An Approach to Detect Traffic Anomalies

Sílvia Farraposo[α], Philippe Owezarski[β] and Edmundo Monteiro[γ]
[α]*School of Technology and Management – Polytechnic Institute of Leiria, Morro do Lena – Alto do Vieiro, 2411-901 Leiria, Apartado4163 – Portugal*
E-Mail : silvia@estg.ipleiria.pt
[β]*LAAS CNRS, 7 Avenue du Colonel Roche,31077 Toulouse, Cedex 4 – France*
E-Mail: owe@laas.fr
[γ]*University of Coimbra, Departamento de Engenharia Informática, Pólo II – Pinhal de Marrocos, 3030-290 Coimbra – Portugal*
E-Mail: edmundo@dei.uc.pt

The occurrence of traffic anomalies is always responsible, at some scale, for a degradation of performance, which can be observable 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. The diversity of anomalies has motivated the development of several approaches to detect them, which at the beginning were mainly oriented toward a particular kind of anomaly, and now are more and more generic, trying to be anomaly independent.

In this work we propose a methodology to detect and identify traffic anomalies. To accomplish that, and as a demarcation from similar work, we combine a multi-scale with a multi-criteria sketch-based analysis process. Coupled to this, we use information that can be easily extracted from any type of data collection: number of packets, bytes, starting flows, IP addresses and TCP ports.

With a complete knowledge of traffic anomalies, we intend to define anomaly signatures that could be used in a large range of applications as intrusion detection, routing, traffic engineering, etc…

***Keywords:* Anomaly Detection, Traffic Analysis, Multi-scaling, Sketches.**

1. Introduction

Assuring Quality of Service (QoS) in a network requires a deeper and deeper knowledge of its traffic behavior. If, at the beginning, the main concern of QoS frameworks was to reserve enough resources to assure an accurate data flowing, nowadays concerns are directed to traffic connections interactions and how they affect available resources.

Because traffic is not well behaved, i.e., always with the same pattern, the knowledge and characterization of traffic irregularities seems to be an important research field. This was the starting point for this work.

Traffic irregularities or traffic anomalies can be described as the result of one or more occurrences that changes the normal flow of data over a network. Such occurrences can be triggered by a diversity of things, such as DoS attacks, flash crowds 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. However, trying to control the extent and harshness of these anomalies is one point where major contributions can arise – and this is the main goal of this work.

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 anomaly might be signaled, such as an UDP or TCP SYN flood. Much work like that can be seen in [3, 7, 8]. 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. [1, 2, 10, 16, 17]). It is then possible to issue signatures for different kinds of anomalies. But the signatures, by themselves do not give information about who was responsible for the anomalies, what packets constitute the anomaly, where are they coming from, etc., and they are then hardly usable for network or security managers.

Particularly, in this work we address the problem of detecting anomalies in traffic traces and their characterization/identification, by using a two-action algorithm, in which the first action permits the location/identification of anomalies, while the second action is intended to classify the anomalies using criterions defined by the algorithm.

The use of multi-scale and multi-criteria sketch-based features assures both premises in our approach. The multi-scale feature guarantees that any anomaly is detectable independently of its duration, by screening for anomalies at different time scales. 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, which is not the case for some DDoS attacks. This multi-scale analysis also provides some elements giving richer signatures for the different anomalies that were encountered during this work.

The multi-criteria sketch-based feature consists in tracking variations in time-series at different levels of traffic aggregation. Three different time series are currently used: number of packets, number of bytes and number of new flows. Particularly, each level of aggregation is a sub-space of the all-IP address space, which is recursively divided and screened for anomalous packets/flows. In practice this feature permits the evaluation of the occurrence of anomalies at different levels of aggregation, and for instance, detect anomalies due to small flows, like SYN flood attacks or some types of port scans, for example.

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 used to define a database of anomaly signatures. The construction of a valid database requires the application of the algorithm to several traffic traces, and to observe trends when doing the classification.

The paper is organized as follows. Section 2 describes the multi-criteria 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 two 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. For example, we applied it on the famous and publicly available Auckland 8 or GEANT traces, etc, as well as some traces captured on Renater, the French network for education and research. 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. Section 4 presents some related work in the area of anomalies detection and contrasts our approach with other ones already developed. Finally, Section 5 concludes this paper, by presenting some possible applications for our anomaly detection approach.

