A Multi-Scale Tomographic Algorithm for Detecting and Classifying Traffic Anomalies

Sílvia Farraposo, Philippe Owezarski, and Edmundo Monteiro

Abstract—**The occurrence of a traffic anomaly is always responsible for a degradation of performance. The anomaly can be observable, at some scale, in different ways: an increase in the number of packets, an increase in the number of bytes, a concentration of packets around a port number, etc.**

In this paper we propose an anomaly independent methodology for detecting such traffic anomalies and to classify them. To accomplish that, we integrate previous work in a multi-criteria tomographic analysis process, criteria being bytes, packets or flow rate, port number or address distribution, etc. As a demarcation from this inspiring work, this new methodology is based on a multi-scale analysis, which always permits the exhibition of anomalies on at least one parameter at one time scale. The motivation for using simple parameters deals with making the interpretation of anomalies simpler, and mitigation mechanisms obvious. In addition, this methodology associates to each anomaly a set of parameters that is able to characterize the anomaly and will serve as a signature for it.

This paper presents this methodology, the related algorithm for anomaly detection, and its application on several real traffic traces captured on several networks: Auckland university, GEANT and Renater.

Index Terms—**Measurement, Traffic Analysis, Anomaly Detection.**

I. INTRODUCTION

TRAFFIC anomalies can be described as the result of one or more occurrences that change the normal flowing of data over a network. Such occurrences can be triggered by a diversity of behaviors, as Denial of Service (DoS) or Distributed Denial of Service (DDoS) attacks, flashcrowds or management operations. Because traffic anomalies might occur at any point of the Internet, have unpredictable behaviors and can range from a single network failure to a complex security attack, being orchestrated through a thousand of separate machines, stopping these anomalies is something that is very difficult to accomplish. So, most of the efforts are engaged in the detection of these anomalies as soon as possible, in such a way that legitimate actions can be taken to limit the extent and harshness of these anomalies.

Nowadays, several methods exist to detect and characterize anomalies. Some approaches are based on simple statistics calculated on some traffic parameters such as the number of UDP packets or the number of SYN packets. Then, when calculated values are above a given threshold, an attack might be signaled, such as an UDP or TCP SYN flood. Much work like that can be seen in [1–5]. Given the variability of the traffic, and the number of false positive of such methods (because thresholds are difficult to fix), more recent work has introduced more complex statistical analysis based on the spectral density of the signal associated to the traffic, its correlation, etc. (e.g. [6-15]). It is then possible to issue signatures for different kinds of anomalies. But the signatures do not indicate which the basic characteristics of the anomalies are: what packets constitute the anomaly, where are they coming from, etc. and are then hardly usable for network or security managers.

However, since anomalies are constantly changing and at some extent being more difficult to detect, some previous techniques to detect anomalies are limited and unable to detect new kinds of anomalies. Our major contribution, with this work, is to present an algorithm that does not have these constraints – i.e., that is not restricted to a set of anomalies, but is able to detect and characterize new anomalies. In addition, one of our main goals was also to develop an algorithm working on understandable for computer networks values, for which an anomaly analysis will indicate precisely what the problem is.

Even if on a first sight, this work can appear as similar as many others, we introduce a multi-scale algorithm and use a tomography like concept. The multi-scale feature assures that any anomaly is detectable independently of its duration. For example, a flash crowd is only visible after a certain amount of time, because of what it is better detected when using a large time scale. This multi-scale analysis provides some elements giving richer signatures for the different anomalies that were encountered during this work. The tomography like concept, as its medical counterpart, is intended to allow the observation of anomalies at different perspectives.

Correlating the information collected at each step of the algorithm, at different time scales and different levels of aggregation, permits the identification of anomalies and the definition of a set of characteristics associated, that can be

This work has been partly supported by the European Union under the E-NEXT NoE FP6-506869. It also has the support of the French MetroSec project granted by the ACI on Security and Computer Science

S. Farraposo is a PhD student with LAAS and DEI/CISUC – University of Coimbra. She is also teacher at Polytechnic Institute of Leiria – Portugal. (phone: +351-964-263-700; e-mail: silvia@estg.ipleiria.pt).

