Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 2458-9403
Vol. 11 Issue 5, May - 2024

Finding Significant Time Intervals In Network Anomaly Detection Using Internet Traffic Characterization

Hoan Le,
School of Information Technology,
Mien Dong Innovative Technology University (Vietnam) & NKUST (Kaohsiung, Taiwan),
Dong Nai, Vietnam,
hoanl@mit.vn & vietnam.lehoan@gmail.com

***Abstract*—Detecting traffic anomalies is a crucial task in numerous network management systems. One significant approach that has emerged involves using statistical analysis to identify a specific class of Internet traffic anomalies. This method leverages a mathematical model of equilibrium to examine strongly correlated flows, which change in a way that reveals this class of anomalies. Notably, this approach identifies anomalies caused by large sets of correlated flows without needing to train a model on historical data. In this paper, we propose a new scheme (called sTime1) to identify significant time intervals by having of a large enough flows according to Gaussian distribution and a good enough error rate under certain statistical assumption. Experiments on real traffic traces demonstrate that the sTime1 scheme can enhance performance of detection by improving the number of detected anomalies as well as reducing the detection time in compare with other approaches.**

Keywords—Anomaly, Anomaly detection, Gaussian distribution, Time Interval, Traffic.

I. INTRODUCTION

Anomaly detection are widely used in network monitoring and network security application [6,17]. Traffic anomaly is a data point that significantly deviates from normal pattern of network traffic data [1,6,9,16]. The goal of traffic anomaly detection is to monitor traffic and flag an alarm whenever some abnormal events happen. Anomaly detection algorithms normally require building a statistical model of normal traffic and defining an anomaly as a deviation from normal [5,7,8]. In these techniques, traffic data are aggregated into one or more time series, example flow counts in fixed-sized time series. Next, the time series is compared to a pre-selected model of normal traffic behavior and an anomaly is flagged whenever the observed traffic deviates from the model. However, besides the computational overhead of periodically re-training the model.

Another approach of anomaly detection is based on the flow changing model of equilibrium [2]. This model is developed based on an empirical observation that the average volume change across the flows in a link is close to zero. This flow equilibrium property holds if all the flows are nearly independent and stationary [4,11,18] and it is violated by traffic changes caused by several small and correlated flows.

In this paper, we are interested in anomaly detection method within time intervals to detect anomalies as soon as possible and in real time. A time interval is said to contain an anomaly whenever the measured value falls outside the meaning value. Besides, we are also interested in relationship of flow counts, time intervals and anomaly count in network anomaly detection by statistical method.

Real-time anomaly detection needs to analyze traffic data in correspondent time intervals with a view to provide a quick and possibly warning of ongoings traffic anomalies. The determination of the initial time interval (t_1) and significant time intervals (*sti*) are very important, as the basis for verifying anomalies when compared with threshold, to shorten computation the time and ensure flow sets is Gaussian distribution with two assumptions on empirical properties of traffic flow properties. Our approach will improve the model by these points. Besides, the performance of the model depends on initiative thresholds and guarantees the needed flows as well as maximize anomalies according to *sti*.

We present in this work *sTime1* detection algorithm to define fast significant minimum time intervals and measured flows under different time intervals to define anomalies as well as appropriate significant time intervals detecting anomalies by comparing with other detectors [10,14,15]. By this way, the scheme can show distribution of anomalies in time interval in time series and number of detected anomalies is detected better. Our improvement by finding *sti* will enhance effect of anomaly detection in real-time and can monitor of anomalies changes in each day. The method is quite simple, low complexity and fast computing time. This is a single method can find many different types of events that without knowing them in advance.

The rest of paper is organized as follows. In section II, we present the problem definition. Our solution will be described in Section III. We present in Section IV the experimental results on real data traces from WIDE project. Section V finally concludes this paper.

II. PROBLEM DEFINITION AND BASIC METHOD

This section outlines the problem of anomaly detection in network traffic and introduces a basis anomaly detection method proposed by Silvera et al. [2].

II.1 Basic parameters of traffic flows

A packet is described by following characteristics:

- TimeStamp: time label of a packet in traces;
- S_IP: Source IP address of packets;
- D_IP: Destination IP address of packets;
- S_Port: Source Port of packets;
- D_Port: Destination Port of packets;
- Protocol: Protocol of packets.

A traffic flow is defined as a serie of packets that share the same values for a given set of traffic features (S_IP, D_IP, S_Port, D_Port and Protocol).

II.2 Model of anomaly detection

a. Assumption

The model of normal traffic behavior in a link lies on two assumptions as follows [4,18].

