Applied Mathematics & Information Sciences
An International Journal

Detecting Hidden Anomalies Using Sketch for High-speed Network Data Stream Monitoring

Aiping Li[1] , *Yi Han*[1], *Bin Zhou*[1], *Weihong Han*[1], *Yan Jia*[1]

[1]School of Computer Science, National University of Defense Technology, Changsha 410073, CHINA

Received: Jul 18, 2011; Revised May. 14, 2011; Accepted May. 16, 2011
Published online: 1 August 2012

Abstract: Monitoring network data streams in real-time to check security event become more and more important along with the rapid growth of Internet applications. The detection typically treats the traffic as a collection of flows that need to be examined for significant changes in traffic pattern (e.g., volume, number of connections). However, as link speeds and the number of flows increase, keeping per-flow state is either too expensive or too slow. We propose building compact summaries of the traffic data using the notion of sketches.In this paper, we proposed an IP address traceability network anomaly detection method at right time based on the summary data structure. In this method, the network traffic information is recorded into sketch online in every circle which is used to detect anomalies. By using EWMA forecasting model to get each circle forecast value, it computes the error sketch between the recoded value and forecast value and detects heavy network traffic change based on Mean-Standard deviation in the error sketch. The method is effective in detecting DDoS attack, scan attack. And it can trace the IP address of victim host. Evaluated by the experiment, the results show that this method takes up little computing and memory resources and is suitable for anomaly detection under the high-speed network traffic.

Keywords: Anomaly Detection, Sketch, IP Traceability, EWMA, Mean-Standard deviation.

1. Introduction

Global Internet threats are undergoing a profound transformation from attacks designed solely to disable infrastructure to those that also target people and organizations [1]. This frightening new class of attacks directly impacts the day-to-day lives of millions of people and endangers businesses around the world. For example, new attacks steal personal information that can be used to damage reputations or lead to significant financial losses. Current mitigation techniques focus on the symptoms of the problem, filtering the spam, hardening web browsers, or building applications that warn against phishing tricks. While tools such as these are important, it is also critical to monitor and manage rapid backbone network traffic, detect anomalies in real-time, trace the IP address of victim host. It is important for network managers to make a timely decision on the anomalies detection and slow down the harm of the attacks.

We study the efficient anomaly detection method under high-speed network traffic. An IP address traceability network anomaly detection method is proposed. It records the key network traffic information into summary data structure in every circle online which is used to detect anomalies. By using EWMA forecasting model to get each interval forecast value, we compute the error sketch between the observed value and forecast value. Then we construct network traffic change reference by Mean-Standard deviation on the error sketch. The method is effective in detecting DDoS attack, scan attack and so on. And it can track the IP address of victim host. Evaluated by the experiment, this method use small computing and memory resources and can be suitable for the backbone network anomaly detection.

The rest of this paper is organized as follows. Section 2 reviews the related works. The data structure and model used in the algorithm are presented is section 3. In section 4, the process of anomaly detection method is introduced. Section 5 presents key portions of the results of our extensive testing of sketch-based change detection on different large and real datasets. We finally conclude the paper in section 6.

* Corresponding author: e-mail: apli1974@gmail.com

2. Related Works

Anomies detection methods can be roughly classified as signature-based approach and statistics-based method [2]. The signature-based method detects attacks based on the properties of known anomies; any action that conforms to the pattern of known anomies is considered anomies. Many software of this classification have been developed, such as Bro, Snort and so on [2]. One limitation of this method is the requirement that the anomaly signatures be known in advance; thus it cannot be applied to identify new anomalies. Also, a malicious attacker can evade signature-based detection systems by garbling the signatures. One can see a parallel in the failure of filter-based spam fighting systems where spammers introduce random hashes in the spam messages. The statistics-based method is based on the normal behaviour of a subject, which does not require prior knowledge about the nature and properties of anomalies and therefore can be effective even for new anomalies or variants of existing anomalies. The main idea of statistics-based approach is change detection, which estimates the difference between the normal traffic model and the detected actions in the network traffic to find anomalies [5]. Our goal in this work is to come up with an efficient, accurate, and scalable change detection mechanism for detecting significant changes in massive data streams with a large number of flows.

