Beyond Shannon: Characterizing Internet Traffic with Generalized Entropy Metrics

Bernhard Tellenbach[1], Martin Burkhart[1], Didier Sornette[2], and Thomas Maillart[2]

[1] Computer Engineering and Networks Laboratory, ETH Zurich, Switzerland
[2] Department of Management, Technology and Economics, ETH Zurich, Switzerland
{betellen,martibur,dsornette,tmaillart}@ethz.ch

Abstract. Tracking changes in feature distributions is very important in the domain of network anomaly detection. Unfortunately, these distributions consist of thousands or even millions of data points. This makes tracking, storing and visualizing changes over time a difficult task. A standard technique for capturing and describing distributions in a compact form is the Shannon entropy analysis. Its use for detecting network anomalies has been studied in-depth and several anomaly detection approaches have applied it with considerable success. However, reducing the information about a distribution to a single number deletes important information such as the nature of the change or it might lead to overlooking a large amount of anomalies entirely. In this paper, we show that a generalized form of entropy is better suited to capture changes in traffic features, by exploring different moments. We introduce the Traffic Entropy Spectrum (TES) to analyze changes in traffic feature distributions and demonstrate its ability to characterize the structure of anomalies using traffic traces from a large ISP.

1 Introduction

Fast and accurate detection of network traffic anomalies is a key factor in providing a reliable and stable network infrastructure. In recent years, a wide variety of advanced methods and tools have been developed to improve existing alerting and visualization systems. Some of these methods and tools focus on analyzing anomalies based on volume metrics, such as e.g., traffic volume, connection count or packet count [1]; others look at changes in traffic feature distributions [2] or apply methods involving the analysis of content or the behavior of each host or group of hosts [3]. However, content inspection or storing state information on a per host basis are usually limited to small- and medium-scale networks. If feasible at all, the link speeds and traffic volumes in large-scale networks hinder a reasonable return on investment from such methods. Most approaches designed for large-scale networks have therefore two things in common: First, they reduce the amount of input data by looking at flow-level information only (e.g., Cisco NetFlow [4] or IPFIX [5]). Second, they use on-the-fly methods that do not rely on a large amount of stored state information. A major drawback of on-the-fly methods is their inappropriateness for approaches relying on the history of traffic feature distributions. A related problem arises, when one wants to visualize the evolution of IP address- or flow size distributions over time. In large-scale networks, these distributions consist of millions of data points and it is unclear how to select a relevant subset.

S.B. Moon et al. (Eds.): PAM 2009, LNCS 5448, pp. 239–248, 2009.

A prominent way of capturing important characteristics of distributions in a compact form is the use of entropy analysis. Entropy analysis (1) reduces the amount of information needed to be kept for detecting distributional changes and (2) allows for a compact visualization of such changes. Evidence that methods based on Shannon entropy capture the relevant changes has been documented [6,7,8].

Here, we propose a new method for capturing and visualizing important characteristics of network activity based on generalized entropy metrics. Our method is a significant extension of the work of Ziviani et al. [9] who recently proposed and studied the use of generalized entropy metrics in the context of anomaly detection. Ziviani et al. introduced a method based on a single generalized entropy value which needs to be tuned to a specific attack. In their evaluation, they provide evidence that generalized entropy metrics are better suited to capture the network traffic characteristics of DoS attacks than Shannon entropy.

Our new method makes the following contributions:

- We define the *Traffic Entropy Spectrum (TES)* for capturing and visualizing important characteristics requiring little or no tuning to specific attacks.
- We demonstrate that the TES can not only be used for the detection of an anomaly but also for capturing and visualizing its characteristics.
- We provide evidence that Autonomous System (AS) entropy is a valuable complement to IP address entropy.
- We confirm the finding of [9] for a broader set of anomalies.

The remainder of this paper is organized as follows: In Section 2, we start with a review of the Tsallis entropy and discuss its advantage over Shannon entropy. Next, we introduce the Traffic Entropy Spectrum (TES) and explain how it is used to capture and visualize distributional changes. Section 3 describes the methodology used for the evaluation. Section 4 discusses the results and outlines how TES could be used to build a detector with integrated anomaly classification. Finally, Section 5 discusses related work and section 6 summarizes the results.

