromised IPSs or a limited number of participating routers.

References

[1] E. Cooke, F. Jahanian, and D. Mcpherson, "The zombie roundup: Understanding, detecting, and disrupting botnets," in *Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI)*, June 2005, pp. 39–44.

[2] H. Hamed and E. Al-Shaer, "Dynamic rule-ordering optimization for high-speed firewall filtering," in *ASIACCS '06: Proceedings of the 2006 ACM Symposium on Information, computer and communications security*. New York, NY, USA: ACM Press, 2006, pp. 332–342.

[3] A. Jsang, S. Pope, J. Diaz, and B. Bouchon-Meunier, "Dempster's rule as seen by little coloured balls," *Information Fusion Journal*, 2005.

[4] N. Brownlee and K. Claffy, "Understanding internet traffic streams: Dragonflies and tortoises," *IEEE Communications Magazine*, vol. 40, no. 10, pp. 110–117, 2002.

[5] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," in *SIGCOMM*, 1999, pp. 251–262.

[6] S. M. Bellovin, "Distributed firewalls," *;login:*, vol. 24, no. Security, November 1999.

[7] S. Ioannidis, A. D. Keromytis, S. M. Bellovin, and J. M. Smith, "Implementing a distributed firewall," in *CCS '00: Proceedings of the 7th ACM conference on Computer and communications security*. New York, NY, USA: ACM Press, 2000, pp. 190–199.

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

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

[10] U. Ultes-Nitsche and I. Yoo, "Steps toward and intelligent firewall - a basic model." Proc. Conference on Information Security for South Africa (ISSA2003), July 2003, pp. 9–11.

[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 Transactions on Dependable and Secure Computing*, vol. 3, no. 2, pp. 141–155, 2006.

[12] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, "Controlling high bandwidth aggregates in the network," *SIGCOMM 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 *Proceedings of 8th Australasian Conference on Information Security and Privacy (ACISP 2003)*, Wollongong, Australia, July 2003, pp. 214–225.