The change detection technologies used in anomalies detection mainly contain the following several kinds: threshold value, forecast model, Markov Model, ANN [3] and so on. However, the data stream arrives quickly from the large scale network. These methods can't be used to in real-time anomalies detection under high-speed network traffic. Take the 10G bandwidth for example, data arrive at the speed of millions per second, the interval of two messages is just 32ns at worst (40 bytes per message). So we can't handle these data stream directly to detect anomalies, and need to reduce the dimension.

To deal with the large amounts of data of internet traffic, anomaly detection procedures are nowadays more and more often relying on dimensionality reduction tools. The techniques adopted in these tools can be classified as two kinds: principal component analysis (PCA) and maintain a sketch of a collection of high-dimensional data streams. PCA involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. A. Lakhina [4] propose a spatial method which takes advantage of PCA to divide network traffic into two parts, the normal part and the anomalistic part, the normal part can be forecasted, but the anomalistic part has noise, including the burr in the network traffic. Sketch is a kind of efficient summary data structure, it has small memory usage, computes and updates easily, can handle the heavy changes in real-time over high speed links. B. Krishnamurthy [6] first to detect flow-level heavy changes in massive data streams at network traffic speeds use sketch, propose a heuristic method that sets the parameter of sketch automatically. R. Schweller [5] propose an efficient reverse hashing scheme to infer the keys of culprit flows from reversible sketches, by introducing modular hashing and IP mangling techniques to improve the performance problem of the reverse sketch. G. Dewaele [7]gave a method based on sketch, which take advantage of the multi time measure character of non Gaussian marginal distribution, can detect longtime attack under low strength and very short time port scan attack.

The method proposed in this paper use sketch to summary the data stream of heavy network traffic, use EWMA model to forecast the value of next interval, compute the error sketch of the observed value and the forecasting value, then build network traffic change reference model based on error sketch. The traffic that doesn't match the reference model is anomaly. This method can reversely trace the anomalistic IP address, which can make network managers take efficient steps to break the attack and slow down the influence of intrusion.

3. Design of Anomalies Detection Method

In order to describe the method proposed in this paper, we take DDoS detection for example, regard packets having the same target IP address as a data stream, if the number of packets in the flow exceptionally remarkably increases in one interval, the target IP may be attacked by DDoS. We use sketch summary data structure to count the number of packets which have different target IP addresses arriving in a time interval, compute the error sketch of detected value and forecasting value, then build network traffic change reference model based on error sketch, if target IP address doesn't match the model, it's anomalistic target IP address. These target IP addresses may be the victim computers attacked by DDoS. In the following sections, we firstly introduce data stream model, summary data structure and EWMA forecast model, then discuss the main structure of detection framework. The discussion of the special process is given in the next section.

3.1. Data Steam Model

There are much kind of models to describe data stream, including Time Series Model, Cache Register Model and Turnstile Model [8]. The Turnstile Model is one of the most general data stream models. In this paper, we adopted the Turnstile Model which is used widely in anomalies detection. Let $L = \{a_i | i \in n\}$, be an input stream that arrive sequentially, item by item. Each item $a_i = (key_i, u_i)$ consists of a key $(key_i \in [n])$, where $[n] = \{1, ..., n\}$ and an update $u_i \in R$. Each key $key \in [n]$ is associated with a time varying signal $U[key_i]$. When a new item (key_i, u_i)

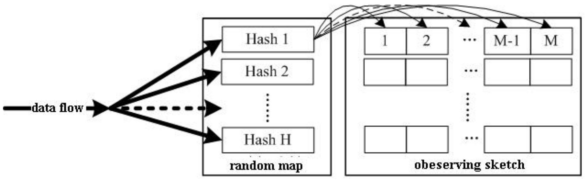


Figure 1 Illustration of summary data structure

arrives, the signal $U[key_i]$ is incremented by u_i . We will detect items whose statistical value changes a lot to find anomalies.

In the paper, data stream is designed as the network data stream in the backbone network links, item is designed as the IP packet, key is designed as the target IP address, and signal is designed as the number of packets. So, the u_i of all items are assigned to 1. We would detect the number of packets that has the same target IP address. Detecting attacks like DDoS is through finding the number of packets changes a lot.

