SAFEM: Scalable Analysis of Flows with Entropic Measures and SVM

Jérôme François, Cynthia Wagner, Radu State, Thomas Engel

Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg – 6 rue R. Coudenhove-Kalergi, L-1359 Luxembourg, Luxembourg
firstname.lastname@uni.lu

***Abstract*—This paper describes a new approach for the detection of large-scale anomalies or malicious events in Netflow records. This approach allows Internet operators, to whom botnets and spam are major threats, to detect large-scale distributed attacks. The prototype SAFEM (Scalable Analysis of Flows with Entropic Measures) uses spatial-temporal Netflow record aggregation and applies entropic measures to traffic. The aggregation scheme highly reduces data storage leading to the viability of using such an approach in an Internet Service Provider network.**

I. Introduction

Network anomaly detection is a challenging task. One kind of anomalies refers to the attacks. They have highly increased in the last years, where the attack efficiency has skyrocketed. Today, attackers leverage a distributed scheme to perform very aggressive attacks, as for instance large spam campaigns [1], while they are difficult to detect due to their origin which can include millions of hosts, in particular botnets [2].

However, even if they are distributed, attack traffic is still visible in an ISP (Internet Service Provider) network but its traffic is divided into multiple instances. For gathering the global attack traffic, aggregating traffic regarding the IP subnets is a potential solution. In the same way, this drastically reduces the amount of data to store. For example, an ISP in Luxembourg collects more than 60,000 flows per second. In addition to the storage, analyzing high volume of data is a computational bottleneck too. As Netflow data [3] is commonly collected by the ISP, this paper proposes to represent them in an aggregated way. This aggregation helps in reducing the amount of data, while it is still considered entirely unlike sampling techniques [4] where some data is discarded.

Entropic metrics have gained interest in the context of anomaly detection [5], [6], [7], [8], [9], therefore SAFEM (**S**calable **A**nalysis of **F**lows with **E**ntropic **M**easures) leverages the entropy, which is calculated on aggregated profiles. This paper shows that, even if data is represented in an aggregated and scalable way, analyzing it is still possible and efficient for discovering network anomalies.

The paper is structured as follows: the architecture of SAFEM and its different components are introduced in section II. Section III focuses on the evaluation while section IV discusses other work related. Finally, the conclusion and future work are included section V.

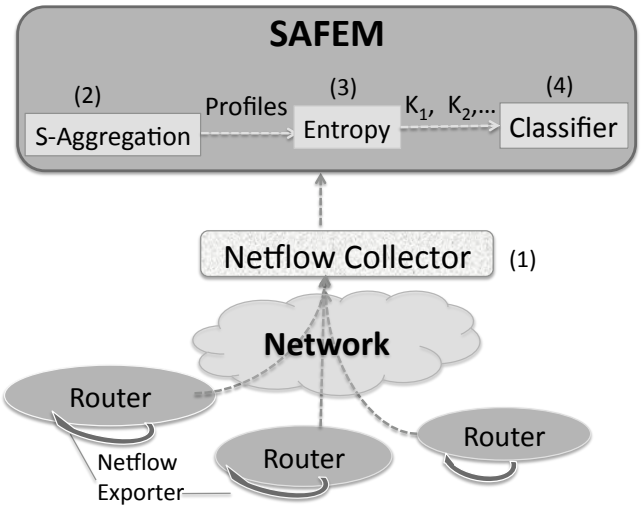


Fig. 1: SAFEM architecture

II. The Architecture of SAFEM

The first component of SAFEM is the *S-Aggregation* module which takes Netflow records as input. They are collected from the network (1) before being aggregated (2) as described in the following section. The entropy is then calculated over the aggregated representation thank to the *Entropy* module (3). Once metrics have been calculated, any classification techniques can be applied.

A. Spatial and Temporal Aggregation

Our method is inspired from AGURI which was presented by Kaizaki et al. [10], but that application is limited to IP packet captures. SAFEM may be directly applied to Netflow records. The records are aggregated regarding two dimensions: space and time.