(1) Flows are modeled by stationary processes. Since the capacity of a link is limited, the total volume of all flows can not always increase. If one flow increase its volume then other flows on the link must decrease their volume to compensate. Stationarity is heavily dependent on timescales in which we observe flows, like the size of time intervals. It is well known that traffic exhibits strong non-stationary behaviors over long timescales, including daily and weekly cycles, and long-term trends. However, several works have shown that, at short timescales, traffic can be well modeled by stationary processes [11,18].

(2) Flows are independent of each others. In fact, several flows are weakly dependent [12]. According to the law of large number, flows can be independent of each othes. Because independence does not hold strictly, small correlations between flows become insignificant compared with the randomness in large traffic aggregates. Thus, the convergence of average of volume change in aggregate time to zero may be slower, requiring a larger number of flows according to the law of large numbers [12].

b. Model

Time is divided into fixed and small interval. Given a flow *f*, the volume of this flow is the number of packets in the flow during corresponding time interval *i*, denoted by $x_{f,i}$. The volume change of flow *f* in two consecutive times is denoted by $VC_{f,i} = x_{f,i+1} - x_{f,i}$.

Given *F* flows, the Average Volume Change *(AVC$_i$)* and the Standard Deviation *(SD$_i$)* between consecutive times *i* and *i + 1* are given by:

$$AVC_i = \sum_{f=1}^{F} \frac{VC_i}{F} \quad (1)$$

$$SD_i = \left[\sum_{f=1}^{F} \frac{(VC_i - AVC_i)^2}{F-1} \right]^{1/2} \quad (2)$$

$$rSD_i = \frac{SD_i}{F} \quad (3)$$

It is observed that AVC_i and relative SD_i (rSD_i) on a link tend to zero when the number of flows *F* in a time interval is large enough. Anomaly Detection Value (*ADV)* is defined as follows.

$$ADV = AVC_i \sqrt{F} / SD_i \quad (4)$$

For two above assumptions and a large number *F*, VC_i is zero mean and independent, identically distributed (*i.i.d*) random variable. AVC_i and *ADV* follow a standard Gaussian distribution random variable. *K(p)* is the percentile of the standard normal distribution and rSD_i are parameters for assessing error of AVC_i to threshold *K(p)*.

SD_i has a *(1-p)* Confidence Interval (*CI)* given by the central limited theorem as follows.

$$CI = [AVC_i - K(p)SD_i / \sqrt{F}, AVC_i + K(p)SD_i / \sqrt{F}] \quad (5)$$

There is an anomaly at a time interval if *CI* does not contain zero. Thus, *ADV* is a value for detecting anomaly in a time interval and flag an alarm if *ADV* is larger than *K(p)*.

III. SOLUTION OF THE PROBLEM

Based on the above analysis, our objectives for this problem are improving the performance of detection and reducing computation time (for a giving scenario on-the-fly). To achieve that goal, we define the significant time intervals to detect anomalies as much as possible by comparing with other detectors; we develop a fast algorithm (*sTime1* detection algorithm) to find minimum and maximum significant time intervals.

III.1 Significant time intervals

Significant time intervals (*sti*) are reasonable time interval values that having maximum of anomalies and minimum of overlap anomalies as well as computing time. By which:

Minimum time interval (t_{min}) is a significant time interval that having large enough flows according to Gauss distribution and to maximize of anomalies.

Maximum time interval (t_{max}) is a minimum reasonable time interval that anomalies can differentatiate between two consecutive time intervals with each step t_{min} in the overall time *T*.

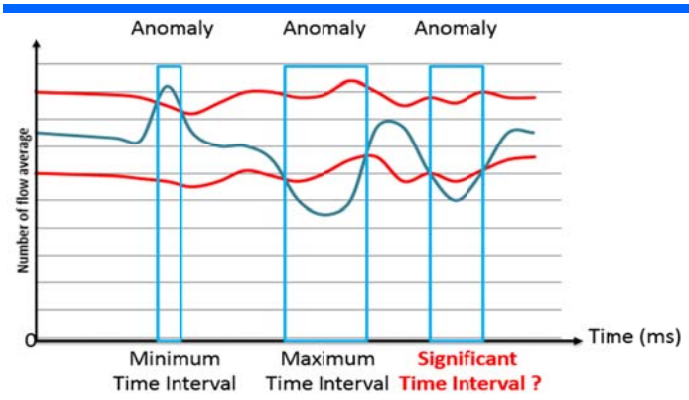

Figure 1: Defining significant time intervals, minimum time interval and maximum time interval

In many the traces, capturing times are from 5 minutes to 15 minutes. However, significant time interval *sti* is an opposite feature with capturing time. The significant time intervals are important to algorithm performance. If it is quite larger value than minimum time interval, some anomalies may be missed because many anomalies only will be happened in a very short of time. On the contrary, we have many alarms. Thus, time intervals for computing flows have reasonable values in networked anomaly detection.