2 The Tsallis Entropy

The Shannon entropy $S_s(X) = -\sum_{i=1}^{n} p_i \cdot \log_2(p_i)$ [10] can be seen as a *logarithm moment* as it is just the expectation of the logarithm of the measure (with a minus sign to get a positive quantity). Given that different *moments* reveal different clues on the distribution, it is clear that using other generalized entropies may reveal different aspects of the data. Two of such generalized entropies relying on *moments* different from the *log-moment* are the Renyi and Tsallis entropies, the latter being an expansion of the former. Here, we use the Tsallis entropy

$$S_q(X) = \frac{1}{q-1}\Big(1 - \sum_{i=1}^{n} (p_i)^q\Big) \tag{1}$$

as it has a direct interpretation in terms of moments of order q of the distribution and has also enjoyed a vigorous study of its properties [11,12,13,14,15] From these properties, it follows that Tsallis entropy is better suited to deal with non-Gaussian measures, which are well-known to characterize Internet traffic [16,17,2], while Shannon's entropy is better adapted to Normal distributions. Note that this and other interesting aspects of the Tsallis entropy are the reason why it has many applications to complex systems[1].

2.1 Meaning of the Parameter q

When using the Tsallis entropy, there is not one Tsallis entropy but as many as there are possible choices for q. Each q reveals different aspects of distributions used to characterize the system under study. Before we take a closer look at the meaning of q, we need to define what kind of distributions we want to use and how we get it. We start with the definition of important terms used in the reminder of the paper:

- *system*: A (set of) network(s) described by an ensemble of network flows
- *feature*: Any flow property that takes on different values and whose characterization using a distribution is potentially useful. Flow properties used in this study are: source and destination IP address, source and destination port and origin and destination Autonomous System (AS).
- *element* i: A specific instance of a feature (e.g., source IP address `10.0.0.1`)
- *activity* a_i: The number of occurrences of element i within a time slot of size T. Slot sizes used for this study are: 5, 10 and 15 minutes.
- *feature distribution*: The probability distribution $P[I = i] = p_i = \frac{a_i}{\sum_j a_j}$ of, e.g., the feature *source port*. Note that p_i can also be interpreted as *relative activity* of i. These feature distributions serve as input for the Tsallis entropy calculation.

We now discuss the meaning of different values of q. First, it is essential to stress that both $q = 0$ and $q = 1$ have a special meaning. For $q = 0$, we get $n - 1$, the number of elements in the feature distribution minus one. For $q = 1$, the Tsallis entropy corresponds to the Shannon entropy. This correspondence can be derived by applying l'Hôpital's rule to (1) for $q \longrightarrow 1$. For other q's, we see that (1) puts more emphasis on those elements which show high (low) activity for $q > 1$ ($q < 1$). Hence, by adapting q, we are able to highlight anomalies that

1. increase or decrease the activity of elements with no or low activity for $q < 1$,
2. affect the activity of a large share of elements for q around 1,
3. increase or decrease the activity of a elements with high activity for $q > 1$.

2.2 The Traffic Entropy Spectrum

To leverage the full capabilities of Tsallis entropy, we introduce a new characterization and visualization method called the Traffic Entropy Spectrum (TES). The TES is a three axis plot that plots the entropy value over time (first axis) and for several values of q (second axis). For convenient 2D presentation, the third axis (showing the normalized entropy values) can be mapped to a color range. Hence, the TES illustrates the temporal dynamics of feature distributions in various regions of activity, ranging from very low activity elements for negative qs to high activity elements for $q > 1$.

But what values should be used for the parameter q and do they need to be tuned to the characteristics of the network traffic at a specific sensor? By experimenting with

