Streaming Estimation of Information-theoretic Metrics for Anomaly Detection (Extended Abstract)

Sergey Bratus[1], Joshua Brody, David Kotz, and Anna Shubina

Dartmouth College, NH 03755, USA

Abstract. Information-theoretic metrics hold great promise for modeling traffic and detecting anomalies if only they could be computed in an efficient, scalable ways. Recent advances in streaming estimation algorithms give hope that such computations can be made practical. We describe our work in progress that aims to use streaming algorithms on 802.11a/b/g link layer (and above) features and feature pairs to detect anomalies.

Information-theoretic statistics applied to monitoring of network traffic can be very useful in detecting changes in its character [6, 4, 3]. These metrics make few modeling assumptions about what constitutes normal and abnormal traffic (e.g., [2]), and thus, theoretically, should do very well at adapting to the traffic characteristics of a specific networks, realizing the "home network advantage" of prior knowledge that defenders have over outside attackers.

However, necessary computations place a heavy load on both the sensor CPU and RAM. Thus scalability of methods that rely on precise real-time computations of entropy and other related statistics remains a challenge. Luckily, a new class of streaming algorithms produce practically usable estimated results with much smaller requirements to CPU and RAM ([5, 1]). They have the potential to allow information-theoretics metrics to be scalably used in practice.

Several experimental systems (including Wi-Fi link layer anomaly detectors being developed at Dartmouth) apply entropy of pre-selected packet or session features to produce alerts. Such mechanisms rely on the idea that a change in the character of a feature distribution is suspicious. In our experience, watching a set of features as if they were independent is highly prone to false positives. A change in the entropy of a feature may be due to factors such as normal business day and other workflow cycles. Even the simplest cases of single protocol features require, e.g., some modeling of when a particular protocol is normally expected to be in use.

Conditional entropy between pairs of features are likely to provide a better metric of normal use, because it relies on tracking the average "predictability" of one feature given the knowledge of another. Such relationships are more likely to persist through diurnal cycles, because they are less related to volumes of traffic.

Unusual use of protocol fields is characteristic of many exploits, but sophisticated attackers take pains to disguise it, as IDSes might be watching for it. It is much harder to disguise unusual payloads in such a way that does not introduce unusual statistical effects in pairs of protocol features. Note that rule-based IDS evasion techniques themselves (e.g., [7]) can produce just such effects.

Streaming estimation algorithms open up the possibility of a scalable sampling-based system that allows keeping track of joint distributions, and thus of mutual information-type statistics. Furthermore, the sampling scheme used in the estimation algorithm can be adjusted dynamically depending on how much precision is meaningful and practicable for a particular network.

The 802.11a/b/g link layer is feature-rich and complex. It includes large (2346–2358 bytes) management frames that can contain an entire ring 0 exploit in their L2 payload alone[1]. Besides the frame type and subtype fields, the link layer header may contain 1–4 MAC address fields, 8 bit flags, and 2 16-bit fields, frame sequence number and duration (the distribution of which has been shown[2] to identify wireless chipset–driver combination as a distinctive fingerprint). The earlier WEP encryption implementations could be leveraged to leak encryption key bits by responding to specially crafted injected frames (e.g., the so-called KoreK attack).

Thus this link layer allows a range of interesting attacks and related statistical distibution anomalies. We distinguish between the four levels of features, based on the sensor RAM and CPU requirements to follow them: (a) PHY layer errors as calculated and reported by the firmware (b) frequency of basic events, such as observing deauthentication frames (c) single header field values' frequency distributions, and (d) joint and conditional distributions of pairs of features. Anomalies in (a) may indicate inteference or jamming, (b) serves as good indicators of various DoS-type flooding and resource consumption attacks, whereas (c) and especially (d) expose other attacks that involve unusual headers and payloads.

References

1. Amit Chakrabarti, Graham Cormode, and Andrew McGregor. A near-optimal algorithm for computing the entropy of a stream. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 328–335, 2007.
2. Guofei Gu, Prahlad Fogla, David Dagon, Wenke Lee, and Boris Skoric. Towards an information-theoretic framework for analyzing intrusion detection systems. In *Proceedings of the 11th European Symposium on Research in Computer Security (ESORICS'06)*, September 2006.
3. Yu Gu, Andrew McCallum, and Don Towsley. Detecting anomalies in network traffic using maximum entropy estimation. In *IMC '05: Proceedings of the 5th ACM SIGCOMM conference on Internet measurement*, pages 1–6, 2005.
4. Anukool Lakhina, Mark Crovella, and Christophe Diot. Mining anomalies using traffic feature distributions. *SIGCOMM Comput. Commun. Rev.*, 35(4):217–228, 2005.
5. Ashwin Lall, Vyas Sekar, Mitsunori Ogihara, Jun Xu, and Hui Zhang. Data streaming algorithms for estimating entropy of network traffic. *SIGMETRICS Perform. Eval. Rev.*, 34(1):145–156, 2006.
6. Wenke Lee and Dong Xiang. Information-theoretic measures for anomaly detection. In *Proc. of the 2001 IEEE Symposium on Security and Privacy*, pages 130–143, 2001.
7. Thomas H. Ptacek and Timothy N. Newsham. Insertion, evasion, and denial of service: Eluding network intrusion detection. Secure Networks, Inc., January 1998.

[1] Demonstrated by Ellch and Maynor at BlackHat 2006 for a vulnerability in MacBook's Wi-Fi kernel driver

[2] J.Cache, "Fingerprinting 802.11 Implementations via Statistical Analysis of the Duration Field", `http://uninformed.org/?v=5&a=1`