Spatial aggregation relies on the IP addresses (source or destination) which are extracted from each Netflow record. For an IP address, the proportion of traffic it matches is computed regarding the total volume of traffic. This measure can be expressed as bytes or packets. Then, the IP addresses are represented within a hierarchical tree using the common CIDR format [11]. Hence, the IP addresses of the same subnet are grouped together as well as subnets on higher levels. This is

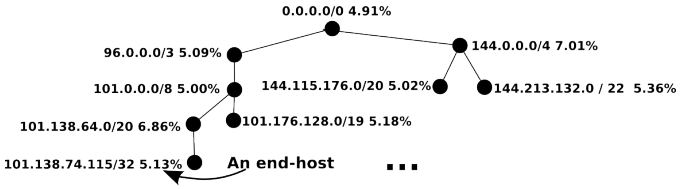


Fig. 2: Partial view of a generated aggregation tree for a byte based destination profile. $\alpha = 5\%$

illustrated in Figure 2. During the hierarchical construction, the volume of traffic associated to nodes is aggregated into their parents. Finally, only nodes having a volume higher (in percentage) than a threshold α are kept.

The second aggregation is performed over time. As usually in anomaly detection schemes, the objective is to detect a deviation from a normal profile. Hence, by splitting the time into periods, SAFEM is able to track changes over a sequence of time windows. The size of time windows is specified by the parameter β. For each time window, the traffic information is aggregated regarding the spatial dimension. So, the temporal aggregation corresponds to divide time into windows and then aggregate data within them.

In brief, traffic is represented by a sequence of trees holding the traffic as in an aggregated way based on two criteria:

- the exchanged bytes or packets,
- the direction of traffic: source or destination of IP addresses.

As noticed, the two main parameters are α and β, which control the granularity of the representation. Hence, with small values very small traffic changes can be detected but this includes also normal microscopic deviations, which should not be considered as anomalies. They can be discarded by increasing α and β. However, if these values are too high, anomalies themselves might be mixed within normal traffic. Thus, a good trade-off has to be made to be able to catch the attack impacts on traffic in an efficient way. The evaluation section will deal with this aspect.

B. Profiles

In our case, the objective is to detect network anomalies and so, traffic deviations from a profile. This kind of methods allows to detect unknown attacks unlike signature based mechanisms which compare traffic with known patterns. There are two main possible options for establishing a profile:

- observe the traffic for a limited period of time and use it as a profile when detecting attacks. Such a profile has to be free of anomalies and so, it is necessary to clean the data prior using another anomaly detection tool or a manual analysis
- consider the profile as the recent past (moving profile). Therefore, the goal is to find traffic changes over the last time windows.

Both approaches have their drawbacks and advantages. The first one may be biased by human related activity patterns (night / day, week days / week-end, holidays / working days). Hence, a profile has to be constructed for every potential pattern and updated periodically since the user activities also change from a long term perspective. For instance, network traffic should be different from one month to another. In brief, establishing all profiles and keep them up to data is quite challenging. However, they may detect stealthy attacks since they can detect slight changes whereas using a moving profile can fail to detect slight changes. A moving profile is not subject to traffic pattern changes like day patterns, if the considered time windows are small enough. For instance, if the order of windows is in seconds, there would not be a drastic changes at the end of the working day, since all employees will not exactly leave at the same time and so not stop working at the same time.

Regarding our context, detecting anomalies at ISP level, a moving profile seems more suitable. First, there will be less drastic changes as mentioned, since Internet user activities are more diversified than in a company. Hence, longer moving profiles can be used, *i.e.* including less recent time windows. This will help in detecting stealthy attacks too.

C. Entropy

Assuming that spatial and temporal trees are available over a sequential set of time windows, the objective is to detect potential deviations in the information they contain. The tree based representation allows to store a large quantity of data into compact representation. This drastically reduces the storage needs. However, the main issue of compacting data is their ability to provide useful and relevant information. Therefore, the Shannon entropy is computed over the subnets represented by the tree. Hence, every IP address of a subnet will be affected by a relative proportion of traffic of the subnet based on the prefix size. Obviously, such an approach discards individual values for each IP address and our evaluation in III highlights its efficiency. The entropy measures the quantity of information in data and so, the dispersion of a data distribution. Therefore, when a network anomaly occurs, traffic from or towards a host should change, involving a traffic distribution change that may be observed using the entropy.

