Performance evaluation of sketch schemes on traffic anomaly detection accuracy

Masamichi Yoshioka, Takefumi Hiraguri, and Hideaki Yoshino*

Graduate School of Engineering, Nippon Institute of Technology,
4–1 Gakuendai, Miyashiro, Minamisaitama, Saitama 345–8501, Japan
a) yoshino@nit.ac.jp

Abstract: Network monitoring for high-speed networks shared by an increasing amount of traffic has become a crucial issue. Especially, traffic anomaly detection technology for high-speed networks is one of the most important issues, because of the increasingly serious nature of cyber-attacks such as worms, port scans, and DDoS. This study focuses on the use of sketch schemes as data reduction methods for traffic anomaly detection. Previous studies on sketch schemes neglected to fully clarify the impact of the sketch parameters on the anomaly detection performance. This study verified the processing time and traffic anomaly detection accuracy with a sketch as a function of two sketch parameters: the number of hash functions and hash table size. The range of the two sketch parameters was clarified by determining the processing time and F-measure, which evaluates both the false positive and false negative rates.

Keywords: traffic anomaly detection, sketch, network monitoring

Classification: Network Management/Operation

References

[1] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, “Network anomaly detection: Methods, systems and tools,” IEEE Comm. Surv. and Tutor., vol. 16, no. 1, pp. 303–336, 2014. DOI:10.1109/SURV.2013.052213.00046
[2] S. Pukkawanna and K. Fukuda, “Combining sketch and wavelet models for anomaly detection,” IEEE International Conference on Intelligent Computer Communications, Cluj-Napoca, Romania, pp. 313–319, Aug. 2010. DOI:10.1109/ICCP.2010.5606421
[3] Y. Kanda, K. Fukuda, and T. Sugawara, “Evaluation of anomaly detection based on sketch and PCA,” IEEE GLOBECOM 2010, Miami, FL, Dec. 2010. DOI: 10.1109/GLOCOM.2010.5683878
[4] “General purpose hash function algorithms,” http://www.partow.net/programming/hashfunctions.
[5] K. Cho, K. Mitsuya, and A. Kato, “Traffic data repository at the WIDE project,” USENIX 2000 FREENIX Track, Jun. 2000.
1 Introduction

The implementation of network monitoring [1] for high-speed networks shared by increasing amounts of traffic has become crucial. Especially, traffic anomaly detection technology for high-speed networks is one of the most important issues, because cyber-attacks such as those involving worms, port scans, and Distributed Denial of Service (DDoS) attacks have become more serious.

Detailed data including packet header information such as packet capture (pcap) data is necessary to detect a traffic anomaly and enable quick actions to be taken. Therefore, data reduction methods are essential for reducing the original huge amount of data generated by increasing traffic on high-speed networks. Data reduction methods for traffic anomaly detection can be categorized into three methods: sampling, filtering, and sketch.

This study focuses on sketch schemes as data reduction methods for traffic anomaly detection. Previous studies concerning sketch schemes did not clarify the impact of sketch parameters on the anomaly detection performance [2, 3]. Thus, our research aimed to evaluate the processing time and traffic anomaly detection accuracy using a sketch as a function of two sketch parameters: the number of hash functions and hash table size.

2 Overview of the sketch schemes

As shown in Fig. 1, the procedure, according to which sketch schemes are executed, consists of the following steps.

Step 1: Input the key, such as the source or destination IP address, in the hash function.

Step 2: Create the hash table by using the output hash values, and update the counter information (or number of packets) at the same time.

Step 3: If key collision occurs when creating the hash table, keys with the same hash value are connected by the chain list using the chain method.

Step 4: Apply the traffic anomaly detection algorithms to obtain counter information.

Applying the above procedure to multiple hash functions facilitates the specification of anomaly-related IP addresses. That is, the use of several hash tables to compare keys to obtain their hash functions makes it possible to determine whether the intersection of a key is an anomaly. Fig. 1 shows an example in which the IP address of key “A” is determined to be anomalous.

Sketch schemes can capture the traffic anomaly by making use of the features of hash functions: fast computation and mapping the same key to the same hash value.

Traffic anomaly detection based on a sketch has previously been proposed [2, 3].

In [2], after splitting the source IP address into keys, wavelet analysis was applied to detect traffic anomalies. Eight different hash functions were used with the hash table size based on eight sketch parameters.

In [3], after splitting the source IP address into keys, it was further split by applying another hash function to the split data, and then Principal Component
Analysis (PCA) was applied to detect traffic anomalies. In this case, eight different hash functions, with a hash table size of 1024, were used as sketch parameters.

In the above reports, the sketch parameters were fixed, and the relationship between the sketch parameters, processing time, and traffic anomaly detection accuracy was not clarified. The use of hash functions as a performance measure for the sketch has also not been clarified. Furthermore, a problem that has arisen in recent years is that the evaluation data does not consider the traffic characteristics.

3 Performance of hash functions

We firstly evaluated the performance of 11 different kinds of hash functions: RS, PJW, ELF, BKDR, SDBM, DJB, DEK, BP, FNV, AP, and JS in [4]. We used WIDE MAWI traffic data [5] as sample evaluation data. The computation environment we used was as follows: CPU: Intel(R) Core(TM) i7-3770 3.40 GHz, RAM: 32 GB, Programming language: C.