3.2. Data Structure

The sketch records summary information of the input data stream, which consists of two parts: hash computing and summary data structure. We first discuss how to design appropriate hash functions, then introduce the summary data structure.

Lots of collisions of hash functions is the key problem to be solved, and it's impossible to avoid collision completely, but we are able to take some steps to make the probability of collision to be acceptable. So we use universal hash.

Definition 1. [Universal Hash Classes] [9] Let G be a class of functions from A to B. We say that A is Universal Hash Classes if for all x, y in A, if G satisfy: $\delta_G(x, y) \leq |G|/|B|$. That is, G is universal Hash Classes if no pair of distinct keys are mapped into the same index by more than one $|B|^t h$ of the functions. Let f be any hash function of G, f satisfy: $\forall x, y \in A$,and $x \neq y$,then $P(f(x) = f(y)) \leq {}^{1}\!/_{|B|}$.

As shown in Equation(1), p is a prime number that is bigger than the number of $[key]$, δ_i is any item in the prime number space $[p]$.

$$h(key) = \left(\sum_{i=0}^{k-1} \delta_i key^i \bmod p \right) \bmod M \tag{1}$$

The summary data structure use H*M table to store statistical value, corresponding to H*M counts. Each row of table is corresponding to a hash function, the M counts in a row mean the M buckets of hash function, and each count records the number of packets when item of input data streams is hashed to this bucket as shown in Figure. 1.

3.3. EWMA Forecasting Model

Exponentially weighted moving average (EWMA) model, sometimes also called exponential moving average (EMA), is a simple algorithm that gets the forecast data of current time from older observation data and older forecasting data in time series. If $\tilde{y}_{t-1}$ is the forecast data of time $t-1$, y_{t-1} is the observation data at time $t-1$, then the forecast data $\tilde{y}_{t-1}$ of time t is shown as in Equation(2) shows:

$$y_t = \begin{cases} \alpha y_{t-1} + (1-\alpha)\tilde{y}_{t-1} & (t > 2) \\ y_1 & (t = 2) \end{cases} \tag{2}$$

In fact, this kind of forecast data is to sum all observation data according to the weighting factors in the past time series. In this model, the weighting for each older data decreases exponentially, giving much importance to recent observations while still not discarding older observations entirely. The degree of weighting decrease is expressed as a constant smoothing factor α, a number between 0 and 1. A higher α discounts older observations faster. This model is accord with the properties of normal network traffic.

3.4. Architecture of Anomalies Detection

The architecture of anomalies detection method proposed in this paper consists of two steps. The first step is data recording and the second step is anomalies detection.

When new data arrives, it is recorded into the sketch, as shown in Figure. 1. When a new packet (key_i, u_i) arrives, the H number of hash functions compute the hash value of the packet according to key_i. We can get H number of counters which are identified by hash value. For each row is corresponded with a counter, the statistical signal value is updated according to the corresponding counter in the sketch. As shown in Equation(3) , the value of signal $S_o(t)[h][hash_h\,(key_i)]$ is incremented by u_i.

$$S_o(t)[h][hash_h\,(key_i)] += u_i\ (h \in \{0, 1, ..., H-1\}))(3)$$

Anomalies detection computes the statistical signal value of each counter in the sketch to determine the anomalistic target IP address, as shown in Figure. 2. It consists of forecast module and detection module. In the forecast module, we use EWMA forecast model to get the forecast sketch $S_f(t)[H][M]$ of current interval. In the detect module, we compute the difference of forecast sketch $S_f(t)[H][M]$ of current interval between observation sketch $S_0(t)[H][M]$ computed in the data recording step. The result is the error sketch $S_e(t)[H][M]$, which is used to determine the anomalies via statistical method.

4. Anomalies Detection Algorithm

Based on the above discussion, we propose a sketch-based anomaly detection framework with IP Address Traceability, which can costs small memory resource and satisfy the

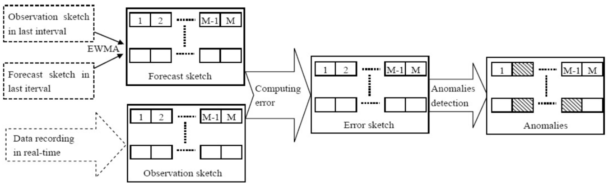


Figure 2 Architecture of Anomalies Detection Method.

real-time detected requirements over high speed links, as shown in Figure. 3. We will introduce the anomalies detection algorithm in this section.