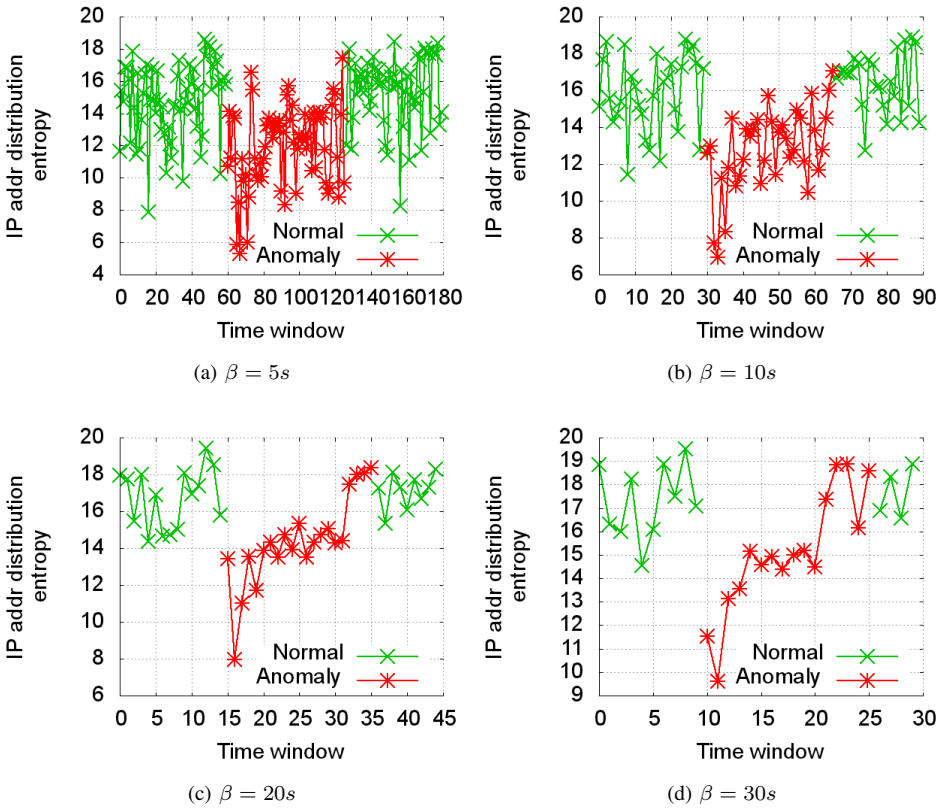
To keep the paper self-contained, the following equation formally defines the entropy:

$$H = -\sum_{i=1}^{n} vol\_ip_i \log_b vol\_ip_i \quad (1)$$

where n is the number of distinct IP addresses, vol_ip_i is the volume associated to the i^{th} IP address. The base of the logarithm scales the result and can be freely chosen.

III. Experimental Results

In this section, the ability of observing anomalies by computing the entropies over the spatial and temporal aggregation of Netflow is assessed. A set of preliminary experiments aims to show, how the entropy value is impacted by the aggregation parameters when they vary. Our evaluation is based on a dataset provided by a major Internet operator in Luxembourg. The global dataset is described in I but 15 minutes have been



(a) $\beta = 5s$ (b) $\beta = 10s$

(c) $\beta = 20s$ (d) $\beta = 30s$

Fig. 3: Temporal aggregation on destination IP addresses – impact of (β)

# Flows	80 792 249
# IP Addresses	470 495 (source), 451 201 (destination)
# bytes	472.6 GB
Avg. bytes/Flow	6 280
# Packets	790 479 054
Avg. Packets/Flow	9.78
# UDP Flows	56 813 288
# TCP Flows	22 021 776
# ICMP Flows	1 870 151
# Other Protocol Flows	87 034

TABLE I: Data Set Description

extracted to highlight the efficiency of our approach to track abnormal changes. There is no need to use the entire dataset as our objective is to show that a short anomaly (around 5 minutes) impacts the entropic metric. However, having longer anomalies could lead to the same conclusions.

