Investigating Real-Time Entropy Features of DDoS Attack Based on Categorized Partial-Flows

Hamidreza Lotfalizadeh
Electrical and Computer Engineering
Purdue University
West Lafayette, USA
hlotfali@purdue.edu

Dongso S. Kim
Electrical and Computer Engineering
Indiana University-Purdue University Indianapolis
Indianapolis, USA
dskim@iupui.edu

***Abstract*—With the advent of IoT devices and exponential growth of nodes on the internet, computer networks are facing new challenges, with one of the more important ones being DDoS attacks. In this paper, new features to detect initiation and termination of DDoS attacks are investigated. The method to extract these features is devised with respect to some openflow-based switch capabilities. These features provide us with a higher resolution to view and process packet count entropies, thus improving DDoS attack detection capabilities. Although some of the technical assumptions are based on SDN technology and openflow p rotocol, t he m ethodology c an b e a pplied i n other networking paradigms as well.**

***Index Terms*—DDoS attack detection, Software-defined networking, SDN, openflow, P artial flow**

I. Introduction

With advancements in technologies such as IoT, server virtualization, cloud computing and etc., Internet and computer networks have entered a new phase, facing challenges some of which are unprecedented. One of the major challenges of computer networks is Distributed Denial-of-Service (DDoS) attacks [5].

The goal of a Denial-of-Service (DoS) attack is to consume resources of network nodes and components such that they would fail to deliver their services to incoming benign requests. Carl et al. categorize resource consuming attack into two; 1) vulnerability attacks, which attempt to abuse bugs in the protocol or system to consume resources and 2) flooding attacks, which tend to saturate and exhaust system resources, making the node fail to respond to incoming requests [2].

Usually, if the attack is coming from a small number of nodes, it is possible to detect their malicious behaviour, and subsequently block any other incoming packet from them. In contrast, the nature of a Distributed Denial-of-Service (DDoS) is such that it uses a large number of nodes to carry out an attack. Analyzing network behaviour of any single malicious node might not reveal a significant abnormality. However, their aggregated behaviour will have drastic effect on the victim, making it deny services to non-malicious requests.

One of the widely adopted methods for DDoS attack detection is entropy method. Entropy methods evaluate entropy level of a set of network flows statistics, such as packet count for each flow. In this method, entropy of a set of flow statistics are compared against a threshold to determine any anomaly. Depending on the criteria, the anomaly may be above or below the threshold. The main advantage of this method is that it can give a sense of collective behaviour of a group of flows, a feature that is crucial to DDoS attack detection.

II. Related Works

In [4], Feinstein et al. used entropy analysis along with frequency distributions to detect anomalies. Yu et al. [9], [10] showed that packet count statistics have higher similarity in attack flows compared to similarity in normal flows. In [8] Wang and Jia designed a flow statistics process based on openflow protocol in the switch. In [3] David and Thomas proposed a method based on fast entropy with adaptive threshold to detect DDoS attack anomalies. In [6] Mousavi and St-Hilaire proposed a method based of entropy variation in destination IP to detect DDoS attacks that exhaust controller resources. In [7] Qin et al. proposed a method to calculate entropy vector of multiple features and applied a clustering algorithm to create normal pattern model.

Entropy-based methods produce results based on aggregated information from one or more group of flows, which means that flows need to be grouped based on some criteria before being fed into the entropy-based detection machine. Grouping of flows will have direct effect on entropy of each group, and hence our deduction from the aggregated behaviour of flows. This work introduces a new set of grouping criteria for flows.

Although entropy calculation of a group of flows can determine existence of abnormality in aggregated behaviour of flows, it cannot by itself give any clue on exactly which flows are contributing the most to the abnormality. Nevertheless, to the best of our knowledge, there are no works that have investigated grouping of flows for entropy analysis.