The framework consists of sketch module (SM) and Anomalies Detection Module (ADM). The data structure used in the exchange of two modules is the sketch data structure, which has added link lists for IP traceability. The sketch module (SM) records the summary information when data stream arrives via hash computing. The result of SM is observed sketches which are used in the ADM. The forecasting component uses the observed sketches in the past intervals $S_o(t')(t' < t)$ to compute the forecast sketch $S_f(t)$ and along with it the error between the observed and forecast sketches as $S_e(t)$ via the error computation component. In this work, we explore several models commonly used in univariate time series forecasting and change detection. The first models are simple smoothing models, including the moving average (MA), EWMA, S-shaped moving average (SMA), and non-seasonal Holt-Winters (NSHW); the other are the family of ARIMA models, but we have not yet examined their application in the networking context. After the evaluation of these models (the procedure is omitted due to the space limitation), we choose the EWMA model to forecast the value of next interval.

Algorithm 1 Summarize Procedure

```
Input: the data stream, a sequence of $(key_i, u_i)$;
Output: the observed sketch of the input data stream in a time interval;
Read the data $(key_i, u_i)$;
while data is not null do
  if $t < t_0 + t_Interval$ then
    for $h = 1$ to $H$ do
      $hash_h(key_i) = (\sum_{j=0}^{k-1} \delta_j key_i^j \bmod p) \bmod M$;
      $S[h][hash_h(key_i)] += u_i \; (h \in \{0, 1, ..., H-1\})$;
      Add $key_i$ to the corresponding link list identified by $hash_h(key_i)$;
    end for
  else
    output the observation sketch to ADM;
    clear the sketch for next interval;
  end if
end while
```

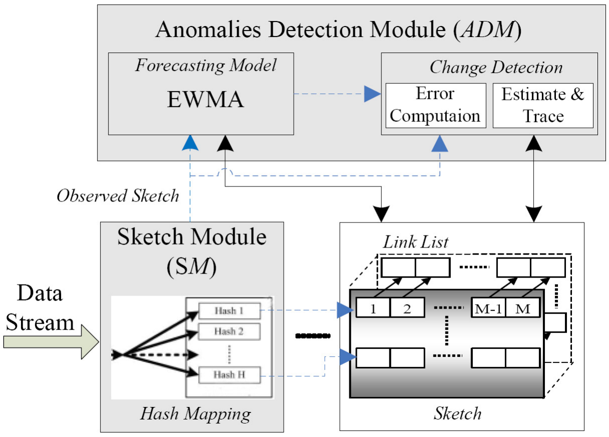


Figure 3 Architecture of sketch-based anomaly detection framework with IP Address Traceability

After constructing the forecast error sketch $S_e(t)$, the change detection component chooses an alarm threshold T_A based on the estimated second moment of $S_e(t)$. Now for any key a, the estimate component can reconstruct its forecast error in $S_e(t)$ raise an alarm whenever the estimated forecast error is above the alarm threshold T_A. Then the trace component reversely traces the input streams of keys via the link list of sketch.

We use the sketch data structure to summarize all the value of items in each interval. The pseudo code description of summarize procedure is shown in Algorithm 1.

Observation sketch data structure $S_o(t)[H][M]$ uses an array which has H-row and M-column to store statistics, each item is a signal counter, also called bucket. So observation sketch has $H * M$ buckets. The item key_i which has been hashed to each bucket is stored by a link list pointed

Algorithm 2 Anomaly Detection

```
Input: the observed sketch;
Output: the anomaly item $Key_i^{abnomal}$;
$S_f(t) = \mathrm{EWMA}(S_o(t'))$;
$S_e(t) = S_o(t) - S_f(t)$;
for $i = 1$ to $h$ do
  $E_t(i) = \sum_{j=1}^{M} |S_e(t)[i][j]| \Big/ M$;
  $V_t(i) = \sqrt{\sum_{j=1}^{M} (|S_e(t)[i][j]| - E_t(i))^2 \Big/ M}$;
  $T_A = E_t(i) + \beta * V_t(i)$;
  for $j = 1$ to $M$ do
    if $|S_e(t)[i][j]| \geq T$ then
      $U_{ij} = U_{ij} \cup List_{ij}$;
    end if
  end for
  $Key_i^{abnomal} = Key_i^{abnomal} \cap U_{ij}$;
end for
```

to the bucket. The link list is used to trace anomalistic IP address in the anomalies detection module.