P. Owezarski is at LAAS – CNRS, 7 Avenue du Colonel Roche, 31077 Cedex 4 Toulouse, France (e-mail: owe@laas.fr).

E. Monteiro is at DEI/CISUC, University of Coimbra, Pólo II Pinhal de Marrocos, 3030-290 Coimbra, Portugal (e-mail: edmundo@dei.uc.pt).

used to define a database of anomaly signatures.

The paper is organized as follows. Section 2 describes the multi-criteria tomographic algorithm. It will insist on the parameters considered by our algorithm (packets, bytes, starting flows, time-scale and level of aggregation). It also gets into details for the 3 different stages and related principles of this algorithm. Section 3 presents some results of the application of our algorithm on real traffic traces captured on different points of the Internet. Then, based on the analysis of the anomalies that were detected in the considered traces, we explain through a set of examples, how the multi-scale aspect of our analysis helps to improve anomalies signature and their classification. Finally, Section 4 concludes this paper, by presenting some possible applications for our anomaly detection approach.

II. Description of the Algorithm

The main goal of this algorithm is to detect and identify traffic anomalies and the flows responsible for those anomalies, i.e., the IP addresses associated (source and destination).

To reach our goal, we use the IP address tomography concept, through which we split the IP address space recursively to obtain specific anomalous IP addresses. This operation is performed at different time scales, i.e. with different levels of aggregation of the traffic: traffic can be aggregated on few ms, few seconds, up to several minutes and more. The algorithm can be divided in 3 phases:

1) Detection of time intervals with anomalies.
2) Identification of the faulty IP addresses.
3) Identification and characterization of anomalies.

A. Detection of Time Intervals with Anomalies

For assessing this algorithm, we applied it on already captured traffic traces, in an off-line way. Obviously, the algorithm, with slight modifications can be run on-line on a living link. Running on-line or off-line, the algorithm has to compute large amounts of data. This size issue may constitute a performance drawback, since it can increase significantly the algorithm's running time. To overcome this, part one of our algorithm permits the restriction of the search space by spotting small time intervals were anomalies are occurring.

For example, given a trace of duration T, the result of this process is a set of N slots where traffic anomalies were detected, and N ∈ [0, T/Δ], and Δ is the time-scale granularity.

The basis to detect an anomaly is that an anomaly is meaningful only when it is responsible for a significant variation in the number of packets, bytes or flows (or altogether). In our algorithm an anomaly is detected through the application of an appropriately formula, which detects significant variations between two consecutive time intervals in the criterions being analyzed.

$$X = \{x_1, x_2, ..., x_n\}, x_i = \{\# \, packets | \# bytes | \# \, flows\} / \Delta$$

$$P = \{p_1, p_2, ..., p_{n-1}\}, p_i = x_{i+1} - x_i$$

$$\begin{cases} pi \geq E(p) + k\sigma, select \\ pi < E(p) + k\sigma, reject \end{cases}$$

In our formula, X is a data series with the number of packets, bytes and flows, per unit of time, extracted from the traffic trace under study, and P is a data series obtained from X, in which each value is the variation between the number of {Packets ∨ Bytes ∨ Flows} in two consecutive time slots. Then, the mean value E(p) of each data series (PB, PP, PF) is calculated, as well as its standard deviation, σ.

The application of the formula states that an anomaly is occurring if the value of p_i (i.e., the variation of the number of {Packets ∨ Bytes ∨ Flows} between two consecutive intervals) exceeds a given threshold. Each threshold has as value E(p) + kσ, where factor k in our formula, permits the detection to be coarser or finer, using smaller or bigger values of k, respectively. Because each p_i is located in well known timeslots of duration Δ, the application of the formula permits the temporal localization of the anomaly, and like this to narrow the search.

The formula's application intends to detect significant variations over the data series, which explains the use of data series P instead of X. Like this we can be aware of the variability of the amplitude of the curve, and not the variability along time, which is meaningless in this case.