III. The proposed method

A. Partial flow statistics collection

This method performs real-time analysis of partial flows. Partial flow is a subsection of the flow which falls in a time window. Therefore, flow statistics of each partial flow pertains only to its corresponding time window. In other words, in each time window, statistics of flows are extracted with no information from previous time windows statistics. Figure 1

This is the author's manuscript of the article published in final edited form as:

Lotfalizadeh, H., & Kim, D. S. (2020). Investigating Real-Time Entropy Features of DDoS Attack Based on Categorized Partial-Flows. 2020 14th International Conference on Ubiquitous Information Management and Communication (IMCOM), 1–6. https://doi.org/10.1109/IMCOM48794.2020.9001690

shows four hypothetical flows in a sequence of time windows. In time window T_1, flows f_1 and f_2 are created and partial flow statistics is performed on the two in this time window. In T_2, f_3 joins and partial flow statistics is performed on three flows. In T_3 however, we don't have any packets for f_2, therefore, partial flow statistics reduces to two flows, f_1 and f_3. In T_4, we see only one packet for f_4 which creates its corresponding flow entry. Also f_2 becomes active again. Therefore in this time window, partial flow statistics is performed on all four flows. The progress goes on until eventually in T_7, we are left with only one flow to perform flow statistics on.

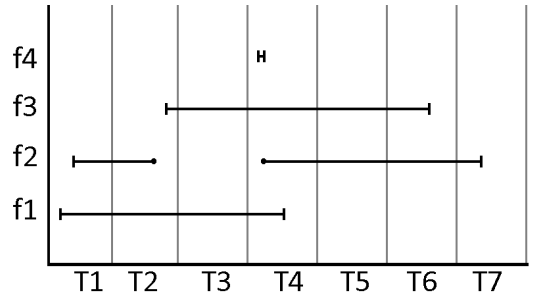

Fig. 1. A hypothetical view of four flows. In each time window, T_1 through T_7, a partial view of each flow is extracted for partial flow statistical analysis.

B. Analysis of aggregated flow tables

After all flow tables are received in controller and aggregated into one, a variety of entropy method can be applied to obtain entropy level and monitor anomalies. Application of entropy method on flow table entries is already well studied. Anomalies can be observed in entropy levels of selected packet features.

In a DDoS attack we expect to see an attack pattern in entropy levels, i.e drastic increase in source IP entropy level, while destination IP entropy level sinks simultaneously. The reason is that since traffic is coming from multiple sources, we expect to see a surge in source IP entropy. In the mean time, high number of packets targeting a victim node should make destination IP entropy level plummet.

C. Grouping flows based on temporal and packet count categories

As mentioned earlier, applying entropy method on a set, does not reveal much about contributing elements. According to Yu et al. in [10], attack flows have similar flow packet count distributions. Also when an attack starts, flow packet distribution of the whole traffic changes.

Smart criteria for grouping partial flows in each time window provides us with a better view into the attack flows packet count distributions.

The first criteria is based on the temporal creation of flows, grouping them into two groups of new and previous flows. DDoS attack flows are usually created in a short time duration. Therefore, we expect to see an attack pattern in entropy level in the group of partial flows that were just created. In this paper, this group of flows are called "new flows" and other non-new flows that were created in a previous time window are called "previous flows". Previous flows are expected to have normal packet count distributions, same as historical patterns when there was no attack. Therefore, entropy level of partial flows that were created in previous windows should have normal entropy pattern.

The next criteria is based on flow packet count. In each time window, partial flows are grouped based on their packet count categories. Normal flows packet count distribution follows an exponential pattern, with the largest group of flows sending only one packet in a time window. Therefore a log-based categorization criteria, with one category specifically for one-packet partial flows would be be the right choice. In experimental results section, $\log_{10}$-based categorization are studied.

After flows are grouped, an entropy formula is applied on each group of partial flows, and the resulting entropy levels is compared for attack analysis. In case of high-rate DDoS attack, we expect to observe attack pattern in entropy level of groups of partial flows with larger number of packet counts, while in low-rate DDoS attacks, the pattern should be observable in groups with low number of packets in each window.

D. Grouping implementation

In this work, flow packet count in each time window is of interest. These information are distributed on flow tables in switches. Each flow entry includes total packet count data. These data will be periodically collected by the controller. In order to aggregate the distributed information, flow tables from edge switches are collected by controller. To calculate flow packet count for the current time window, controller calculates difference of packet count of entries in the newly received flow tables against entries from previously received flow tables.

Furthermore, in order to reduce traffic between switch and controller, only flow entries that have at least one packet in the current time window will be sent to the controller. One way to implement this is by using dirty flag for flow entries. Initially, dirty flag for all entries are reset. With each incoming packet, dirty flag of its corresponding flow entry will be set. Therefore, upon request from the controller, switch can send flow entries with dirty flags set. After successful transmission of flow table to the controller, all dirty flags will be reset and ready for the next time window.