As shown in Figure. 3, the anomalies detection module consists of two components. The forecasting component uses the observed sketches in the past intervals to compute the forecast sketch. The change detection component consists of error computation process, estimate and trace process. This component uses the observation sketch and forecast sketch to compute and detect anomalies.The algorithm of anomalies detection is described in Algorithm 2.

When update the forecasting value $S_f(t)$ use the $S_o(t')$ according to EWMA forecast model (line 1), we use the algorithm as shown in Equation(4). The forecasting value is updated according the EWMA model if the counter C_{ij} is normal; the forecasting value will not be updated if the counter C_{ij} is anomaly. The experiment proves this process method is more precise.

The final anomalistic target IP addresses set may be the victim computer set. We can analysis the packets of these target IP addresses to determine the type of attack.

$$S_f(t)[i][j] = \begin{cases} \begin{array}{c} \alpha S_o(t')[i][j]+ \\ (1-\alpha)S_f(t')[i][j] \end{array} & (C_{ij}\text{normal}) \\ S_f(t')[i][j] & (C_{ij}\text{abnormal}) \end{cases} \quad (4)$$

The key technique used in the anomaly detection of network traffic is the choice of right normal network traffic reference model. In this paper, we choose the mean and mean square deviation approach to build network traffic change reference model based on the error sketch of observed and forecasting value. The advantage of this model is it needs not know the advance knowledge of network traffic and distribution presuppose, while it is able to describe the change of network traffic.

5. Experimental Evaluation

In this section, we use large amounts of real Internet traffic data to evaluate and validate our approach. We first describe our datasets and the experimental parameter settings. We then introduce the results for heavy change detection.

5.1. Experimental Data

We use real trace data taken from Internet2 test network of NLANR PMA group, choose IPLS-KSCY data, this data is OC192 link data from USA Indianapolis to Kansas City, the time is 2004.8.19, every data file lasts for 10 minutes, we take 6 trace files at 2:00 .p.m orderly to be a one-hour file. The main character of the 6 trace data is as shown in Table 1.

Table 1 The main character of trace data file

Trace	Rate(Mbps)	IP packet	TCP packet
140000	625.339	51984093	43631137
141000	750.993	53690155	45028538
142000	829.982	54378499	45767584
143000	780.441	53453631	44963990
144000	834.752	52827335	45247975
145000	714.669	51155444	43740481

5.2. Experimental Parameter Settings and Results Analysis

Our change detection framework includes sketch-related parameters as well as control parameters for forecasting models. There are two sketch-related parameters: the number of hash functions (H), and the size of hash tables (M). There are three control parameters for forecasting models: the forecast model flatness coefficient α, the mean square deviation coefficient β and the detection time interval T. In this experiment, after the synthetically analysis and repetitious experiments (the choose experiments' discussion is omitted due to space limitation), we set the parameters as below:

1) H = 8, M = 1024;
2) α = 0.4;
3) β = 4;
4) T = 5 minutes.

We compare our framework with the change detection framework proposed in Ref [5]. The summary introduction of change detection framework in Ref [5] is described as below: if the change (observed sketch value vs. forecast sketch value) of a counter is bigger than a certain proportion of the total flow change, the counter is thought to be anomaly, it's an anomalistic bucket. They let D_i is the sum of the ith row in the error sketch $S_e(t)$, i.e.$D_i = \sum_{j=1}^{M} S_e(t)[i][j]$, if Equation(5) can be satisfied, then the counter C_{ij} is anomaly, otherwise it's normal. Where ϕ in Equation (5) is the proportion threshold of total change.

$$\begin{array}{c} \frac{S_e(t)[i][j]-D_i/M}{1-1/M} \geq \phi * D_i, \\ \text{i.e.}S_e(t)[i][j] \geq (\phi * D_i\,(1-1/M)+D_i/M) \end{array} \quad (5)$$

In this paper, we use the same change detection framework sketch data structure parameter we choose by experiments, including $H, M, , T$ as those in Ref [5], the proportion threshold of total change ϕ is equal 0.02.