B. Identification of Anomalous Addresses

The result of the previous part is the localization in time of the most significant amplitude variations in the data series, and an indication of which parameters were affected by the anomaly. The second phase of this process is intended to identify the flows responsible for each anomaly detected. For this work, we have been inspired by the flow definition presented by Claffy *et al.* [16] that states that a flow is a set of packets moving from one source to a destination point, and that is identified through a five-tuple masking (Source Address, Destination Address, Protocol, Source Port, Destination Port) and a timeout value. The algorithm presented in this paper has a broader definition and considers a flow as a sequence of packets from any source to a destination identified by the tuple (IP network, mask), and a timeout limit of 64 seconds to the inter-arrival time between two packets of a same flow. Nonetheless, if our flow definition is not considering all the parameters of the Claffy et al.'s five-tuple, it is prepared to include them.

To pop up anomalous flows, we have exploited a tomography based structure. Like medical tomography, in our algorithm we are opening windows in the network to explore what is going wrong, and taking slices snapshots to extract accurate information. While the window opening was of the responsibility of phase one, this part of the algorithm exploits the plan division part, which is in this case the division of the all-IP address space in several plans, each corresponding to a collection of flows.

To apply the tomography-like concept to our algorithm we have considered as starting point, that the all-IP address space is represented by network 0.0.0.0 and mask 0.0.0.0, and that each new space is obtained through the separation of packets, meaning an increase of the mask. This approach permits to screen all the IP space looking for anomalous flows. Anomalous flows are spotted at each level of aggregation, which permits us to see its "evolution" as the level of aggregation decreases. To accomplish this, at each level of aggregation the formula is applied, and all the anomalies are located. Since, our objective is to track anomalies at certain time slots, those obtained in phase one, only those time slots are analyzed. The visibility or not of an anomaly at a certain level of aggregation tells, for example, on the size of the flows involved in the anomaly. If the flow only appears in high aggregation levels, and disappears at /24, most probably, the anomaly was due to a collection of small flows, that when disaggregated is not detectable.

Conceptually, the tomography approach described above is able to screen all the IP space address from its root (IP 0.0.0.0) to all its leaves, and to extract the destination IP addresses associated with the anomalies. However, due to performance constraints, and because at some cases it was meaningless (for example, looking for aggregates /24 or higher in a core trace) the tomographic analysis was not implemented as we have presented it. So, some simplifications were assumed, like to screen only some levels of aggregation, as /8, /16, /24 and /32, and previously select the addresses to scan.

C. *Identification and Characterization of Faulty Anomalies*

At this point of the algorithm, anomalies were located in time, and associated destination IP addresses identified. However, even if at this point it is possible to have a clue about the type of anomaly, none identifies it clearly and defines its signature. These are the main goals of phase three.

The characterization process of anomalies is accomplished by relating the information presented in Table 1. To characterize anomalies, we use a distribution function of the volume parameters (packets, bytes and flows) versus the IP feature parameters (IP address and port). The resulting relationship permits the observation of how each volume parameter is affected by each anomalous flow.

TABLE I
DESCRIPTION OF PARAMETERS TO CHARACTERIZE ANOMALIES

Parameter	Description
Number of packets, bytes, new flows	Volume information associated to anomalous flows.
Source and destination IP addresses	List of source and destination IP addresses involved with an anomaly.
Source and destination ports	List of source and destination IP ports involved with an anomaly.

To identify and characterize correctly an anomaly, the parameters presented in Table 1 must be analyzed in a multi-scale perspective and at different levels of IP addresses aggregation, which is straightforward since all the information required can be collected by our algorithm.

Because of the time-scaled decomposition we are able to detect changes in network behavior that may appear at some time resolution but go un-noticed at others. This aspect has revealed to be important when detecting some types of anomalies, such as port scans, that do not appear at some time scales, or some weak DDoS attacks.