[1] See `http://tsallis.cat.cbpf.br/biblio.htm` for a complete bibliography

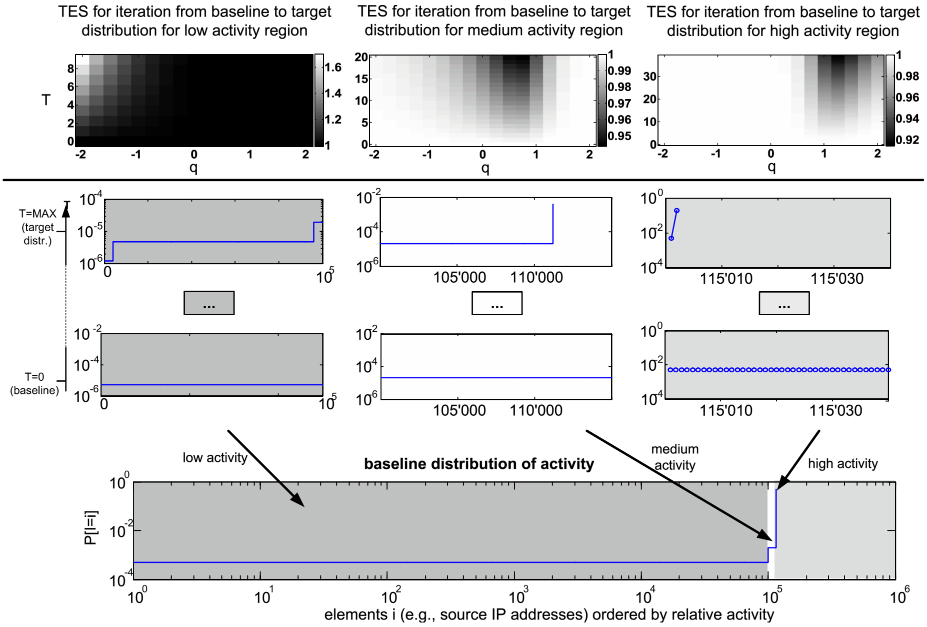


Fig. 1. Impact of changes to different regions of the distribution. Bottom: Baseline and target distributions for low, medium and high activity regions. Top: Resulting TES when altering the distribution in the respective region from the baseline to the target distribution in multiple, even sized, steps.

traces from different sensors and years (2003 to 2008) showing largely differing traffic characteristics, we found that the selection $q = -2, -1.75, ..., 1.75, 2$ gives sufficient information to detect network anomalies in all of those traces. Large values $q > 2$ or smaller values $q < -2$ did not provide notable gains. Hence, this choice of qs worked for many different traces and is therefore strong empirical evidence that it requires little or no tuning to the traffic characteristics of a sensor.

To illustrate the meaning of the parameter q and the TES, we make use of an artificial feature distribution $P[I = i]$ of elements i (see Figure 1) where we identify exactly three different regions. Each region contains elements that show either *low*, *medium*, or *high activity*. Note that for simplicity, all elements in a region have the same absolute activity. We first look at the impact of modifications that are (1) limited to one of those regions and (2) that do not affect the total contribution of this region to $\sum p_i = 1$. To see how the TES reacts to such changes, we specify suitable target distributions for each region (see Figure 1). We then iteratively transform the distribution of a region starting from the baseline distributions in time slot $T = 0$ to the target distributions. We then divide the entropy values we get for the time slots T by the value of the baseline ($T = 0$). Hence, values less than one denote a decrease and values greater than one an increase in entropy compared to the baseline. Figure 1 shows the response of the TES for the transformations of the different regions. Inspecting the TES for the different modifications reveals that they behave as expected:

- high activity: reducing the # of elements decreases entropy for $q > 1$
- medium activity: reducing the # of elements decreases entropy for $-1 < q < 1$
- low activity: reducing the activity of some elements increases entropy for $q < -1$

3 Methodology

For anomaly detection with real traffic traces, we calculated the TES on the activity of the following flow features: Source- and destination IP address, source- and destination ports, origin and destination Autonomous System. We did this for each of the protocols TCP, UDP, ICMP and OTHERS separately.

3.1 Calculating the TES

The calculation of the TES is straightforward. We aggregated the sample distribution of the various traffic features over an interval of 5, 10 and 15 minutes. While the results using the 15 minutes interval are much smoother, shorter intervals are better suited to point out anomalies that last only tens of seconds or a few minutes. At the end of each interval, we calculated the Tsallis entropy values for the different qs and stored them for visualization using the TES. Note that with our selection of qs, we need to store a set of 17 values per interval only.

After calculating the TES, we apply two different normalization methods to compensate for the large absolute difference of the entropies for different q's:

- Global normalization using the maximum and minimum entropy value for a given q during the observation period as follows $S_{normalized,q} = \frac{S_q - minS_q}{maxS_q - minS_q}$. This maps all entropy values to the range [0,1].
- Normalization using the maximum and minimum entropy for a given q on a training day, for instance before the anomaly under scrutiny. Here, we map entropy values between the minimum and maximum of the training day to [0,1]. Other values are either above 1 or below 0.