To get a better original forecasting value, we use the mean of the statistics of first three intervals as the original forecasting value. Then we begin anomalies detection at the 4th interval.

As shown in Figure. 4, our approach can detect all the anomalistic IP addresses that can be detected in Ref [5]. In Fig.4 (a), three IP addresses change a lot, for example, datagram arrive to 10.1.130.153 is just about 200,000 from

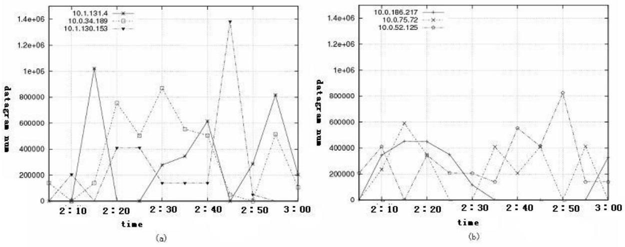


Figure 4 The character of IP address that can be detected by both two models

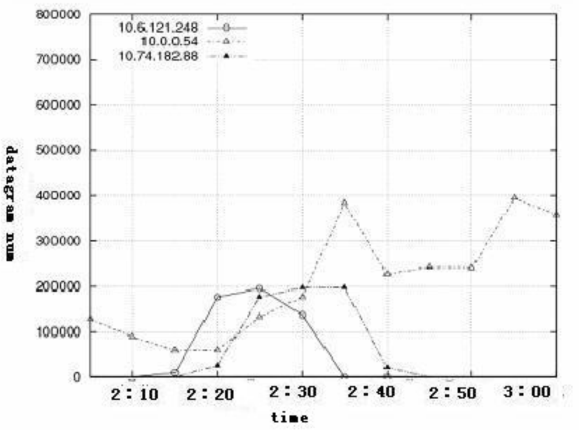


Figure 5 The character of IP address that can't be detected by the model in Ref [5]

2:35 to 2:40, but the number increases to 1,400,000 from 2:40 to 2:45, the great change means obvious flow anomalies. It may be caused by DDoS attack or flash crowd. In Fig.4 (b) IP addresses 10.0.75.72 and 10.0.52.125 also change a lot, but 10.0.186.217 changes a little, it may be a miscarriage of justice by the inaccurate of algorithm.

As shown in Figure. 5, our detection approach can also detect some extra IP addresses that can't be detected in Ref [5]. Although Ref [5] can also detect these IP addresses by reducing the proportion threshold ϕ of total change, it would cause great probability of miscarriage of justice. As shown in Figure. 7, the IP address 10.0.0.54 changes a lot from 2:30 to 2:35, it represents certainly anomaly, and so as from 2:50 to 2:55. The other two IP addresses don't change a lot; it may also be caused by the inaccurate of algorithm.

As shown in Figure. 6, take IP address 10.1.130.153 as an example, the most advantage of our approach is when the total packet number maintains smooth, while some counter of the sketch may be anomaly. As shown in Figue. 6, the total network flow changes a little, but IP address10.1.130.153 changes a lot. Detection based on sketch is better than detection based on per flow not only in memory cost, but also in computation cost; meanwhile anomalies detection based on sketch can find anomalies hiding behind lots of flat network flows.