Besides time-scale variation, another variable that revealed to be important in our identification/characterization process is the level of aggregation of IP addresses being studied. Some of the volume parameters, such as the number of flows, are particularly sensitive to the level of IP addresses aggregation. This behavior is one additional variable to take into account when defining anomaly profiles.

III. RESULTS

Effective detection and identification of anomalies in traffic requires the ability to separate them from regular network traffic. In this Section we start by presenting data traffic over which our study was conduced, and show step by step how to use our approach to detect and identify anomalies.

A. *Description of Data*

Besides developing an approach able of detecting and characterizing anomalies, which is possible with several other approaches as the ones presented in [10, 13, 15], our intention with this algorithm is also to define a database with traffic anomaly signatures, to be used with applications "interested" in network traffic anomalies. Because of that, it is our intention to test this approach on as many traffic traces as possible. For this work, we have used traces captured on three different environments: Auckland 8, Renater, and GEANT.

The Auckland-8 data set is a two weeks GPS-synchronized IP header trace captured with an Endace DAG3.5E tap Ethernet network measurement card in December 2003 by NLANR [18]. Capture was made at the Internet link access of the university campus. All traces collected were anonymized, however preserving addresses structure.

The called Renater data-set was obtained in the context of the MetroSec project [19], a French project granted and funded by the French ministry of research. The MetroSec project intends, among other goals, to analyze collected traces and study the nature and impact of anomalies on QoS. Several French institutions work on the project, and maintain a database with collected traces. The traces available in the project were captured last year and this year, have durations ranging from some minutes to a few days, and include traffic anomalies under study, namely DDoS and flash crowds. As the Auckland 8 captures, these traces were obtained with a DAG card, at the Internet access link of LAAS.

The GEANT [20] network interconnects the European research and educational networks. Particularly, the GEANT trace available for this work, was captured in 2005 during 4 months (June to August), and at 23 PoPs distributed in Europe. Packets were captured with NetFlow, aggregated into flows at the network prefix level, and reported in 10 minutes bins. Because of that, all data necessary to use the approach is not available, and usage of GEANT trace was conditioned.

Despite of some restrictions, the diversity of traces available permits some accuracy on results.

B. *Diagnosing Anomalies*

Without any anomaly diagnosis tool, the most popular procedure to detect any misbehavior in network traffic is visual inspection to some graphs, looking for significant variations in the number of bytes, packets and flows, during the period under analysis – which would, quite probably, reveal some anomalies. But are these volume increases anomalies, or just a small change in traffic due to a new flow, to a more intensive download? Answering these uncertainties is undoubtedly one main concern of all traffic anomaly detection approaches.

1) *Detection of an UDP Flood*

Packet flooding is probably one of the most common types of denial of service attacks, because they are easy to perpetrate and quite effective. These ones are characterized as brute force attacks, where a significant amount of packets (ICMP, UDP or TCP) is sent from one or more sources to a restricted set of destination addresses (most of the times one destination address). The presence of this type of anomaly is signaled by our algorithm, at all time scales, by a persistent increase in the number of bytes and packets being sent to a specific destination address, as can be seen in Fig. 1 (because plots are very similar and due to size limitations, only plots related with packets are presented). All the graphs from the figure result from the application of steps one and two of our algorithm in a multi-scale perspective – first the identification of the anomalous time intervals, and then the identification of the anomalous flows. In this particular case, the anomaly was detected at all time scales being considered, which is not always the case. Moreover, the detection of an anomaly at several time scales depends on the type of anomaly itself.

The observation of Fig. 1 only shows the occurrence of one high frequency point, but do not give any information about the type of anomaly that is occurring. Depending on what we intend to do with the information being obtained, this level of analysis could be enough. For example, if we were detecting anomalies just with routing purposes in mind, the knowledge of the destination would be enough to act on flows correspondingly – for instance, allocate more bandwidth for that flow.

However, defining the correct action over anomalous traffic is not simple. As presented before, an anomaly can range from an elephant flow (which could be admissible) to a DoS attack or flash crowd event. Should we act on the same way over traffic?