The original data is assumed to be free of known anomalies, as it was checked by the operator itself using its own commercial solution [12] which has discovered any kind of anomalies. To assess our method, a synthetic anomaly was injected using Flame [13]. It consists into a TCP flooding attack composed with a burst of 10 packets following by a silence of around 10ms. To enhance the validity, the attack is distributed by changing the origin of packet at every burst.

Due to space limitation, only the impact of β is studied whereas α is set to 1% after a preliminary set of experiments. Figure 3 shows the variation over the time windows. β varies from 5 to 30 seconds. In this epxeriment, the spatial aggregation has considered the traffic volume measure in bytes.

When β is low like in Figures 3(a) and 3(b), many windows containing anomalies have values close to the normal traffic. However, from a global point of view, they are clearly lower. Therefore, by stronger aggregating data, this should enhance the distinction between benign and attack traffic. This case is highlighted in Figures 3(c) and 3(d).

Since TCP flooding attack increase, traffic towards the victim, the distribution is less uniform, leading to a drop of the entropy value, which is clearly distinguished when β is enough high. However, few time windows containing the anomaly could not be easily detected based on a simple threshold. Thus, more advanced technique, such as a machine learning based classifier relying on multiple features (packets and bytes values, source and destination addresses), has to be used for correctly classifying the traffic.

IV. Related Work

Pre-established signatures are well known and efficient techniques to discover known attacks and anomalies. For example, Snort [14] is a common solution that examines every individual packet and compares it to a set of signatures. Rule-based methods, such as in [15], are also helpful for monitoring abnormal host behaviors.

Monitoring all packets is not scalable for Internet operators, which prefer compressed representation like Netflow [3]. Thus, Netflow based intrusion detection has appeared in the past [16]. For example in [17], the authors identify attacks by looking at the number of connections per host, while traffic volume is leveraged in Botminer [18] to detect botnets. Visualization techniques have been proposed in [19]

We proposed kernel functions in the past [20], [21] for measuring dissimilarities in IP addresses based on traffic load, which has led to good results for detecting attacks. However, the computational overhead is very high. This paper uses a a compressed representation of Netflows similar to Aguri [10] for IP packets. There are other alternatives for a scalable storage and representation of flows data [4], [22], [23]. The authors of [22] introduce a column-oriented technique for storing data which provides a better scalability than common solutions relying on row-based techniques. Sampling [4], [23] can also highly reduce the volume of data, but setting a proper sampling rate is still a major issue.

Entropy-based techniques for anomaly detection have been explored in different papers, as for instance [6], [7], [8], [9]. DDoS attacks are detected by statistical methods in [6], but the scalability is limited, since capturing IP packets is needed for extracting some features of them. A worm can be discovered by computing entropies over IP flows [7]. Entropy calculation and time series are combined in [9]. The entropy is also a relevant metric over sampled data as highlighted in [8].

V. Future Work and Conclusion

SAFEM is introduced in this paper. This tool leverages an aggregated representation of Netflow records that compacts data to store, and the entropy computed on a such source of data is still accurate for detecting anomalies. This has been illustrated with a set of preliminary experiments, and future work will focus on the anomaly detection. Especially a classifier will be instantiated and run over the aggregated tree. Such a classifier could be also take multiple features as input, like source and destination IP address, in order to be more accurate. We also plan to study the complexity of SAFEM.

Acknowledgment

This work is partly funded by OUTSMART, a European FP7 project under the Future Internet Private Public Partnership programme. It is also supported by MOVE, a CORE project funded by FNR in Luxembourg. Furthermore, we especially acknowledge Prof. T. Duhautpas from Restena Luxembourg for his counsel.

References

[1] A. Pathak, F. Qian, Y. C. Hu, Z. M. Mao, and S. Ranjan, "Botnet spam campaigns can be long lasting: evidence, implications, and analysis," in *SIGMETRICS: international joint conference on Measurement and modeling of computer systems*. ACM, 2009, pp. 13–24.

[2] R. Hund, M. Hamann, and T. Holz, "Towards next-generation botnets," in *EC2ND: European Conference on Computer Network Defense*. IEEE Computer Society, 2008, pp. 33–40.