We evaluated the processing time and coefficient of variation of a different number of destination IP addresses of every hash value: 0 to 1023 using 10 seconds of data (1.1 million packets).

As a result, the processing time of each hash function was 0.676 seconds at the maximum and 0.637 seconds on average; thus, the processing time is short and almost stable. However, the coefficient of variation of five of the hash functions has a large value: for PJW and ELF it is 0.881, SDBM 0.672, DEK 0.376, and BP 3.567. On the other hand, the coefficient of variation of the other six hash functions has a small value of 0.2 or less.

Since uniformity is an important requirement for the hash function in sketch schemes, we selected six of the hash functions, i.e., RS, BKDR, DJB, FNV, AP, and JS that satisfy this condition.

4 Processing time of sketch schemes

We evaluated the processing time as a function of two sketch parameters: the number of hash functions \(N\) and the hash table size \(M\). We used \(1 \leq N \leq 6\) different hash functions, with \(32 \leq M \leq 16384\). The sample evaluation data and the computation environment were the same as those in the previous section.
The results are shown in Fig. 2. The processing time became longer as $N$ increased and shorter as $M$ increased. This is because few collisions occur when registering keys in the hash table. Conversely, if $M$ is decreased, the number of collisions when registering keys increased, and the processing time for constructing the chain also increased. In this example, since we used 10 seconds of data, the processing time would be required to be below 10 seconds. Therefore, the blue colored combination of $(N, M)$ shown in Fig. 2 represents the desirable parameters.

5 Traffic anomaly detection accuracy of sketch schemes

We finally evaluated the traffic anomaly detection accuracy of the sketch schemes based on the result obtained in the previous section. We again used $1 \leq N \leq 6$ different hash functions, this time with hash table sizes $M$ of: 2048, 4096, and 8192. As a traffic anomaly detection algorithm, we used the Bollinger-band defined by

$$B_t = \mu_t + \alpha \sigma_t,$$

where $\mu_t$ is the moving average, $\alpha$ is a constant value, and $\sigma_t$ is the standard deviation. In this study, we set $\alpha = 3$. The interval used to determine the moving average and standard deviation was set to 180 seconds.

In terms of performance measures, we used the False Positive Rate ($FPR$), False Negative Rate ($FNR$), and $F$-measure defined as

$$FPR = \frac{FP}{FP + TN},$$

$$FNR = \frac{FN}{TP + FN},$$

where $TP$: The number of times an anomaly was judged to be an anomaly, $FP$: The number of times normal was judged to be an anomaly, $FN$: The number of times an anomaly was judged to be normal, and $TN$: The number of times normal was judged to be normal.
\[ F = \frac{1}{\frac{1}{2}\left(\frac{1}{R} + \frac{1}{P}\right)} = \frac{2RP}{R + P}, \]  

where

\[ R = \frac{TP}{TP + FN}, \]  

\[ P = \frac{TP}{TP + FP}. \]  

The \( F \)-measure is a comprehensive measure that takes a value from 0 to 1 (\( F = 1 \) means the most accurate). In this evaluation, we defined a destination IP address that communicates more than 1000 packets in 10 seconds as an anomaly.

Fig. 3(a) shows the \( FPR \) and \( FNR \) results. As the number of hash functions \( N \) increases, the \( FPR \) decreases; this is a gentle tendency when the hash table size \( M \) is large. The reason for this observation is that, as \( M \) increases, the number of IP addresses registered for each hash value decreases, and only a small number of packets are regarded as being anomalous.

As \( N \) increases, the \( FNR \) becomes higher. Especially, when \( M \) is 2048, the \( FNR \) is particularly high. In terms of this factor, if one of the values of \( N \) does not lead to

![Fig. 3(a) FPR and FNR](image)

![Fig. 3(b) F-measure](image)

Fig. 3. Traffic anomaly detection accuracy
an anomaly, it is judged to be normal. On the other hand, when $M$ is 4096, and 8192, the $FNR$ is not that high because it commonly results in an anomaly on multiple hash functions.

Fig. 3(b) shows the result of the $F$-measure. When $M$ is 2048, the $F$-measure exhibits almost no change when the value of $N$ is 3 or more. Furthermore, when $M$ is 4096 and 8192, the $F$-measure can be expected to show an increasing trend even when $N$ is 6 or more.

In summary, it is desirable to set $N$ to 3 or more, and the smaller $M$ is the more accurate the results, but as the processing time is restricted as shown in the previous section, it is necessary to consider these trade-offs when making any decisions.

6 Conclusion

In this study, we focused on sketch schemes as data reduction methods, then evaluated the processing performance of the sketch and its impact on the traffic anomaly accuracy, and clarified the following.

- Increasing the number of hash functions increases the processing time, and increasing the hash table size shortens the processing time.
- Increasing the number of hash functions lowered the false positive rate, but enhanced the false negative rate.
- Increasing the hash table size reduced the false negative rate, but resulted in a higher false positive rate.
- The required number of hash functions was found to be at least three, and the hash table size needs to be determined in consideration of the processing time. Future work includes evaluation with other anomaly detection algorithms.

Acknowledgments

This research was supported by Grant-in-Aid for Scientific Research (C) Number 26330111.