The answer is no, and this is why the identification of the anomaly is necessary, and motivates part three of our algorithm. To accomplish our goal, at this point we introduce port information and source information. The consideration of these two IP features is important because they represent another vector of analysis: the level of traffic aggregation. Fig. 1 was obtained considering flows at level of aggregation /8. When considering higher values, for example /24 or /32, traffic flows are being disaggregated, and if the anomaly is not persistent, it will smooth and not being detected by our formula – this aspect will permit to infer about the harmfulness of the anomaly.

Particularly, for this example, we have identified one common flow, 140.0.0.0, with a high frequency peak, at port 2744. So, the application of part three of our algorithm involves the decomposition of that flow, in smaller flows, and to verify if peaks are still present, and which source/destination addresses are involved. Fig. 2 shows how packets are distributed per flow, when considering a level of aggregation /32. As we can see, the anomaly is persistent, since a high frequency point still persists on all three plots. More particularly, the flow responsible for this anomaly was generated at source 132.227.72.202, and has as destination the address 140.93.192.71 and port 27444, which is Trinoo slave port, an UDP flood attack tool.

2) *Flash Crowd Event*

As with denial of service attacks, the occurrence of flash crowd events slows down significantly the flowing of traffic. However, these events cannot be considered as illegitimate, since usually users are performing authorized actions, and a set of characteristics must be found to differentiate both types of traffic.

When using our algorithm to study traces with flash crowd events, high frequency points are detected. Contrary, of what happened with the UDP flood attack, the flash crowd event is responsible for more than one high frequency point, each associated at one of the parties of the flash crowd event (one server, and several clients for HTTP flash crowds), as it is showed in Fig. 3, which presents a distribution of the number of packets per destination address (a flow is plotted only if it was significantly variable).

Fig. 4 presents a zoom of what happens at the HTTP server side, showing the different sources that are sending traffic. As expected, several high frequency points, from different source addresses (clients trying to access the server) are visible, which denotes a bi-directional flowing of data. Moreover, an analysis of the files obtained by our algorithm shows a similar behavior in the reverse order, i.e., a high frequency point in the number of packets/bytes exchanged from the server (source) to the client (destination).

3) *Network Scan Anomaly*

Network scanning is a procedure for identifying active hosts on a network, either for the purpose of attacking them or for network security assessment. Usually, a network scan is identified when a source attempts consecutively to scan a restricted set of ports, at different destination addresses. Using our algorithm such kind of anomaly is detected if such procedure is responsible for a significant increase in the number of packets, bytes or new flows, at one or more time/aggregation scale.

This was not the case for the trace used to present the flash crowd event. However, when analyzing the rightmost plot of Fig. 4, some lines of the plot are quite interesting, suggesting a network scan procedure (lines perpendicular to the x-axis). Two of these lines are x-axis 2186 and 2708, which for a fixed source address suggest the scanning of several destination IP addresses. An analysis of the files obtained to plot the graph showed up the occurrence of a Trojan Horse named Backdoor.CrashCool that allows unauthorized access to the victim machine.

The occurrence of the network scan anomaly showed that in some cases the graphs obtained by our algorithm might also be used to exploit some types of anomalies, that otherwise would be unnoticed, because they are not responsible for significant variations in one of our multi-criteria parameters.

IV. CONCLUSION

In this paper we have proposed a three step algorithm to detect and characterize traffic anomalies. To perform that, the approach works at three different axes: the multi-scale axis, the multi-criteria axis and the multi-IP space axis, each of them responsible for inputs that are related to obtain relevant information. Particularly important to this approach are the multi-scale axis, which permits detection of time-scale dependent anomalies, and the multi IP-space axis, which using a tomography-like approach permits efficiently looking for anomalous flows in all IP address space.

The validity of our approach was tested over several traffic traces, and particularly using a Renater trace, we presented a step by step application of the algorithm. Besides detecting the anomalies, our approach permitted the definition of anomaly signatures which can be used as input to other domains. More trace analysis still need to be run for completing our current anomaly signatures database.

