1. INTRODUCTION

Despite their diversity, it is well known that traffic anomalies share a common characteristic: they introduce changes in the traffic behavioral aspects defined by certain traffic features, i.e. packet header fields. To apply traffic features for anomaly detection and identification, one promising class of approaches, which were proposed recently [1] [2] [3], have their basis on some analysis of the distribution of the amount of traffic (or the number of flows) over the possible values of the chosen traffic features, namely feature histograms. However, all these approaches have to address the inherent challenge in feature histograms: while feature histograms offer a finer overview of traffic behavior than the classically used volume time series, they suffer from the curse of dimensionality issue.

To address the dimensionality challenge in making use of feature histograms in anomaly detection, the authors of [1] proposed to use entropy as a compact representation of a feature histogram and make anomaly detection based on entropy time series. However, this entropy representation is so compact that a lot of detailed traffic information implied in the feature histogram is lost. To construct a fine-grained model that captures the details of feature histograms, the authors of [2] applied various aggregation strategies such as a hash function technique that reduces the dimensionality of feature histograms while keeping a lossless observation of the entire distribution. However there is a dilemma: a coarse aggregation results in a hard traffic tractability, while to provide better traffic tractability, the use of a fine-grained aggregation results in an increase in memory consumption and processing overhead. Further investigation of feature histogram dimensionality reduction is found in [3] where the authors first restricted their analysis on the well known ports (0-1203) and then applied principal component analysis (PCA) and used the principal components to represent the obtained histograms. While in this way, the histogram dimensionality can be dramatically reduced, it is unclear if the proposed technique in [3] can scale well when the network size is larger: the studied networks in [3] are Class C subnets. In addition, PCA maps the original traffic histogram data into a different space represented by the principal components, making it difficult to track the origin of a detected anomaly in the original traffic feature space.

In this work we propose a novel and simple technique for feature histogram dimensionality reduction while keeping an overview of the original traffic feature data. We find that a feature histogram, when ordered, follows a power-law distribution that can be captured by (1):

\[ |x_i| \leq R_i^{1/(1-p)}; p \leq 1 \]

Figure 1 shows the reordered coefficients of flow counts per destination AS and per source port. The measurement was conducted on a national research and education network, and the traffic feature histograms are based on measurement data collected in a 5min time-bin. It is evident from this figure that these coefficients \( x_i; i = 1, 2 \ldots N \) have a rapid decay when sorted. Particularly the flow count per feature exhibits a power law distribution that can be captured by (1):

\[ |x_i| \leq R_i^{1/(1-p)}; p \leq 1 \] (1)

With this power-low distribution, the flow count per feature signals can effectively be compressed [4], and we call them compressible signals. Specifically, compressible signals \( X \) whose coefficients \( x_i \) obey (1) can be closely approximated by the first \( K \) coefficients resulting in a compressed representation called the best \( K \) sparse approximation such that \( K \ll N \).
The $K$ sparse approximation has an error term [4]:

$$\sigma_K = \|x - x_K\| = (ps)^{(-1/2)}RK^{(-s)}; s = 1/p - 1/2.$$  

(2)

The analysis of the traces collected showed that the flow count for the very first $K$ coefficients, where $K \approx 4 \times 10^{(-3)}N$, is able to approximate the total number of flows per destination AS with less than 20% error, while the flow count $K \approx 4 \times 10^{(-4)}N$ is able to approximate the total number of flows per source port with less than 3% error. To investigate the stability of the $K$ components for feature histogram approximation over time, Figure 2 is presented. Figure 2 evaluates the $K$ components which achieve 20% approximation error for source AS and destination port over a month long period from the first to the 31th January 2011. The figure shows that the number of $K$ components able to approximate the traffic features to achieve a given error is stable over time. To obtain an additional view on the stability of the $K$ components, we investigate in Figure 3 the impact of traffic feature's nature (source/destination) on the accuracy of the number of the $K$ sparse approximation to achieve the same error. In this figure, a single dot at $(x, y)$ indicates the correlation between the number of $K$ approximation, $x$, for a source feature and $K$ approximation, $y$, for a destination feature. Note that the studied features in this case are ports and ASs. Figure 3 shows fairly clear clusters in data points, which indicates a strong positive correlation between the $K$ sparse approximation of the source feature and that of the destination feature. This indicates that the nature (source/destination) of a traffic feature has a low impact on the approximation accuracy since a comparable approximation error is achievable for both the source and the destination feature with the same number of sparse components. Accordingly, the same $K$ can be chosen for both source and destination features, leading to reduced tuning parameters for feature histograms approximation. To give a more direct feeling of the proposed technique, Table 1 presents the number of $K$ sparse components as a rule of thumb for the network we measured.

3. REFERENCES