The TES based on global normalization is used to identify dominating changes. If such a dominating change is present, it stands out at the cost of a decreased visibility of non-dominating changes. The second normalization is used to assess whether changes stay within the variations of the training day. Using the second normalization method, it is easy to develop a simple anomaly detector. Values going below the minimum or above the maximum of the training day, expose the anomalous parts of the TES only. Even though this detection procedure is very straight-forward, our evaluation shows that this simple method is already sufficient for detecting and classifying critical anomalies in network traces.

3.2 Anomaly Characterization

Malicious attacks often exhibit very specific traffic characteristics that induce changes in feature distributions known to be heavy-tailed. In particular, the set of involved values per feature (IP addresses or ports) is often found to be either very small or very large. In a DDoS attack, for instance, the victim is usually a single entity, e.g., a host or a router. The attacking hosts, on the other hand, are large in numbers, especially if source addresses are spoofed. Similarly, if a specific service is targeted by an attack, a single destination port is used, whereas source ports are usually selected randomly. In general, specific selection of victims or services leads to *concentration* on a feature and, in turn, to a change in the high activity domain. In contrast, random feature selection results in *dispersion* and impacts the low activity domain (e.g., spoofed IP addresses only occur once in the trace). Knowing this, it is possible to profile an attack based on the affected activity regions for each feature.

4 Application on Network Traces

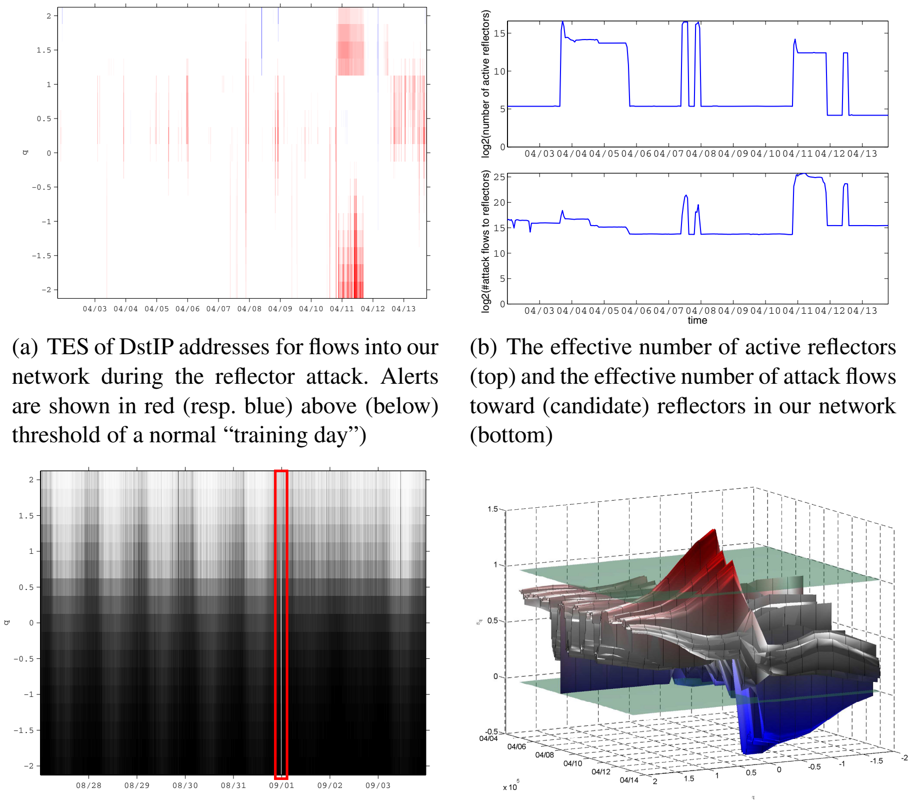
The data used in this study was captured from the five border routers of the Swiss Academic and Research Network (SWITCH, AS 559) [18], a medium-sized backbone operator that connects several universities and research labs (e.g., IBM, CERN) to the Internet. The SWITCH IP address range contains about 2.4 million IP addresses, and the traffic volume varies between 60 and 140 million NetFlow records per hour. The records are collected from five different border routers which do not apply any sampling or anonymization. We study the effect of TES using five well-understood events:

- **Refl. DDoS:** A reflector DDoS attack involving 30,000 reflectors within the SWITCH network, used to attack a web server. Two weeks of traffic were analyzed including some preliminary scanning activity (April 2008). Figure 2(a) shows the TES for incoming DstIPs. The attack is clearly visible around 04/11 and lasts for almost one day. Figure 2(b) shows the effective activity of the reflectors during a two-week period. The sustained activity on 04/04 and 04/05 without attack flows suggests that attackers are scanning the network for potential reflectors.
- **DDoS 1:** A short (10 min.) DDoS attack on a router and a host with 8 million spoofed source addresses (Sept. 2007). DstPort is TCP 80. Figure 2(c) plots the TES for incoming Autonomous System (AS) numbers. The attack is nicely visible for $q < 0$ on the 09/01. Although the covered period is 8 days, the attack is visible with an excellent signal to noise ratio and *no false alarms*. Note that for Shannon entropy ($q = 1$) the peak is insignificant.
- **DDoS 2:** A long (13h) DDoS attack on a host with 5 million spoofed source addresses (Dec. 2007/Jan. 2008). DstPort is TCP 80.
- **Blaster Worm:** Massive global worm outbreak caused by random selection/ infection of new hosts, exploiting a RPC vulnerability on TCP DstPort 135 (Aug. 2003).
- **Witty Worm:** Fast spreading worm exploiting a vulnerability in ISS network security products. Uses UDP SrcPort 4000 and random DstPort (March 2004).

4.1 Patterns in Real Traffic

In this Section we analyze the spectrum patterns exhibited by the attacks described previously. For describing these patterns we use a shorthand notation representing the state of S_q with respect to the thresholds by a single character c_q:

$$c_q = \begin{cases} \text{'+' if } S_q \geq max\ S_q \text{ of the training day (positive alert)} \\ \text{'-' if } S_q \leq min\ S_q \text{ of the training day (negative alert)} \\ \text{'0' else (normal conditions)} \end{cases}$$



(a) TES of DstIP addresses for flows into our network during the reflector attack. Alerts are shown in red (resp. blue) above (below) threshold of a normal "training day")

(b) The effective number of active reflectors (top) and the effective number of attack flows toward (candidate) reflectors in our network (bottom)

(c) TES of origin autonomous systems in the incoming traffic during the DDoS 1 attack represented with global normalization

(d) 3D TES for incoming SrcPorts before and during refl. DDoS attack for $q = -2...2$. Diagonal axis: date (10 days), vertical axis: normalized entropy. Transparent layers: MIN and MAX at normal week days

Fig. 2. Reflector DDoS and DDoS 1

By a *spectrum pattern* we denote the consecutive c_q's for a representative set of values of q. In particular, we compute the pattern for $q = [-2, -0.5, 0, 0.5, 2]$. For instance, the pattern `--0++` means that S_q is below threshold for $q = [-2, -0.5]$, above threshold for $q = [0.5, 2]$ and in the normal range for $q = 0$. The following table shows the spectrum patterns for the described attacks:

		Src IP					Dst IP					Src Port					Dst Port					AS				
	q =	-2	-½	0	½	+2	-2	-½	0	½	+2	-2	-½	0	½	+2	-2	-½	0	½	+2	-2	-½	0	½	+2
Refl. DDoS	in	+	+	0	-	-	+	0	0	0	+	-	-	0	+	0	+	+	0	-	-	+	+	0	-	-
	out	+	0	0	+	0	+	+	0	-	-	+	+	0	-	-	-	-	0	+	0	+	+	0	-	-
DDoS 1	in	+	+	+	+	0	+	+	0	-	-	+	+	0	-	-	0	0	0	-	-	+	+	+	+	0
	out	0	0	0	-	-	+	+	+	+	0	0	0	0	-	-	+	+	0	-	-	+	+	+	+	0
DDoS 2	in	+	+	+	+	0	+	+	0	-	-	+	+	0	+	+	+	+	0	-	-	+	+	0	-	0
	out	0	0	0	+	0	0	0	0	0	0	0	+	0	0	0	0	0	0	-	-	0	0	-	-	-
Blaster W.	in	+	+	+	-	0	+	+	+	+	0	+	+	0	-	0	+	+	-	-	-	+	+	+	-	0
	out	+	+	0	0	0	+	+	+	+	0	+	+	0	-	0	+	+	0	-	-	+	+	+	-	-
Witty W.	in	0	0	0	-	-	+	+	+	+	0	+	+	0	-	-	+	+	+	+	0	0	0	0	+	+
	out	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

For each attack, incoming and outgoing traffic is considered separately. Selected features are src/dst addresses and ports as well as the AS numbers.[2]