A value added of this approach, when compared with others, is the simplicity of its detection method which does not use complex statistical methods, and is still being efficient. Then, it is easy for a network administrator to understand what is going wrong: the understandable information is directly provided. With methods working in the frequency space, for instance, there is a long way to come back from the frequencies to understandable for the administrator bytes, packets and flows.

Looking further, when we will have analyzed many more traces containing examples of any kinds of anomalies for completing our database, we intend to use this algorithm for two different applications (among many other which could benefit from such algorithm – see the applications mentioned in the related literature). The first one deals with improving traffic engineering. If an anomaly arises, and if it is classified by our algorithm as legitimate (as a flash crowd for instance), we will get some important information for a fruitful change of the routes or of the load balancing strategies between routes. In addition, given the latencies for route changes at the scale of an AS (Autonomous System), the performances of the current algorithm are sufficient.

The second application we have in mind is an IPS (Intrusion Prevention System) when our algorithm detects an illegitimate anomaly. In that case, phases 2 and 3 provide us enough information for pointing out specific flows or packets. We then just need discarding those flows or packets. Nevertheless, such IPS has to react in short times (of the order of one minute maximum). The current performance of our algorithm does not permit such speed for analyzing anomalies. We then have to gain a speed factor of around 5 to fit the IPS requirements. This is one of the main objectives of our work in the following weeks.

REFERENCES

[1] J. Brutlag, Aberrant behavior detection and control in time series for network monitoring. In Proceedings of 14th Systems Administration Conference (LISA 2000), (New Orleans, LA, USA), USENIX, December 2000.

[2] D. Denning. An intrusion-detection model. IEEE Transactions on Software Engineering, February 1987.

[3] F. Feather, D. Siewiorek, R. Maxion. Fault Detection in an Ethernet Network using anomaly signature matching. In Proceedings of ACM SIGCOMM, 1993.

[4] M. Thottan and C. Ji. Anomaly Detection in IP Networks. In IEEE Transactions on Signal Processing, Vol. 51, no. 8, August 2003.

[5] V. Yegneswaran, P. Barford, and J. Ullrich. Internet Intrusions: Global Characteristics and Prevalence. In Proceedings of SIGMETRICS'03, (San Diego – California, USA), June 2003.

[6] M. Baldi, E. Barladis, and F. Risso. Data mining techniques for effective and scalable traffic analysis. In IM'05, 2005.

[7] P. Barford, J. Kline, D. Plonka, and A. Ron. A signal analysis of network traffic anomalies. In Proceedings of ACM SIGCOMM Internet Measurement Workshop, (Marseilles, France), Nov. 2002.

[8] C. Cheng, H. Kung, and K. Tan. Use of spectral analysis in defense against DoS attacks. In Proceedings of IEEE Globecom 2002, 2002.

[9] A. Lakhina, M. Crovella, and C. Diot. Diagnosing network-wide traffic anomalies. In Proceedings of SIGCOMM'04, (Portland, Oregon, USA), September 2004.

[10] A. Lakhina, M. Crovella, and C. Diot. Mining Anomalies Using Traffic Feature Distributions. In Proceedings of SIGCOMM'05, (Philadelphia, Pennsylvania, USA), August 2005.

[11] T. Gil and M. Poletto. MULTOPS: a data-structure for bandwidth attack detection. In USENIX 2001, (Boston. Massachusetts, USA), June 2001.

[12] U. Hengartner, S. Moon, R. Mortier, and C. Diot. Detection and analysis of routing loops in packets traces. In Proceedings of IMW'02, (Marseille, France), 2002.

[13] M. Roughan, T. Griffin, Z. Mao, A. Greenberg, and B. Freeman. IP Forwarding Anomalies and Improving their Detection Using Multiple Data Sources. In Proceedings of SIGCOMM'04 Workshops, (Portland, Oregon, USA), August 2004.

[14] S. Kim and A. L. N. Reddy. A Study of Analyzing Network Traffic as Images in Real-Time. In IEEE INFOCOM'05, (Florida, USA), 2005.