2. Description of the Algorithm

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

To reach our goal, the algorithm recursively executes a set of actions that allows from a traffic trace with several hours and millions of packets, to locate in time the anomaly, and to converge to a restricted set of packets that is responsible for the anomaly. Each level of recursivity is defined by the

pair (time scale, level of aggregation). The relationship among the time scales and the levels of aggregation at which an anomaly is detected with permits the identification of the anomalies.

The two recursive actions executed by our algorithm are:

1. Detection of anomalies.
2. Characterization of anomalies.

2.1. Detection of Anomalies

The first action of our algorithm is to detect the existence of an anomaly, i.e., an event that is able to change significantly one of the traffic parameters under evaluation. The first time this action is executed the all IP-address space is used, i.e., the level of aggregation /0 is used. The result is a set of time intervals, extracted from a traffic trace, in which there are suspect packets.

The extraction of those time intervals is assured by the application of the following formula, which will be used at each recursive level that will follow.

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

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

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

So given a trace of duration T, the result of this process is a set of N slots where traffic anomalies were detected, and $N \in [0, T/\Delta]$, and Δ is the time-scale granularity. The full process to detect anomalies is based on the definition that an anomaly is responsible for a variation in the number of packets, bytes or flows (or altogether). In our algorithm an anomaly is detected through the application of the formula above, which detects significant variations in the criterions being analyzed.

In the 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 $\vee$ Bytes $\vee$ 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 *pi* (variation of the number of {Packets $\vee$ Bytes $\vee$ Flows}) exceeds a given threshold. Each threshold has as value $E(p) + k\sigma$, where factor *k* in our formula, permits the detection to be coarser or finer, using smaller or bigger values of k, respectively. Because each *pi* is located in well known timeslots of duration Δ, the application of the formula permits the temporal localization of the anomaly, and like this narrows the search space.

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.

As stated above, the action of detecting traffic anomalies in traffic traces by looking for significant parameters variations is executed in a recursive way. Each recursive level considers for a given time granularity a different level of aggregation, in which the formula above is applied. When considering a level of aggregation other than /0, the IP information of packets starts to be relevant, since the notion of traffic flow is introduced. For this work, we are using the flow definition presented by Claffy et al. [5] 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. Particularly, the algorithm presented in this work considers a flow as a sequence of packets from any source to a destination identified by a tuple (IP network, mask), and a timeout limit of 64 seconds to the inter-arrival time between two packets of a same flow: this is a quite standard definition of an IP flow. Nonetheless, if the method is not considering all the parameters of the five-tuple flow definition, the approach is prepared to include them.

This approach permits to screen all the IP space looking for faulty flows. Faulty flows are spotted at each level of aggregation, which permits us to see its "evolution" as the level of aggregation

decreases. 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.

Because traffic traces captured at core equipments certainly present packets with a level of aggregation that is different from the one presented by traffic traces captured at border routers, it is possible to parameterize between which levels of aggregation we intend to perform the IP screening. So, while for a core network trace it will be enough to screen the IP space between the mask /0 and /24, when considering an access network screening the range between /16 and /32, is more appropriate.

Conceptually, the sketch-based approach described above is able to screen all the IP space from its root (IP 0.0.0.0) to all its leaves, and to extract the IP addresses associated with the anomalies. However, due to performance constraints it was not implemented as we have presented it – the time required to screen the all-IP address space is not negligible. So, some simplifications were assumed, as considering only the following levels: /0, /8, /16, /24 and /32.

2.2. Characterization of Anomalies

The previous action when executed over traffic traces helps to locate the anomalies in time, and to identify the faulty flows. 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. With this action we intend to create collections of data that are able to characterize traffic anomalies, and that can be available to other applications. To assure this it is our intention to create an anomaly signatures database by relating the following information: time granularities, levels of aggregation, traffic parameters – the information obtained from the previous action.

Most of previous work based the identification of anomalies on traffic volume changes as their principal metric [2, 15, 22, 16]. However, as it was showed, some types of anomalies are not able to be detected with those metrics, because they are not directly reflected in the number of packets, bytes or flows. New approaches to solve this problem suggest the use of IP packet header information [17] to obtain more information about the anomalies. Particularly, the use of the IP addresses and ports will permit the characterization of the anomalies being detected.

The characterization process is accomplished by relating the parameters presented in table 1, and by using a distribution function of the volume parameters (packets, bytes and flows) versus the IP feature parameters (IP address and port). The resulting relationships allow the observation of how each volume parameter is affected by each anomalous flow, and its persistence in similar types of anomalies permits the definition of signatures.