The web servers used as reflectors in the refl. DDoS attack appear in the incoming DstIPs (requests from the real attackers). The number of reflectors (30,000) was large enough to increase the area of the high activity domain, resulting in a positive alert for $q = 2$. The relative activity of rare events was further reduced, amplifying their impact in the low activity domain and resulting in another positive alert for $q = -2$. The victim, being a single high activity host, had a contrary influence on the outgoing DstIPs and AS. The relative activity of other hosts was reduced by the appearance of the new heavy hitter and thus the overall area of the high activity domain was decreased. The reduction in relative activity also occurred to the already rare hosts, again amplifying their impact in the low activity domain. A similar effect is observed in the incoming DstPorts, where a concentration on port 80 is induced by the attack. However, the incoming SrcPorts where randomly distributed and activated virtually all ports. As a consequence, the former rare ports experienced a lift in activity and did not contribute to the low activity domain anymore, leading to negative alerts for $q < 0$. Figure 2(d) nicely illustrates the observed pattern (`--0+0`). Note that the patterns are symmetric with respect to the diagonal. That is, changes in incoming SrcIP/SrcPort columns are reflected in outgoing DstIP/DstPort columns and vice versa. This indicates that the reflectors actually managed to reply to all requests (no egress filter was in place).

The main difference between the refl. DDoS and the ordinary DDoS attacks is that the former uses real hosts (the reflectors), whereas the latter uses massively spoofed source IP addresses. For both attacks, the incoming SrcIP TES was affected over a wide range (`++++0`), including the SrcIP count ($q = 0$). For the DDoS 2, however, the alerts in outgoing DstIPs is missing because no response flows were generated.

For both, the Blaster and the Witty worm, destination addresses of spreading attack traffic were generated randomly, much the same way as sources were spoofed during the DDoS attacks. And in fact, the pattern exhibited by incoming worm DstIPs is exactly the same as the pattern for incoming DDoS SrcIPs. The pattern produced by random feature selection (`++++0`) is also visible in incoming DstPort for the Witty worm. On the other hand, the pattern specific to feature concentration (`++0--`) is for instance visible in incoming Witty SrcPort (fixed to UDP 4000), incoming refl. DDoS DstPort (fixed to TCP 80) or incoming DstIPs for DDoS 1 and 2.

Random feature selection can have a different impact on ports than on IP addresses. Whereas incoming DstPort for Witty shows the typical pattern, the one for incoming SrcPorts of the refl. DDoS looks quite different (`--0+0`). Random selection of IP addresses leads to many addresses with very low activity because the range of potential addresses is big. For ports, the range is limited to 65535 values. Thus, if intensive random port scanning is performed, all ports are often revisited and become frequent, basically eliminating the low activity area. This is what happened in the refl. DDoS case, indeed. We conclude that for ports, the strength (volume) of the attack plays a crucial role. For low volume attacks, the random port pattern looks like the random IP pattern, however, increasing attack volume shifts the pattern toward `--0+0`.

[2] Note that our traffic is recorded at a single stub AS. Consequently, source AS are shown for incoming and destination AS for outgoing traffic, respectively.

Summing up, we see that fundamental distribution changes such as concentration or dispersion of features are well reflected by different TES patterns and can therefore be used to infer underlying traffic structure. In future work, we will consider the effect of attack volume as well as additional patterns, e.g., the distribution of flow sizes and durations. The final goal is to develop a comprehensive and diverse set of TES patterns, suitable to accurately detect and classify network anomalies. For this, we need to do a more in-depth evaluation to prove that the improved detection sensitivity does not come along with a high ratio of false positives. Because our preliminary results suggest that TES is very robust (e.g., 8 days without a false alarm in 2(c)) even when using our trivial detection approach, we are positive that this will not be the case.

5 Related Work

Shannon entropy analysis has been applied successfully to the detection of fast Internet worms [6] and anomaly detection in general [7,8]. A different application of entropy is presented in [19], where the authors introduce an approach to detect anomalies based on Maximum Entropy estimation and relative entropy. The distribution of benign traffic is estimated with respect to a set of packet classes and is used as the baseline for detecting anomalies. In [9], Ziviani et al. propose to use Tsallis entropy for the detection of network anomalies. By injecting DoS attacks into several traffic traces they search for the optimal q-value for detecting the injected attacks. However, our results suggest that looking at a single time series for a specific value of q is not enough for revealing different types of anomalies. Furthermore, they do not look at negative values of q for which the entropy is very sensitive to changes in the low-activity region of the distribution. This might be linked to the fact that their evaluation is based on sampled or even anonymized traces. Truncation of 11 bits in IP addresses (as applied to the Abilene traces) might remove the formerly rare elements by aggregating them on the subnet level. However, aggregation is not necessarily a bad thing. Our results show that if multiple levels of aggregation such as IP addresses (fine grained) or Autonomous Systems (coarse grained) are used, aggregation turns out to be a powerful tool to reveal and classify anomalies.