[15] S. Kim, A. L. N. Reddy, and M. Vannucci. Detecting traffic anomalies through aggregate analysis of packet header data. In Networking' 04, 2004.

[16] K. Claffy, H. Braun, and G. Polyzos. A parameterizable methodology for Internet traffic flow profiling. In Selected Areas in Communications, IEEE Journal, vol. 13, p 1481-1494, October 1985.

[17] DAG. At http://www.endace.com/products.htm.

[18] NLANR. At http://pma.nlanr.net/Special/auck8.html.

[19] MetroSec project. At http://www2.laas.fr/METROSEC/.

[20] GEANT Project. At http://www.geant.net/.

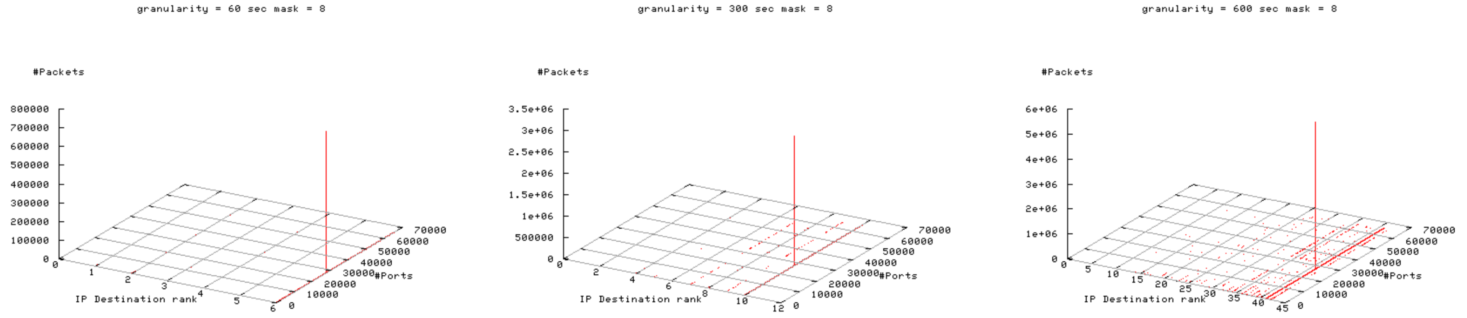

Fig. 1. Distribution of the number of packets received per destination IP address at one particular port, with a level of aggregation /8. Each IP destination address presented is associated to an anomalous flow. From left to right, each plot shows an observation at a different time granularity: 60, 300 and 600 seconds, respectively.

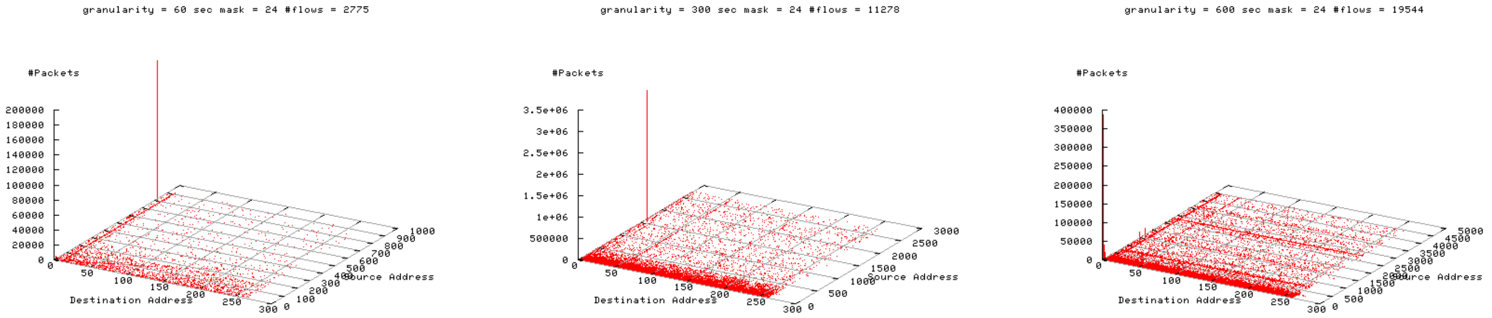

Fig. 2. Distribution of the number of packets sent from a source to a destination address. Each plot is a zoom of the high frequency peak observed on Fig. 1. Zooming was obtained considering a level of aggregation /32.