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

Table 1. Parameters to characterize anomalies

The multi-scale feature of our approach, is particularly important in this action, since it allows choosing the timescales over which anomalies detection is performed. Because of the time-scaled decomposition we are able to detect changes in network behavior that may appear at some resolution but go un-noticed at others. This aspect has revealed to be important when differencing between types of anomalies, since some are only visible at smaller time scales (e.g., DDoS attacks), while others are noticed at higher time scales (e.g., flash crowds).

Besides time-scale variation, another variable that is important in our 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 aggregation. This behavior is one additional variable to take into account when defining anomaly profiles.

As mentioned before, screening all the IP address is time consuming, and even being acceptable for a post-mortem analysis, it is not for an on-line analysis, which will be our next step. Because of that, when implementing our algorithm some assumptions were assumed to simplify the process and to shorten times of execution. One of the assumptions was about the level of aggregation of flows. So, if we consider a level of aggregation equal to /16 to analyze traffic, the running time of our method is less than 1 minute, but if we consider a range of levels of aggregation, such as /16 to /32, the running time can easily take more than 1 hour. Of course, the choice of a range of levels of aggregation in detriment of a single level of aggregation gives more accurate results.

3. Illustration on few examples

Effective detection and identification of anomalies in traffic requires the ability to separate them from normal network traffic. In this Section we start by presenting data traffic over which our study was conduced, and show how to use our approach to detect anomalies, and to characterize them. Only two types of anomalies are presented, although we have detected other types of anomalies with our approach (as flash crowds, port scans, alpha flows). The choice of a flooding attack and a network scan was due to the different characteristics that they present, and to evidence the diversity of plots that might be necessary to accurately characterize an anomaly.

3.1. Description of data

Our approach has no restrictions about the process used to capture data traffic, being our only restriction the availability on that data of link counts parameters and some IP address and port information. The lack of such information, might compromise the accuracy of the results obtained, but not the algorithm execution.

Besides developing an approach able of detecting and characterizing anomalies, our intention with this algorithm is also to define a database with anomaly signatures, which could be used with any application “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 [19], 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. Capture was made at the Internet link access of the university campus. All traces collected were anonymized, however preserving its address structure.

The called Renater data-set was obtained in the context of the MetroSec project [18], 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, have durations ranging from some minutes to a few days, and include traffic anomalies under study, namely DDoS and flash crowds. As Auckland 8 trace, captures were accomplished with a DAG card, at the Internet access link of LAAS.

The GEANT [9] 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 minute 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 on data sets, the diversity of available traces will permit some accuracy on results.

3.2. Diagnosing anomalies

Despite of all the developments in anomaly diagnosis, the most popular procedure to detect any misbehaviour in network traffic is still visual inspection of plots, 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.

3.2.1. Detection of a flooding attack.

Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks are one of the most common malefic actions over the Internet. This type of attacks consumes the resources of a remote host or network that would otherwise be used to serve legitimate users. Nowadays a diversity of tools is available to accomplish DoS and DDoS attacks, and packet flooding is one of the most common approaches to accomplish it.

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 small time scales, by a significant increase in the number of bytes and packets being sent to a specific destination address, as can be seen in Fig. 1. There we can observe that a destination address (the same on all plots) receives several packets/bytes to several of its ports, and one of those ports receive a large amount of data (port with the high peak). In this particular case, the anomaly was detected at time scales less than 60 seconds (those plots are not showed because they are very similar to the ones presented), which is not always the case. The time-scales at which an anomaly is detected or not, is directly related to the type of anomaly itself, its intensity, its duration, and the other traffic flows.

At this point of the algorithm execution the destination address of anomalous traffic is known. However, defining the correct action to take 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 all traffic? The answer is no, and this is why the type of anomaly must be known, in order to take the correct actions. This is accomplished by action two of our algorithm, which considers also port information and source information, at different levels of traffic aggregation. The consideration of these two IP features is important because they represent another vector of analysis: the level of traffic aggregation.

In the example being presented, the information available at this point is that the destination network, which will be represented as *aaa.0.0.0/8*, is the target of one anomaly, and has had a sudden increase in the number of packets and bytes. So, the application of action two of our algorithm permits the identification of all sources and ports that had sent packets to that destination, and to verify if significant variations had existed. Particularly, Fig. 2 shows for the destination network *aaa.bbb.ccc.0/24* (a subnet of *aaa.0.0.0/8*) which source addresses are responsible for sending the anomalous packets, and which ports are involved, when considering a level of aggregation /24. As we can see, three different sources are sending anomalous packets to different ports of our destination network. Moreover, at this stage of the algorithm it was seen that the flow responsible for the high frequency peak of Fig. 1 was generated by a specific source, and has as destination a specific address of network *aaa.bbb.ccc.0/24* and port 27444, which is Trinoo slave port, an UDP flood attack tool. Also, as we can see in Fig.2 a similar plot is obtained when considering time granularity 60 seconds.