6 Conclusion

The characterization and visualization of changes in feature distributions involves the analysis and storage of millions of data points. To overcome this constraint, we propose a new method called Traffic Entropy Spectrum. Our evaluation shows that the TES is very sensitive to changes that are small compared to the overall size of the observed network. Furthermore, we demonstrate that we can capture changes introduced by different types of anomalies using just a few Tsallis entropy values and find that our method does not require adaptation of its parameters even though the network and the underlying traffic feature distributions change significantly. On the detection side, we propose to use the information from the TES to derive patterns for different types of anomalies. We present ideas how we could use them to automatically detect and classify anomalies. In a next step, we plan to do a detailed analysis of the patterns of different anomalies and cross-validate them with traces from various networks. This will eventually enable us to develop a TES-based anomaly detection and classification engine.

References

1. Barford, P., Kline, J., Plonka, D., Ron, A.: A signal analysis of network traffic anomalies. In: IMW 2002: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment, pp. 71–82. ACM, New York (2002)
2. Scherrer, A., Larrieu, N., Owezarski, P., Borgnat, P., Abry, P.: Non-gaussian and long memory statistical characterizations for internet traffic with anomalies. IEEE Transactions on Dependable and Secure Computing 4(1), 56–70 (2007)
3. Dubendorfer, T., Plattner, B.: Host behaviour based early detection of worm outbreaks in internet backbones. In: 14th IEEE WET ICE, pp. 166–171 (2005)
4. Cisco Systems Inc.: Netflow services solutions guide, `http://www.cisco.com`
5. Quittek, J., Zseby, T., Claise, B., Zander, S.: Rfc 3917: Requirements for ip flow information export (ipfix) (October 2004)
6. Wagner, A., Plattner, B.: Entropy based worm and anomaly detection in fast ip networks. In: 14th IEEE WET ICE, Linköping, Sweden (June 2005)
7. Lakhina, A., Crovella, M., Diot, C.: Diagnosing network-wide traffic anomalies. In: ACM SIGCOMM, Portland (August 2004)
8. Li, X., Bian, F., Crovella, M., Diot, C., Govindan, R., Iannaccone, G., Lakhina, A.: Detection and identification of network anomalies using sketch subspaces. In: Internet Measurement Conference (IMC), Rio de Janeriro, Brazil, pp. 147–152. ACM, New York (2006)
9. Ziviani, A., Monsores, M.L., Rodrigues, P.S.S., Gomes, A.T.A.: Network anomaly detection using nonextensive entropy. IEEE Communications Letters 11(12) (2007)
10. Shannon, C.: Prediction and entropy of printed english. Bell System Tech. Jour. (January 1951)
11. Tsallis, C.: Possible generalization of boltzmann-gibbs statistics. J. Stat. Phys. 52 (1988)
12. Tsallis, C.: Nonextensive statistics: theoretical, experimental and computational evidences and connections. Brazilian Journal of Physics (January 1999)
13. Tsallis, C.: Entropic nonextensivity: a possible measure of complexity. Chaos (January 2002)
14. Dauxois, T.: Non-gaussian distributions under scrutiny. J. Stat. Mech. (January 2007)
15. Wilk, G., Wlodarczyk, Z.: Example of a possible interpretation of tsallis entropy. arXiv cond-mat.stat-mech (November 2007)
16. Willinger, W., Paxson, V., Taqqu, M.S.: Self-similarity and heavy tails: Structural modeling of network traffic. In: Statistical Techniques and Applications (1998)
17. Kohler, E., Li, J., Paxson, V., Shenker, S.: Observed structure of addresses in ip traffic. In: Proceedings of the SIGCOMM Internet Measurement Workshop, pp. 253–266. ACM, New York (2002)
18. SWITCH: The swiss education and research network, `http://www.switch.ch`
19. Gu, Y., McCallum, A., Towsley, D.: Detecting anomalies in network traffic using maximum entropy estimation. In: IMC 2005, pp. 1–6. ACM, New York (2005)