E. Entropy formulation

Equation 1 shows Shannon's entropy formula

$$H(X) = -\sum_{i \in \Omega} p_i \log(p_i) \tag{1}$$

where Ω is the range of the random variable X and p_i is its probability and $\sum_{i \in \Omega} p_i = 1$.

In the formulation pertaining to this paper, $H(X)_k$ is entropy level of the k-th set, Ω_k, and probability distribution of each element in the set is based on the equation 2,

$$p_i = n_i / \sum_{i \in \Omega_k} n_i \tag{2}$$

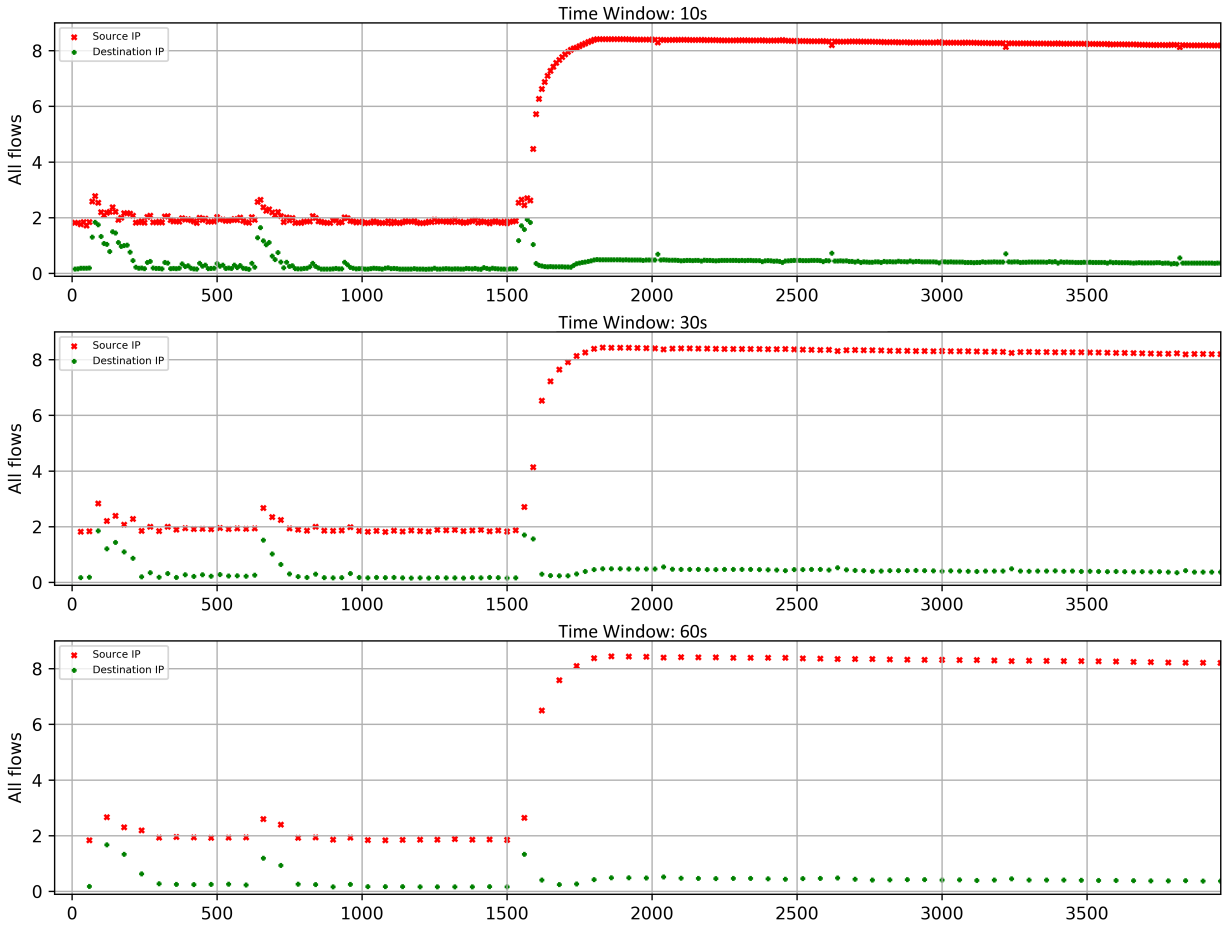

Fig. 2. Source IP and destination entropy analysis of partial flows at different time intervals. No grouping is applied.

where n_i is the number of all packets that have feature i, and $\sum_{i \in \Omega_k} n_i$ is sum of all packets pertaining to partial flows in the group Ω_k.