[3] B. Claise, "Cisco systems netflow services export version 9," 2004. [Online]. Available: http://tools.ietf.org/html/rfc3954

[4] C. Estan, "Building better netflow," in *Conference on applications, technologies, architectures, and protocols for computer communications (SIGCOMM)*. ACM, 2004.

[5] W. Lee and D. Xiang, "Information-theoretic measures for anomaly detection," in *Symposium on Security and Privacy*. Washington, DC, USA: IEEE, 2001.

[6] L. Feinstein and D. Schnackenberg, "Statistical approaches to DDoS attack detection and response," in *DARPA Information Survivability Conference and Exposition (DISCEX)*. IEEE, 2003.

[7] A. Wagner and B. Plattner, "Entropy based worm and anomaly detection in fast IP networks," in *International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise (WET ICE)*. IEEE, 2005.

[8] A. Lakhina, M. Crovella, and C. Diot, "Mining anomalies using traffic feature distributions," in *Conference on applications, technologies, architectures, and protocols for computer communications (SIGCOMM)*, 2005.

[9] G. Nychis, V. Sekar, D. G. Andersen, H. Kim, and H. Zhang, "An empirical evaluation of entropy-based traffic anomaly detection," in *Conference on Internet measurement*. ACM SIGCOMM, 2008.

[10] R. Kaizaki, O. Nakamura, and J. Murai, "Characteristics of denial of service attacks on internet using AGURI," in *Information Networking*, ser. Lecture Notes in Computer Science, H.-K. Kahng, Ed. Springer, 2003, vol. 2662, pp. 849–857.

[11] Y. Rekhter and T. Li, "Rfc1518: An architecture for ip address allocation with cidr," 1993.

[12] Arbor Networks, "Peakflow SP: Traffic anomaly detection," accessed on 29/03/11. [Online]. Available: http://www.arbornetworks.com/en/peakflow-sp-traffic-anomaly-detection-4.html

[13] D. Brauckhoff, A. Wagner, and M. May, "Flame: a flow-level anomaly modeling engine," in *Conference on Cyber security experimentation and test*. USENIX Association, 2008.

[14] J. Koziol, *Intrusion Detection with Snort*. Sams, 2003.

[15] F. Beck, T. Cholez, O. Festor, and I. Chrisment, "Monitoring the Neighbor Discovery Protocol," in *The Second International Workshop on IPv6 Today - Technology and Deployment - IPv6TD 2007*, Guadeloupe/French Caribbean, Guadeloupe, Mar. 2007.

[16] A. Sperotto, G. Schaffrath, R. Sadre, C. Morariu, A. Pras, and B. Stiller, "An overview of IP flow based intrusion detection," *Communications surveys and Tutorials*, vol. 12, no. 3, pp. 343–356, 2010.

[17] Q. Zhao, J. Xu, and A. Kumar, "Detection of super sources and destinations in high-speed networks: Algorithms, analysis and evaluation," *IEEE Journal on Selected Areas in Communications*, vol. 24, 2006.

[18] G. Gu, R. Perdisci, J. Zhang, and W. Lee, "Botminer: clustering analysis of network traffic for protocol- and structure-independent botnet detection," in *USENIX Security Symposium (SS)*, San Jose, CA, July 2008, pp. 139–154.

[19] C. Wagner, G. Wagener, R. State, and T. Engel, "Digging into ip flow records with a visual kernel method," in *LNCS: Computational Intelligence in Security for Information Systems 2011*, vol. 6694. Springer, 2011, pp. 41–49.

[20] C. Wagner, J. François, R. State, and T. Engel, "Machine learning approach for IP-flow record anomaly detection," in *IFIP Networking 2011*. Springer, 2011.

[21] ——, "Danak:finding the odd!" in *International Conference on Network and System Security (NSS)*. IEEE, 2011.

[22] P. Giura and N. Memon, "Netstore: An efficient storage infrastructure for network forensics and monitoring," in *Recent Advances in Intrusion Detection (RAID)*. Springer, 2010.

[23] I. Paredes-Oliva, "Portscan detection with sampled netflow," in *International Workshop on Traffic Monitoring and Analysis (TMA)*. Springer, 2009.