Summarizing, our algorithm states that the occurrence of a “strong” packet flood attack is detected at smaller time granularities (when considering granularities higher than 60 seconds, the increase of traffic due to the anomaly is hidden by other traffic), and it increases significantly the number of packets and bytes exchanged during the attack. Also, graphically Trinoo attacks present a high frequency peak at the IP destination address that is still visible when disaggregating the IP addresses involved.

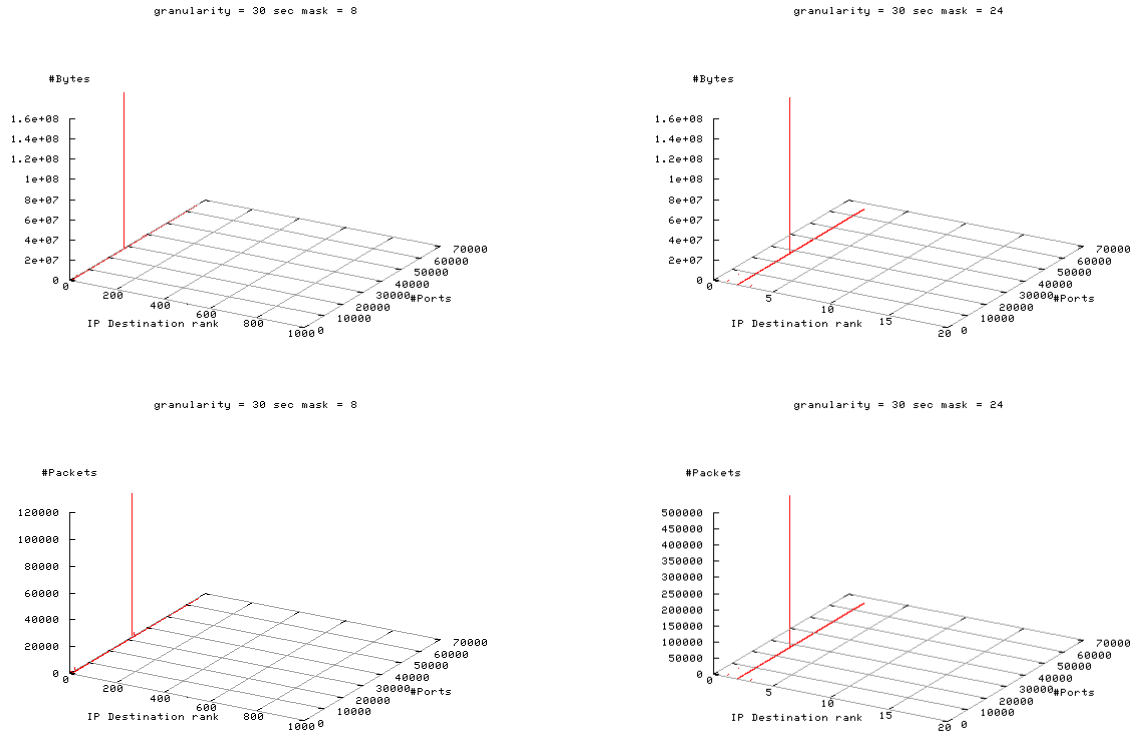

Figure 1 Distribution of the number of bytes and packets received per destination IP address at one particular port, with a level of aggregation /8 and /24. Each destination IP address presented is associated to an anomalous flow.

