# Towards Systematic Traffic Annotation

Romain Fontugne[1], Pierre Borgnat[2], Patrice Abry[2] and Kensuke Fukuda[3]
[1]The Graduate University for Advanced Studies, Tokyo, JP [2]Physics Lab, CNRS, ENSL, Lyon, FR
[3]National Institute of Informatics / PRESTO JST, Tokyo, JP

**Categories and Subject Descriptors:** C.2.0 [Computer-Communication Networks]: General

**General Terms:** Security.

**Keywords:** anomaly detection, comparison, traffic classification, traffic annotation.

## 1. INTRODUCTION

Maintaining network resources available and secured in the Internet is an unmet challenge. Hence, traffic classification and anomaly detection have received much attention in the last few years, and several algorithms have been proposed for backbone traffic. However, the evaluation of these methods usually lacks rigor, leading to hasty conclusions. Since synthetic data is rather criticized and common labeled database (like the data sets from the DARPA Intrusion Detection Evaluation Program [6]) is not available for backbone traffic; researchers analyze real data and validate their methods by manually inspecting their results, or by comparing their results with other methods. Our final goal is to provide a common evaluation tool by labelling the MAWI database [2] which is an archive of real backbone traffic traces publicly available. Since manual labeling of backbone traffic is unpractical, we build this database by cross-validating results from several methods with different theoretical backgrounds. This systematic approach permits to maintain updated database in which recent traffic traces are regularly added, and labels are improved with upcoming algorithms. In this paper we discuss the difficulties faced in comparing events provided by distinct algorithms, and propose a methodology to achieve it.

This work will also help researchers in understanding results from their algorithms. For instance, while developing anomaly detector, researchers commonly face a problem in tuning their parameter set. The correlation between analyzed traffic and parameter set is complicated. Therefore, researchers usually run their application with numerous parameter settings, and the best parameter set is selected by looking at the highest detection rate. Although this process is commonly accepted by the community a crucial issue still remains. Let say a parameter set $A$ gives a similar detection rate than a parameter set $B$, but a deeper analysis of reported events shows that $B$ is more effective for a certain kind of anomalies not detectable with the parameter set $A$ (and vice versa). This interesting case highlights the capacity of the algorithm to detect different kinds of anomalies with distinct parameters. However, it cannot be observed with a simple comparison of detection rate. A deeper inspection of the algorithm output is required to effectively compare the results provided by an anomaly detector.

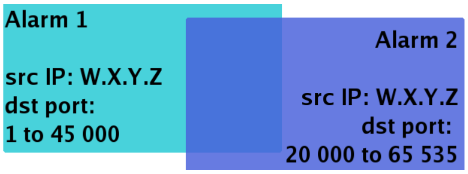

**Figure 1: Two events reporting the same port scan. Alarm 1 could identify only the beginning of this activity, event 2 identify the end of it.**

## 2. DIFFICULTIES

Comparing events reported by several anomaly detectors or traffic classifiers seems at first glance to be trivial, but in practice, it is a baffling problem. The main issue is that events from different algorithms are expressed in distinct ways that are difficult to systematically compare.

The heterogeneity of events results from the diverse traffic abstractions, dimensionality reductions and theoretical tools employed by anomaly detectors and traffic classifiers. For example; (1) hash based (sketch) anomaly detectors [3, 5] usually report only IP addresses and corresponding time bin, no other information (e.g. port number) describe identified anomalies. (2) In previous work [4] we developed an anomaly detector based on image processing that reports events as a set of IP addresses, port numbers and timestamps corresponding to a group of packets identified in analyzed pictures. (3) Several intrusion detection systems take advantage of clustering techniques to identify anomalous traffic [7]. These methods classify flows in several groups and report clusters with abnormal properties. Thereby, the events reported by these methods are sets of flows.

The easiest way to compare those different kinds of event is to digest all of them in the same form. A usual way is to reduce all events to the less restrictive form; meaning in our case that we examine only the source or destination IP addresses. This level of abstraction allows to handle the case illustrated in Fig. 1. However, comparing only IP addresses introduce approximations and errors. An event reporting http traffic from a certain host and another event reporting ssh traffic from the same host should be differentiated.

*CoNEXT Student Workshop'09,* December 1, 2009, Rome, Italy.
ACM 978-1-60558-751-6/09/12.

If we also examine the port information to compare events, then we have difficulties in handling the simple example given in Fig. 1; and so forth, inspecting more event information makes the task harder.

## 3. THE PROPOSED APPROACH

We are now investigating a solution able to handle any kind of event and analyzing all their details. The main idea underlying this approach is to create a graph with events as nodes and find community structure in it. Thus, a community is a set of events, and it represents traffic identified by these events. The main difficulty in constructing such graph is to link the events (nodes) with respect to their similarities. How can we evaluate the similarity between events?