III.2 Propose detection algorithm sTime1

Our s*Time1* algorithm for detecting anomalies is valid when the number of flows *F* is large. According to a rule of thumb in statistics for determining sample size [3,13] and a reasonable sample size is good with acceptable false positive rate *p*=5% to 1% or lower, we should find significant minimum time interval with at least F_0 flows so that the central limit theorem provides a good approxiamation for at least 95% confidence level and t_0=1ms. When 5% or lower significance is obtained, researchers are fairly confident that the results are real, in other words not due to chance factors alone.
We measured with different time intervals in real data sample traces with basic time is minimum time interval. The detail of detection algorithm *sTime1* as follows:

//Step 0: Input: false positive rate p, F_0 , first time interval $t=t_0$, time bin T
//Output: t_1, STI (t_{min},t_{max}) and anomalies $a(kt_1)$

//Step 1: Find initial significant time interval t_1
Define Average of Flow (*AF*) of time intervals *t*
while (*AF*<F_0 *and rSD>0.01) $t=F_0t_0/AF$*
$t_1=t$

//Step 2: Find anomaly within time intervals t_1
$a(t_1)=0$
Compute *ADV* as equation (4) for each time interval t_1 and flag an alarm if *ADV>K(p)*
$a(t_1)=a(t_1)+1$

//Step 3: Find anomaly in other time intervals $a(kt_1)$
$k=2$
Repeat step 2 with each time interval $t_2=kt_1$
$k=k+1$
Until $T/t_2>F_0$

//Step 4: Find out anomalies in traces and times, respectively
$k=1$
$rr(k)=a((k+1)t_1)/a(kt_1)$
if $rr(k)\geq 0.99$ *then* $t_{max}=kt_1$
Find out $t_1=t_{min}$, *STI* and anomalies $a(kt_1)$

III.3 Threshold K(p) by Gaussianity distribution

Likewise time intervals, *K(p)* affects to performance of the method. So *K(p)* need a reasonable value in the trace. This important since it relates directly false positive rate. Threshold controls the false positive rate, it is a probability of flagging an alarm when traffic is normal. For the number of flows *F* in a time interval is large, *AVC* has a *(1-p)* confidence level given by the central limit theorem as equation *(5)*. By the standard normal distribution, we define *K(p)* is the percentile *1-p/2* Gaussian distribution. Besides, Gaussianity depends on the number of flows in time interval, in the trace we consider time intervals from $t_1=t_{min}$ and most of the time intervals more than 400 flows so traffic Gaussianity is corresponded. With a target false positive rate of 1%, which corresponds to a detection threshold of 3 in our method. This value is used to prove again for corresponding of initial significant time interval t_1.

IV. EXPERIMENTAL EVALUATION RESULTS

In this section, we present our initial experiment result and some comparison results. We evaluate in the trace that consider only traffic on non-saturated links and using short-timescales. We test false positive rate *p=1%, t_0=1ms, k≥2*. Firstly, we find the minimum time interval t_1 in traces, then implemented algorithm to define anomalies and comparison with other detectors. The ground truth for evaluation is provided by MAWI datasets [10,14,15]. Given a combination of trace and parameter values, we compute the fraction of time intervals in the trace that are considered anomalous by our method.

IV.1 Test enviroment and applied traffic traces

We investigate traffic data from MAWI in 2001 to find initial time interval through investigating the result of average volume change. In next step to detection phase, we also identify significant time intervals and the number of anomalies in these time intervals by comparison with PGHK detector [10,14,15]. Some applied traffic trace information displayed in *Table 1* as follows.

Average	*Period 19-23/3*	*Period 16-20/4*
Time (s)	943.20	729.20
Size (MB)	200.40	200
Throughput Mbps	18.20	23.8

Packets	2997000	2997000
Flows	88726	86412
Average	*Period 19-23/5*	*Period 16-20/6*
Time (s)	494.60	691.80
Size (MB)	200	200
Throughput Mbps	21	22.4
Packets	2997000	2991000
Flows	52227	72514

Table 1: Some applied traffic trace in MAWI

IV.2 The variation of AF in different time intervals and initial significant time interval t_1

We implement *sTime1* detection algorithm with real trace traffic MAWI project to find initial time interval t_1 for minimum flow average 400 flows. Results is displayed in *Figure 2.*
To prove again time t_1 whether it is appropriate or not, we compute *rSD* is small enough for *K(p)=3*, the result is also figured out in *Figure 2.*