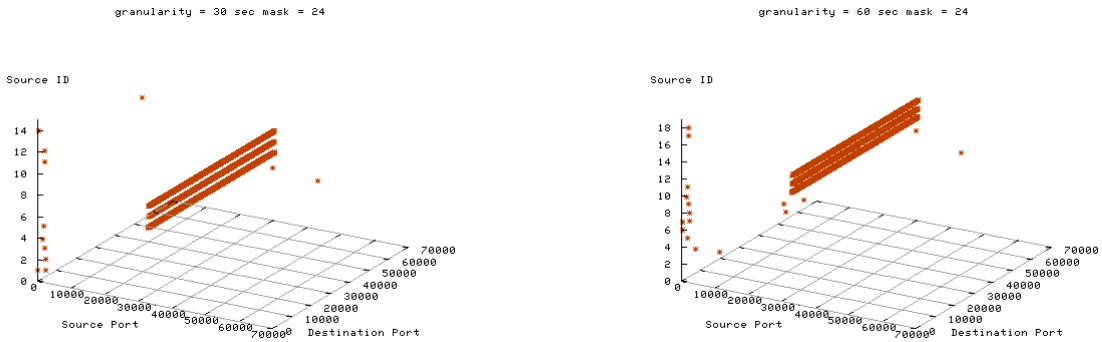

Figure 2 Relationship between the IP source addresses, the source and destination ports toward an anomalous IP destination. These plots were obtained by analyzing the "anomalous" destination address of Figure1 with time granularity 30 and 60 seconds..

3.2.2. Network scan anomaly

Network scanning is a procedure that identifies 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. Usually this type of abnormality remains unnoticed, since small quantities of packets are sent towards

each host. However, by using our approach, and particularly due to its multi-scale feature it is possible to detect such types of anomalies, as the one depicted in Figure 3.

Although, the network scan presented in this example was only detected at small time scales (lower than 60 seconds), this is not always the case. Sometimes, it is only detected at 300 seconds or more. However, in all cases it was seen that this anomaly solely affects the number of packets. Another particularity is that this type of anomaly is only noticed when traffic has some level of desegregation, for instance at level /16 or higher, but not /32. When considering each flow independently, they figure out to be harmless, and not "sharply" variable.

From plots of Fig. 3 we can see a high density of destinations addresses, receiving almost the same quantity of packets (approximately 250). This shape corresponds to the screening of addresses, which is confirmed by the background files created by the algorithm.

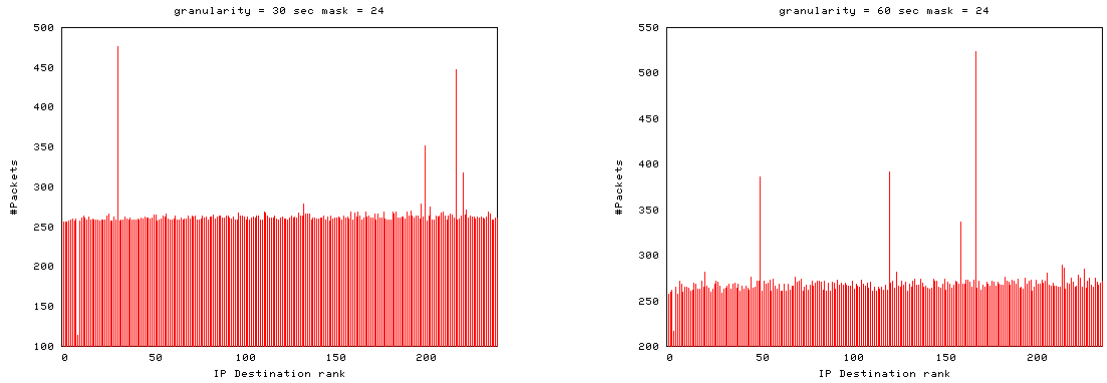

Figure 3 Distribution of the number of packets received per destination network IP address at one particular port, with a level of aggregation /24. Each destination IP address presented is associated to an anomalous flow.

The identification of the network scan against several network destinations with the address 140.93.X.X, was also corroborated by our signature plots. These plots are a sequence of four shapes that are the same for each type of anomaly. For a network scan the visual signature is the one presented in Fig. 4.

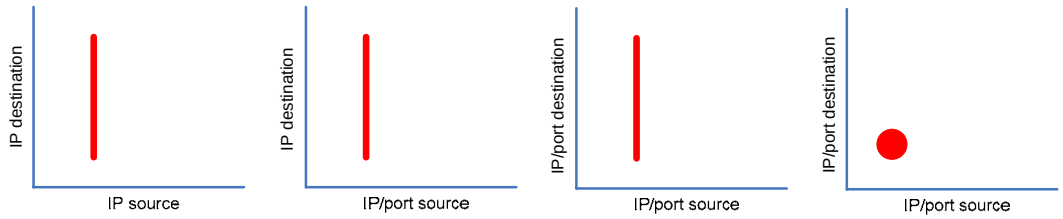

Figure 4 Visual signature for a network scan. The sequence of shapes shows that one IP source is sending packets to several destination addresses, using the same ports numbers.