IV. Experimental Results

In this work, the proposed method was applied on the CAIDA "DDoS Attack 2007" dataset [1]. We applied $\log_{10}$-based categorization, grouping flows in 4 sets, each set being split into newly created and previously created flows. Also each categorization criteria was experimented on time window intervals of 10, 30 and 60 seconds. In total, 24 results were obtained, proofing strength of this real-time partial-flows categorization method.

Applying $\log_{10}$-based criteria for this experiment, the partial flows are separated into groups using equation 3. In this equation, $pktCnt$ is the number of packets of a flow, and $groupNo$ is number of the group to which the flow belongs. Group numbers and corresponding packet count range are presented in table I.

$$groupNo = \begin{cases} 0 & \text{if pktCnt=0} \\ \lceil \log_{10}(pktCnt) \rceil + 1 & \text{if } 0 < \text{pktCnt} \leq 100 \\ 4 & \text{if } 100 < \text{pktCnt} \end{cases} \quad (3)$$

TABLE I
$\log_{10}$-BASED GROUPING PER PACKET COUNT

Group No.	Packet count range
1	1
2	2-10
3	11-100
4	101 and up

Figure 2 shows source and destination IP entropy levels of partial flows at various time windows with no categorization. The timeline for analysis starts from time 0, indicating timestamp of the first packet in this dataset. According to entropy analysis with 10 seconds time window, the DDoS attack starts in time window 1580-1590 seconds. In this window, source IP entropy level jumps drastically to 4.61, while destination IP entropy level dips to 0.85 which is far lower than its historical level in previous windows. The two other entropy diagrams show partial flow entropy analysis with time windows 30 and 60 seconds, which are consistent with the 10 seconds interval diagram. Using basic machine learning tools we can observe start of a DDoS attack in real time.

Subsequent figures show entropy level of groups of partial

flows based on categories. Each figure pertains to a category of flows and has three grid charts, ”prv” (group of previously created partial flows in this category), ”new” (group of newly created partial flows in this category) and ”any” (union of ”prv” and ”new” groups).

Figures 3 through 6 show entropy charts of 10 seconds time windows. It can be clearly observed that the group of partial flows contributing the most to the attack are in categories 3 and 4, sending 10 or more packets in each window of 10 seconds. Also, start and end of the attack is vividly observable by the ridge in entropy of category 3, from windows 1590-1600 to 1790-1800.

Figures 7 through 10 have time window of 30 seconds. Start and end of the attack is vividly observable by the ridge in entropy of categories 3 and 4, from windows 1590-1620 to 1770-1800.

Comparing ”prv” and ”new” entropy charts in categories 3 and 4, it is conspicuous that during attack, entropy level in ”prv” chart is one time window behind ”new”. Rapid surges in source IP entropy level of ”new” partial flows in a current windows will have effect on the group of ”prv” partial views in the subsequent window. This is clear, since newly created flows in the ”new” group will be considered previously created in the subsequent window, being moved from ”new” to ”prv” group.

Figures 11 through 14 show the results for time window 60 seconds. The results of time window are consistent with the previous ones, although with worse time resolution. Nevertheless, start and end of the attack is observable between windows 1560-1620 to 1760-1800.

V. Conclusion

In this paper, the idea of partial flow entropy analysis was proposed. Furthermore, grouping partial flows into groups based on flow packet count categories was proposed and investigated.

The proposal was investigated on three various time windows of 10, 30 and 60 seconds. The 10 seconds time window provides us with better view into groups of flows that contribute the most to the attack. However, collecting partial view statistics from switches in short time intervals can induce a huge traffic to the network. Therefore, a trade off needs to be considered. Alternatively, variable time windows can be applied, zooming time window from larger range to smaller time window. This zooming can be applied if any anomaly is detected.