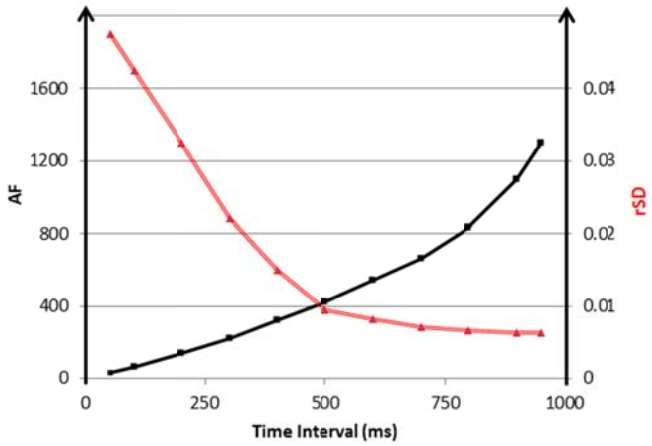

Figure 2: Defining initial significant time interval

IV.3 The variation of rSD in different time intervals and Confidene Interval in different flows

We also investigated confidence intervals in equation (5) for the variation in traffic flow average with 99% of confidence interval above as results in *Figure 3* together with the variation of *rSD*.

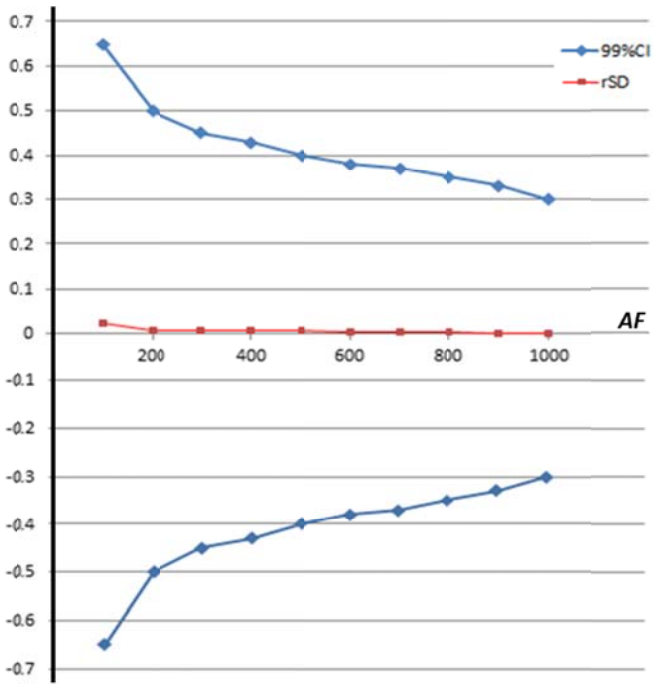

Figure 3 The variation of CI and rSD in different flows

These results can use for identifying types of anomaly in detail and flows level distribution to trigger this anomaly the quicker the better.

IV.4 Anomaly detection in different time intervals and significant time intervals

On the basis of the initial time interval t_1 and implement *sTime1* algorithm in the MAWI traffic trace in a row, we identify t_{max} as follows: we compute the anomaly rate *rr(k)* between two consecutive time interval in time *T* with each step $t_{min}=t_1$. We define t_{max} when *rr(k)*≥0.99. This *rr(k)* can satisfy that anomalies can differentatiate between two consecutive time intervals with each step t_{min} in the overall time *T*. The results show the rate of anomaly in *Figure 4.*
Thus, in the investigated trace traffic, significant time intervals are from 500ms to 5000ms. We have determined the initial significant time interval is 500ms, significant maximum time interval is 5000ms.

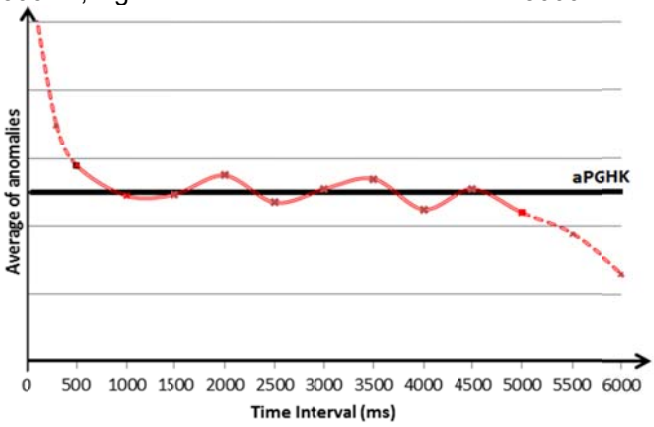

Figure 4: The significant time intervals in Internet traffic anomaly detection

In this time interval, we can detect maximum of anomalies. If time interval is smaller *sti*, we have

many alarms that overlap anomalies. Thus, we need analysis more details flows to differentiate alarms and kind of anomalies.