Fig. 5 shows the four shapes that we have obtained when analyzing the network scan presented above. These plots relate the IP source/destination addresses with the source/destination ports of the anomalous flows (the IP features used to obtain the visual signature). Considering the plots from left to right, and top to down, we can see that the same sequence of shapes, as the one presented in Fig. 4 is depicted: in plot 1 we can see a straight line, near source number zero, which is also present at

plots 2 and 3. Such straight line is reduced to one point at plot 4. So, related to the network scan under study, we can say that a single source, using the same port number has sent a few packets to several IP destination addresses.

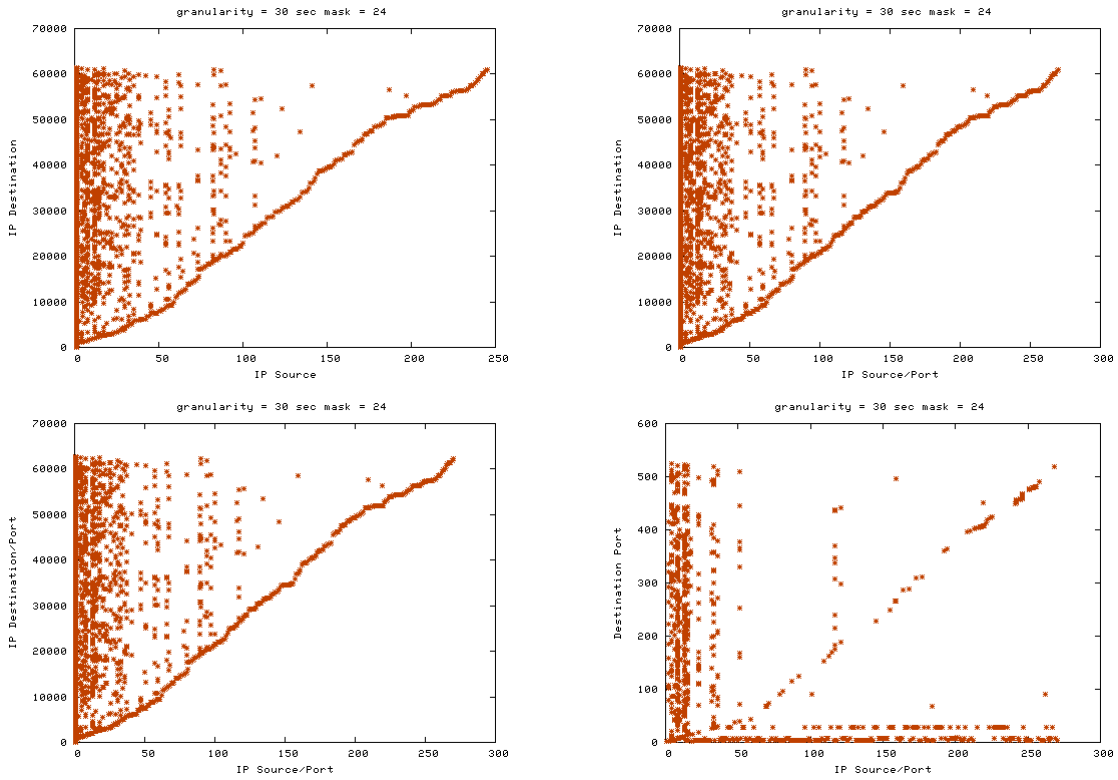

Figure 5 Distribution of IP features in an anomalous flow. Each plot shows how an IP feature is distributed in a network scan.

4. Related Work and Discussion

Several approaches to recognize anomalies in traffic traces have been presented until now, however, most of them only screens for anomalies over one dimension, which could be a limitation. This is true when the goal is to develop an accurate anomaly detection approach, since more and more, anomalies have a multi-dimensional effect over measurement parameters, and a multi-dimensional structure, that considers parameters as time scales, aggregation of addresses, traffic volume, etc. is required to correctly detect, identify and classify them. This is the basis of our work.

Interest in anomaly detection is something that is coupled to data transmission, almost since the beginning. Earlier studies were mainly focused in the definition of models able to predict the occurrence of anomalies. The first model to appear was created by D. Denning [7] in 1983 and was based on the creation of profiles of users based upon their activities. Her work was just the first one, and gave the premises for the definition of profile based intrusion detection systems (IDS), which primary challenges are modelling typical application behaviour, so that attacks by their atypical effects can be recognized without raising too many false alarms. Other pattern based models were developed, as the one of Feather et al. [8]. However, all these models present an important drawback – they require a previous knowledge of all anomalies to detect, named signatures, which limits the time response of the system in front of a new type of anomaly.