The next goal is designing a machine learning tool which obtains categorical partial flow entropies and separate flows into normal and attack flows.

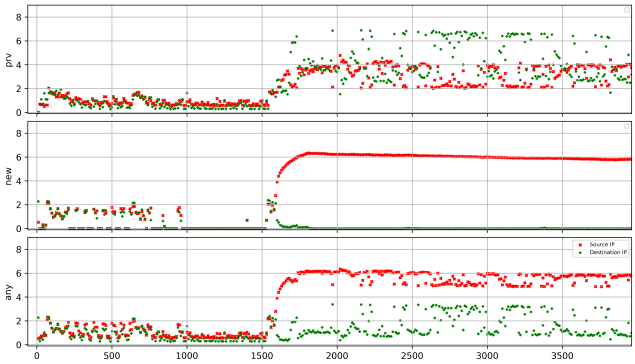

Fig. 3. Entropy charts of category 1 (1 packet partial flows) with time window 10 seconds.

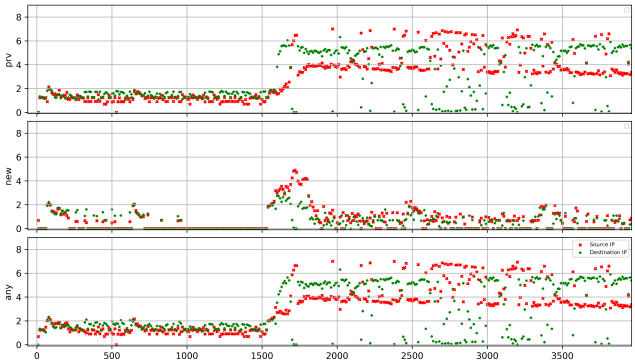

Fig. 4. Entropy charts of category 2 (2-9 packets partial flows) with time window 10 seconds.

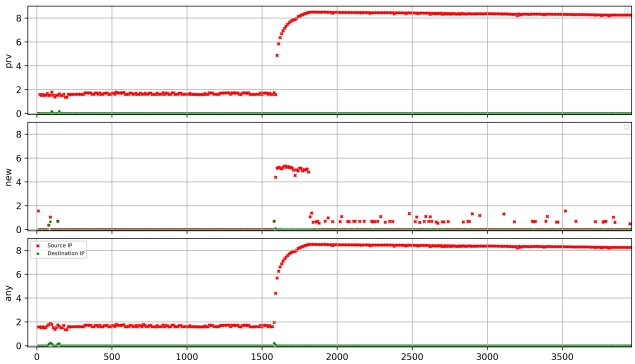

Fig. 5. Entropy charts of category 3 (10-99 packets partial flows) with time window 10 seconds.

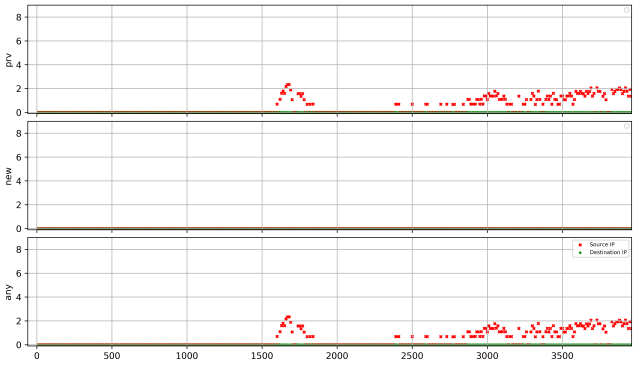

Fig. 6. Entropy charts of category 4 (100+ packets partial flows) with time window 10 seconds.

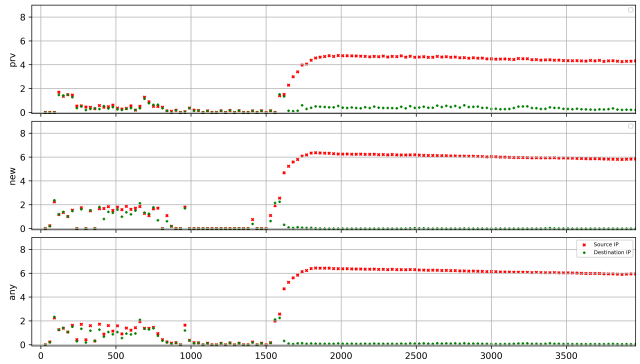

Fig. 7. Entropy charts of category 1 (1 packet partial flows) with time window 30 seconds.