On the contrary, time interval is bigger *sti* then anomalies can be missed because anomalies occur in the short time. We can adjust threshold as a result, it can be overlapped anomalies. Moreover, in bigger time it can hardly to detect anomalies online.

V. CONCLUSION AND FUTURE WORK

This paper presents a novel anomaly detection technique for network traffic analysis. The method leverages empirical flow properties, enabling anomaly identification without pre-defined models of normal behavior. This unsupervised approach offers key advantages:

- **Simplicity:** Our method requires no training data, making it less susceptible to data poisoning attacks.
- **Efficiency:** We introduce the *sTime1* algorithm, which analyzes real traffic traces from the MAWI archive to identify significant time intervals. This statistically-driven approach reduces computation time and enhances detection efficiency.

Our results demonstrate that anomalies can be effectively detected by analyzing deviations in flow properties that follow a Gaussian distribution. We employ a threshold (K(p)) to distinguish between normal and anomalous behavior. Additionally, our experimental time scheme achieves anomaly detection within 120 seconds.

While the proposed method exhibits promising results, including comparable anomaly detection rates to the established PGHK detector, further investigation is necessary:

- **Anomaly Type Classification:** We currently do not delve into identifying specific types of anomalies or the flow distributions that trigger them. Future work will explore methods to assess these relationships for each anomaly.
- **Online Detection and Machine Learning Integration:** Our current approach focuses on offline analysis. We plan to investigate techniques for online anomaly detection and incorporate machine learning algorithms to potentially improve detection accuracy and identify specific anomaly types.

VI. REFERENCES

1. T.H.A. Musa and A. Bouras, “Anomaly detection: A survey”, Proceedings of sixth International Congress on Information and Communication Technology, London, 2021, https://link.springer.com/chapter/10.1007/978-981-16-2102-4_36 (last access 2024/05).
2. S. Fernando et al., “ASTUTE: Detecting a Different Class of Traffic Anomalies”, SIGCOMM’10. India, 2010.
3. R. Carmen et al., “Understanding Power and Rules of Thumb for Determining Sample Sizes”, University of Wisconsin-La Crosse, 2007.
4. C. Barakat et al., “A flow-based model for Internet backbone traffic”, In Proceedings of IMW-2002.
5. A. Lakhina et al., “Structural analysis of network traffic flows”, In Proceedings of SIGMETRICS-2004.
6. N. Moustafa, J. Hu and J. Slay, “A holistic review of Network Anomaly Detection Systems: A comprehensive survey”, Journal of Network and Computer applications, volume 128, pp. 33-55, February 2019.
7. P. Barford et al., “A Signal Analysis of Network Traffic Anomalies”, In Proceedings of IMW-2002.
8. Yarimtepe, “Anomaly Detection using network traffic characterization”, 2009.
9. Petar Cisar et.al, “A flow-based algorithm for statistic anomaly detection”, The 7th International symposium of Hungarian researchers on Computational Intelligent, pp.423-432, 2006.
10. R. Fontugne et al., “A Hough-transform-based anomaly detector with an adaptive time interval”, ACM SAC’11, 2011.
11. K. Thomas et al., “A non-stationary Poisson view of Internet traffic”, In Proceedings of INFOCOM2004, 2004.
12. W. Feller, “An introduction to probability theory and its applications”, John Wiley & Sons, 1968.
13. Ronam Conroy, http://www.beaumontethics.ie/docs/application/samplesizecalculation (last access 2024/05).
14. R. Fontugne et al., “MAWI: Combining diverse anomaly detectors for automated anomaly labeling and performance benchmarking”, CoNEXT’10 ACM. USA, 2010.
15. http://mawi.nezu.wide.ad.jp/mawi/samplepoint-B/2001 (last access 2024/05).
16. J. Barnard and C. Stryker (2023), “What is anomaly detection?” https://www.ibm.com/topics/anomaly-detection (last access 2024/05).
17. S. Gajin, “Network Traffic Anomaly Detection and Analysis - from Research to the Implementation”, BISEC’22, Serbia, 2022.
18. J. Cao et al., “On the nonstationarity of Internet traffic”, In Proceedings of SIGMETRICS, 2001.