The opposite approach, that appeared latter is based on heuristics or rules, rather than patterns or signatures, and detects any types of misuses that fall out with normal system operation. In order to determine what attack traffic is, the system must be taught to recognize normal system activity. Most of these models use one of the following methods to predict normal traffic: the Holt-Winters method [3, 2, 15], the Decomposition method [20] and the Exponentially Weighted Moving Average (EWMA) [22].

Intrinsically related to the detection process is how anomaly detection is accomplished: following off-line approaches, or on-line approaches. While the post-mortem analysis permits a more complete study of data and the extraction of information that otherwise would be impossible, the on-line approach reveals itself more desirable since it does not require saving large amounts of data and the detection process can be accomplished on the fly. However, the applicability of such on-line approaches in backbone networks usually requires special equipment able to process data quickly. Some important contributions were presented in both areas, as the ones of [3, 2, 25, 21] in off-line analysis, and [13, 10] in on-line analysis.

Most of times, performing on-line or off-line analysis constrains how the data to be analyzed is obtained, the latter having a larger choice of options. One of the eligible source for post-mortem analysis is SNMP data, which is available for almost all network equipments. Another common data source is NetFlow, and routing protocols tables, as the BGP tables. One common point among these sources is that they are usually configured to present sampled data, and not all data packets reaching a monitoring point. If on one side this reduces significantly the saved data, on the other side sampled data may not be an accurate representation of data flowing, particularly if sampling times are on the order of several minutes. The use of on-line data usually requires dedicated hardware (as DAG cards [6]) and/or software, which are usually expensive. In both cases, due to the significant quantity of data captured per unit of time, a good processing unit is required. The counterpart of using on-line data is the difficulty to capture information at some points of the network, for example, the backbone of current operator networks, in which very large quantities of information are routed.

Two related and important issues, when defining a methodology to detect and analyze anomalies are the format of data to work on, and what to do with data. A large amount of approaches use what is called volume-parameters, as the number of bytes, packets, or flows. These parameters are available at almost all data sources, and because of that have been widely used by anomaly detection algorithms [8, 3, 2, 15, 21, 16]. However, using volume parameters in solo is anymore a good approach, particularly if the goal is to develop an accurate method to detect and classify anomalies. This is due to the complexity of the anomalies and, how, when and why they affect differently measurement parameters.

Some works have tried to solve this by considering more than one source of data as input [21]. Other authors [13, 17, 14, 24] showed that the use of other criterions, besides volume ones, could strength the detection, since anomalies do not affect all parameters evenly, particularly the most used, as the number of packets and bytes. Considering IP addresses and port information is actually the trend in anomaly detection.

To reach its goal, any approach to detect anomalies must correlate some or all the aspects presented above: the type of system pretended, the type of data to work, the range of anomalies to detect, the better scale to work, etc. Because of this, the diversity of approaches to extract anomalies from traffic data is probably the major wealth in the anomaly detection context. Some methodologies are statistical analysis oriented, using a combination of models and thresholds to detect anomalies as the tools developed by Brutlag et al. [3] and C. Ji [12], or using wavelet analysis as Barford et al. [2], or spectral analysis [4]. Others use methods coming from the artificial intelligence area as [1], while it is also possible to detect anomalies with image processing techniques [13]. Some approaches are oriented to detect specific types of anomalies, as DoS attacks [4, 10], routing problems [21, 11], or worms and viruses attacks [25], while others are more generalist [2, 17] (the actual trend).

The approach presented in this paper constitutes an alternative in the context of detection and analysis of traffic anomalies. When comparing our approach with the few ones presented above, we can present our algorithm as a statistical one, in the sense that a network anomaly is modeled as correlated abrupt changes in network data. An abrupt change is defined as any change in the

parameters of a time series that occurs on the order of the sampling period of the measurement. However, instead of one time series, we work on with three time series (packet-count, byte-count, and flow-count), and over multiple time scales. Our multi-criteria option intends to spread detection, and increase the number of different anomalies able to be detected, since it was showed that anomalies do not affect equally parameters. Our multi-scale option, that we have not seen in any approach to detect anomalies, is related with our conviction that some anomalies appear clearly at some scales, than others. This is the case of anomalies associated to large flows – if only small time scales are used, the long time behavior of the flow might be truncated, and not recognized. The use of different time scales also will permit this approach to be applicable in several cases. For example, if we intend to use this approach in routing area (modify routing behavior because traffic anomalies had been identified), larger time scales are preferable – only anomalies that stay longer than routing updates times are meaningful. On the other side, in the security field, even small-lived anomalies could be of interest.