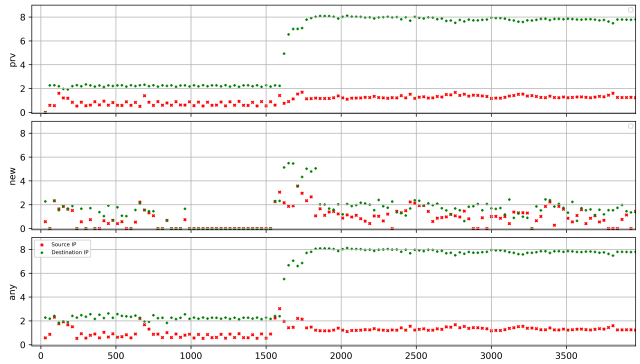

Fig. 8. Entropy charts of category 2 (2-9 packets partial flows) with time window 30 seconds.

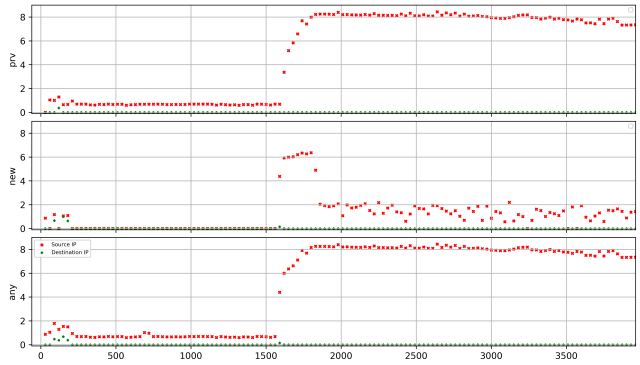

Fig. 9. Entropy charts of category 3 (10-99 packets partial flows) with time window 30 seconds.

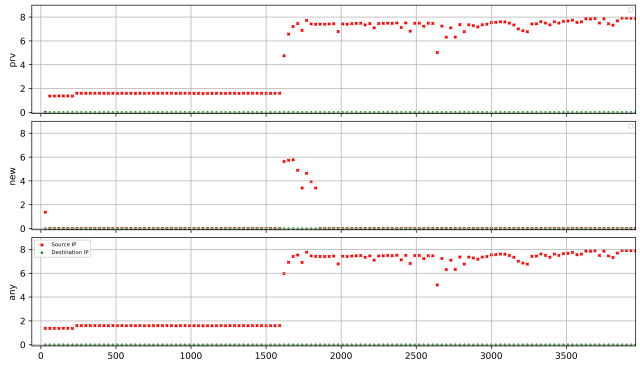

Fig. 10. Entropy charts of category 4 (100+ packets partial flows) with time window 30 seconds.

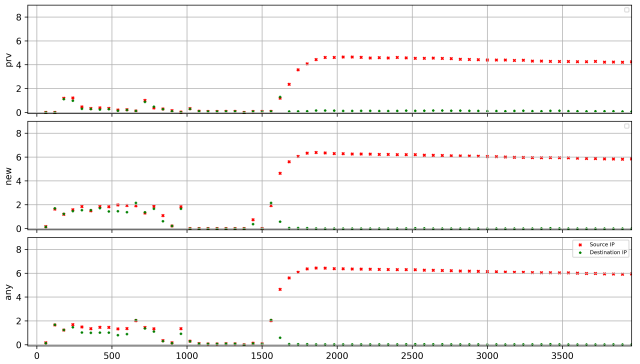

Fig. 11. Entropy charts of category 1 (1 packet partial flows) with time window 60 seconds.

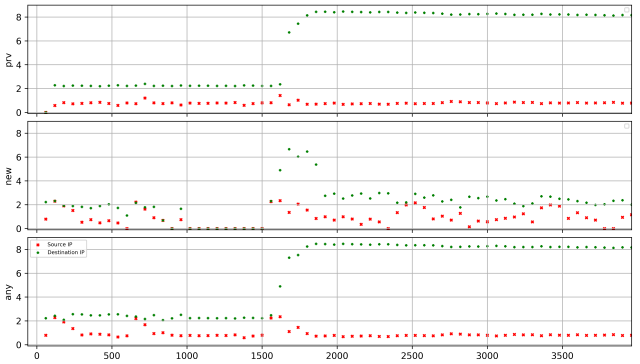

Fig. 12. Entropy charts of category 2 (2-9 packets partial flows) with time window 60 seconds.