In order to measure events similarities, we need to retrieve the original traffic. For example, let $X$ be an event corresponding to traffic emitted from a single host, and $Y$ an event representing traffic received by another host. $X$ and $Y$ can represent exactly the same traffic but from two different points of view, one reports the source whereas the other reports the destination of the traffic. The only way to verify if these events are related to each other is to investigate the analyzed traffic. If all traffic reported by $X$ is also reported by $Y$, then we conclude that they are strongly correlated. Also, we need a measure to score their similarities.

In constructed graphs, nodes are linked with weighted edges representing the level of similarity between them. The weight of an edge linking events $X$ and $Y$ is computed with the following equation:

$$w = f(X,Y)/min(f(X), f(Y))$$

where f is a function computing the number of packets corresponding to all events given as parameters. $w$ is included in $(0, 1]$, 1 means that events are strongly related whereas values close to 0 represent weak relationships.

One can find similar events by looking at connected component. However, when graphs are generated from events reported by numerous algorithms using several parameter sets, then loose events connect distinct components with edges having a low weight. Algorithms finding community structure [1] helps us in separating those distinct components. For example Fig. 2 is a graph of alarms reported by an anomaly detection method based on gamma modeling [3] (blue circles) and one based on image processing [4] (red and green circles representing results obtained with distinct parameters). Dashed lines represent the community structure found by the algorithm proposed in [1]. In such partitioning of the graph, a community consists of several alarms reporting the same anomalous traffic, and all communities stand for distinct anomalies.

## 4. CONCLUSION

This paper presented the difficulties in examining events reported by anomaly detectors or traffic classifiers. A methodology to compare events expressed in different ways has been proposed. Our approach relies on the abstraction level of graph theory to group different kinds of events standing for same traffic. Graphs are generated from events reported by several methods and the original traffic to uncover the similarities of events. A community mining algorithm permits to distinguish sets of events standing for different traffic. Preliminary results are promising, but still, the evaluation of the proposed method has to be conducted. The proposed methodology will help us in building a common database providing valuable assistance for researchers inspecting backbone traffic. For instance, such database can be used as a ground truth to validate upcoming algorithms.

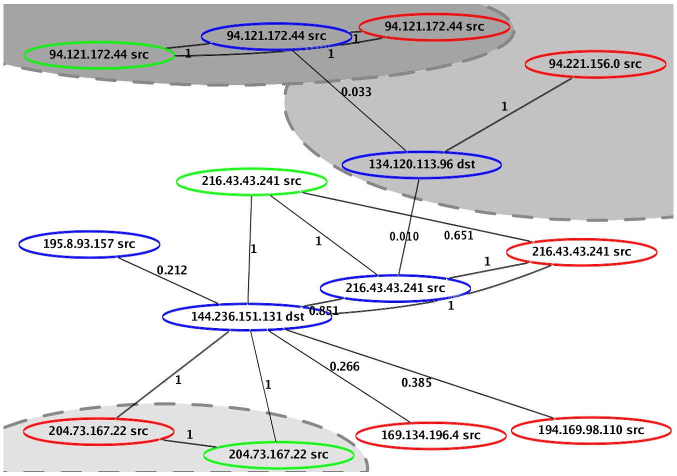

**Figure 2: Example of distinct components linked together. Dashed lines represent the separation in community structure [1]. The green and red circles are alarms reported by a method based on image processing [4], their labels are rough as they stand only for the prominent IP addresses of reported traffic. Blue circles are alarms reported by a method based on gamma modeling [3], their labels are the single IP address reported for each alarm.**

## 5. REFERENCES

[1] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *J.STAT.MECH.*, 2008.

[2] K. Cho, K. Mitsuya, and A. Kato. Traffic data repository at the WIDE project. In *USENIX 2000 Annual Technical Conference: FREENIX Track*, pages 263–270, June 2000.

[3] G. Dewaele, K. Fukuda, P. Borgnat, P. Abry, and K. Cho. Extracting hidden anomalies using sketch and non gaussian multiresolution statistical detection procedures. *SIGCOMM LSAD '07*, pages 145–152, 2007.

[4] R. Fontugne, T. Hirotsu, and K. Fukuda. An image processing approach to traffic anomaly detection. *AINTEC '08*, pages 17–26, 2008.

[5] X. Li, F. Bian, M. Crovella, C. Diot, R. Govindan, G. Iannaccone, and A. Lakhina. Detection and identification of network anomalies using sketch subspaces. *SIGCOMM '06*, pages 147–152, 2006.

[6] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das. The 1999 darpa off-line intrusion detection evaluation. *Computer Networks*, 34(4):579 – 595, 2000.

[7] R. Sadoddin and A. A. Ghorbani. A comparative study of unsupervised machine learning and data mining techniques for intrusion detection. *MLDM '07*, pages 404–418, 2007.