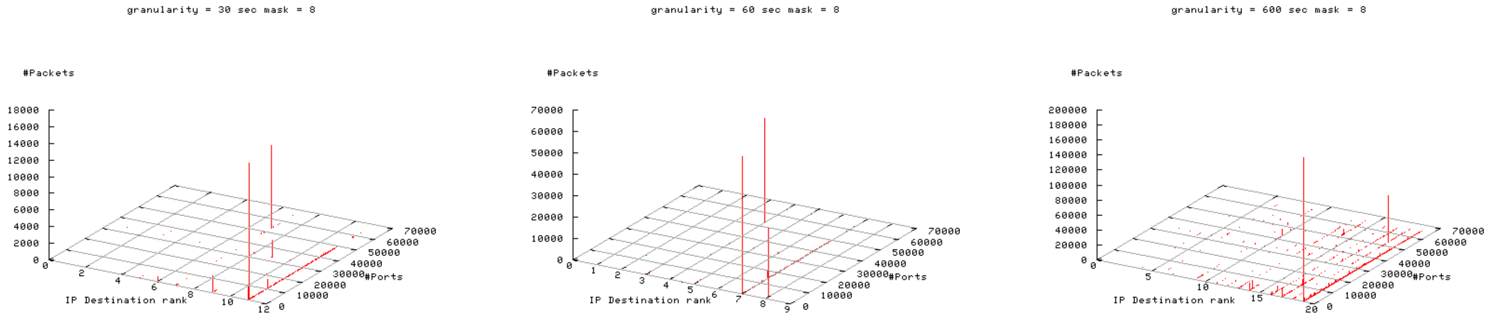

Fig. 3. Distribution of the number of packets received per destination IP address at one particular port, with a level of aggregation /8. Each IP destination address presented is associated to an anomalous flow. From left to right, each plot shows an observation at a different time granularity: 30, 60 and 600 seconds, respectively.

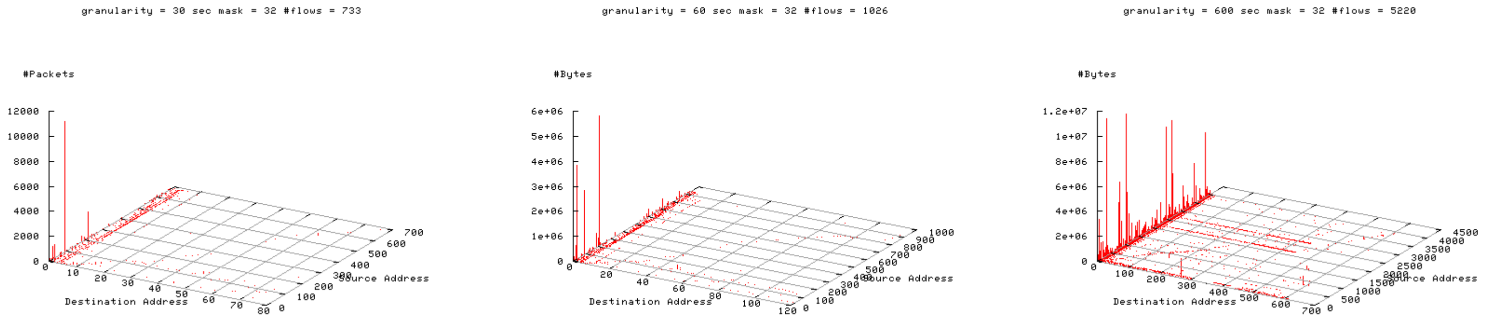

Fig. 4. Distribution of the number of packets sent from a source to a destination address. Each plot is a zoom of one high frequency peak observed (server side) on Fig. 3. Zooming was obtained considering a level of aggregation /32.