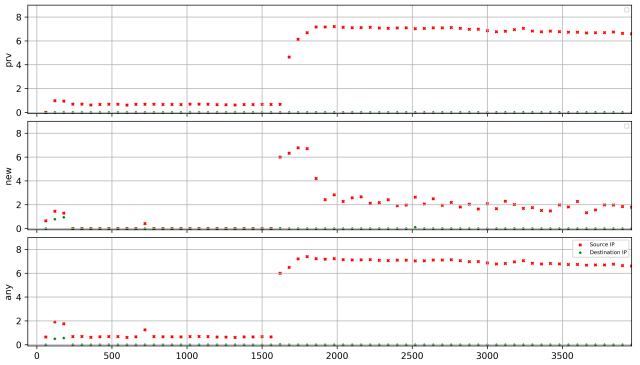

Fig. 13. Entropy charts of category 3 (10-99 packets partial flows) with time window 60 seconds.

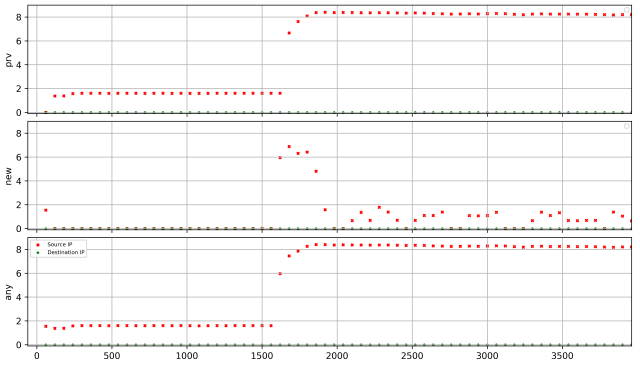

Fig. 14. Entropy charts of category 4 (100+ packets partial flows) with time window 60 seconds.

REFERENCES

[1] The CAIDA UCSD "DDoS Attack 2007" Dataset. http://www.caida.org/data/passive/ddos-20070804_dataset.xml. Accessed: 2019-01-30.

[2] Glenn Carl, George Kesidis, Richard R Brooks, and Suresh Rai. Denial-of-service attack-detection techniques. *IEEE Internet computing*, 10(1):82–89, 2006.

[3] Jisa David and Ciza Thomas. DDoS attack detection using fast entropy approach on flow-based network traffic. *Procedia Computer Science*, 50:30–36, 2015.

[4] Laura Feinstein, Dan Schnackenberg, Ravindra Balupari, and Darrell Kindred. Statistical approaches to DDoS attack detection and response. In *Proceedings DARPA information survivability conference and exposition*, volume 1, pages 303–314. IEEE, 2003.

[5] Jelena Mirkovic and Peter Reiher. A taxonomy of DDoS attack and DDoS defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34(2):39–53, 2004.

[6] Seyed Mohammad Mousavi and Marc St-Hilaire. Early detection of DDoS attacks against SDN controllers. In *2015 International Conference on Computing, Networking and Communications (ICNC)*, pages 77–81. IEEE, 2015.

[7] Xi Qin, Tongge Xu, and Chao Wang. DDoS attack detection using flow entropy and clustering technique. In *2015 11th International Conference on Computational Intelligence and Security (CIS)*, pages 412–415. IEEE, 2015.

[8] Rui Wang, Zhiping Jia, and Lei Ju. An entropy-based distributed DDoS detection mechanism in software-defined networking. In *2015 IEEE Trustcom/BigDataSE/ISPA*, volume 1, pages 310–317. IEEE, 2015.

[9] Shui Yu and Wanlei Zhou. Entropy-based collaborative detection of DDoS attacks on community networks. In *2008 Sixth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 566–571. IEEE, 2008.

[10] Shui Yu, Wanlei Zhou, and Robin Doss. Information theory based detection against network behavior mimicking DDoS attacks. *IEEE Communications Letters*, 12(4):318–321, 2008.