To obtain more accurate results, we intend to combine the multi-scale and multi-criteria options with a network multi-sketch process – which will permit identifying the anomalies, and characterize them. The use of the multi-sketch process is consistent with the use of packet features trend presented above. Using a multi-sketch like process to screen all or partly the IP address space also permits our approach to be used as a backbone or edge application, i.e., working with more aggregated information (/8 to /16) for the first case, and less aggregated in the second one (/24 to /32).

5. Conclusion

In this paper we have proposed a two-actions 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 the detection of time-scale dependent anomalies, and the multi IP-space axis, which using a sketch-based approach permits efficiently looking for faulty 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 an accurate 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, the algorithm provides us enough information for pointing out specific flows or packets. We then just need discarding those flows or packets.

6. References

[1] *Data mining techniques for effective and scalable traffic analysis*, M. Baldi, E. Barladis, and F. Risso,", proceedings of IM'05, 2005

[2] *A signal analysis of network traffic anomalies*, P. Barford, J. Kline, D. Plonka, and A. Ron, ACM SIGCOMM Internet Measurement Workshop, Marseilles – France, Nov. 2002

[3] *Aberrant behavior detection and control in time series for network monitoring*, J. Brutlag, 14th Systems Administration Conference 2000, New Orleans – USA, Dec 2000

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

[5] *A parameterizable methodology for Internet traffic flow profiling*, K. Claffy, H. Braun, and G. Polyzos, Selected Areas in Communications, IEEE Journal, vol. 13 pp 1481-1494, Oct 1985

[6] *DAG*. At http://www.endace.com/products.html.

[7] *An intrusion-detection model*, D. Denning,, IEEE Transactions on Software Engineering, Feb 1987

[8] *Fault Detection in an Ethernet Network using anomaly signature matching*, F. Feather, D. Siewiorek, R. Maxion,", ACM SIGCOMM, 1993

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

[10] *MULTOPS: a data-structure for bandwidth attack detection*, T. Gil and M. Poletto, USENIX 2001, Boston – USA, Jun 2001

[11] *Detection and analysis of routing loops in packets traces*, U. Hengartner, S. Moon, R. Mortier, and C. Diot, IMW'02, Marseille – France, 2002

[12] *Proactive network fault detection*, C. Hood and C. Ji, IEEE Infocom'97, 1997

[13] *A Study of analyzing network traffic as images in real-time*, S. Kim and A. Reddy, IEEE Infocom'05, Florida – USA, 2005

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

[15] *Sketch-based change detection: Methods, evaluation, and applications*, B. Krishnamurthy, S. Sen, Y. Zhang, and Y. Chen, ACM SIGCOMM Internet Conference, Miami – USA, Oct 2003

[16] *Diagnosing network-wide traffic anomalies*, A. Lakhina, M. Crovella, and C. Diot, SIGCOMM'04, Portland – USA, Sept 2004

[17] *Mining anomalies using traffic feature distributions,* A. Lakhina, M. Crovella, and C. Diot, SIGCOMM'05, Philadelphia – USA, Aug 2005

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

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

[20] *Experience in measuring Internet backbone traffic variability: models, metrics, measurement and meaning*, M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewicz, J. Yates, and Y. Zhang,, International Teletraffic Congress, 2003

[21] *IP forwarding anomalies and improving their detection using multiple data sources*, M. Roughan, T. Griffin, Z. Mao, A. Greenberg, and B. Freeman, SIGCOMM'04 Workshops, Portland – USA, Aug 2004

[22] *Grouped data exponentially weighted moving average control charts*, S. Steiner, Applied statistics, vol. 47, no. 2, 1998

[23] *Anomaly detection in IP networks*, M. Thottan and C. Ji, IEEE Transactions on Signal Processing, Vol. 51, no. 8, Aug 2003

[24] *Profiling Internet backbone traffic: behavior models and applications*, K. Xu, Z. Zhang, and S. Bhattacharyya, ACM SIGCOMM' 05, 2005

[25] *Internet intrusions: Global characteristics and prevalence*, V. Yegneswaran, P. Barford, and J. Ullrich, SIGMETRICS'03, San Diego – USA, Jun 2003