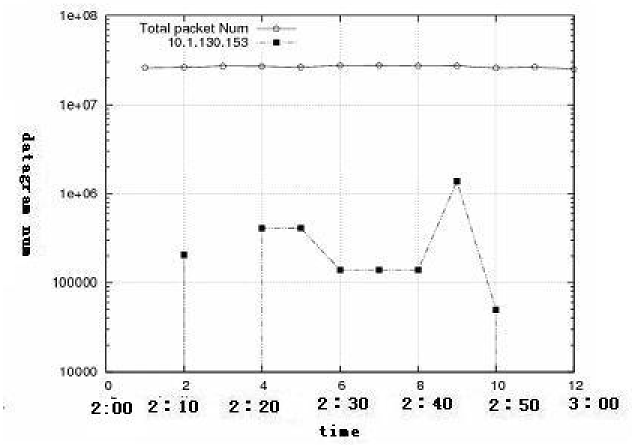


Figure 6 The anomalistic change of IP address based on totally flat network flow

6. Conclusion

A novel anomaly detection method based on sketch is proposed, it can detect attacks such as DDoS, scan attack and so on. Mean and mean square deviation are used to build change reference model. Through the reversely trace anomalistic IP address ability, network manager can take steps to break exceptional actions, let off the influence of attack. We use mean and mean square deviation to build network flow change reference model. We evaluated the accurate detection results by the experiment compared with the model of Ref [5]. What's more, it can detect more anomalies because the good properties of the anomalies detection method. Then we use examples to show the applications of great change detection based on sketch, and give an original tentative plan of distributed sketch detection.

There will be lots of applications based on sketch great change detection, although this method has been used to detect some special anomalies. How to improve the mathematic theory of this method and find a self-contained and least sketch structure to summary the network traffic is full of challenge.

Acknowledgement

The authors acknowledge the financial support by National Key Technology R&D Program (No. 2012BAH38B -04),

”863” program (No. 2010AA012505, 2011AA010702) and NSFC (No. 60933005) The author is grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] V. Paxson. Bro: A System for Detecting Network Intruders in Real-time [C], Proc. 7th conference on USENIX Security Symposium, 229-233 (1998).

[2] M. Roesch, Snort: Light weight Intrusion Detection for Networks [C], Proc. 13th USENIX conference on System administration, 229-238 (1999).

[3] R. Smith, C. Estan, S. Jha, and S. Kong. Deflating the big bang: fast and scalable deep packet inspection with extended finite automata [C], Proc. ACM SIGCOMM 2008 conference on Data communication, 207-218 (2008).

[4] A. Lakhina, M. Crovella, and C. Diot. Diagnosing network-wide traffic anomalies [C], Proc. 2004 conference on Applications, technologies, architectures, and protocols for computer communications , 219-230 (2004).

[5] R. Schweller, A. Gupta, E. Parsons, and Y. Chen. Reversible sketches for efficient and accurate change detection over network data streams [C]. Proc. 4th ACM SIGCOMM conference on Internet measurement, 207-212 (2004).

[6] B. Krishnamurthy, S. Sen, Y. Zhang, and Y. Chen. Sketch-based change detection: methods, evaluation, and applications [C]. Proc. 3rd ACM SIGCOMM conference on Internet measurement, 234-247 (2003).

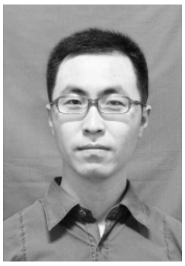
[7] G. Dewaele, K. Fukuda, P. Borgnat, P. Abry, and K. Cho. Extracting hidden anomalies using sketch and non Gaussian multiresolution statistical detection procedures [C]. Proc. 2007 workshop on Large scale attack defense, 145-152 (2007).

[8] S. Muthukrishnan. Data streams: algorithms and applications [C]. Proc. fourteenth annual ACM-SIAM symposium on Discrete algorithms, 413-413 (2003).

[9] J. Lawrence Carter and Mark N. Wegman. Universal classes of hash functions (Extended Abstract) [C]. Proc. ninth annual ACM symposium on Theory of computing, 106-112 (1977).

[10] B. Chen, R. Curtmola, G. Ateniese, and R. Burns. Remote data checking for network coding-based distributed storage systems [C]. Proc. 2010 ACM workshop on Cloud computing security workshop , 31-42 (2010).

Aiping Li is an Associate Professor at NUDT, Changsha. He obtained his PhD from National University of Defense Technology(CHINA). His major is network security, database and AI. He has published more than 20 research articles in reputed conferences and international journals of information sciences.



Yi Han is an Assistant Professor at NUDT, Changsha. He obtained his PhD from National University of Defense Technology(CHINA). His major is social computing and database. He has published more than 10 research articles in reputed conferences of information sciences.

Bin Zhou is a Professor at NUDT, Changsha. He obtained his PhD from National University of Defense Technology(CHINA). His major is social computing and database. He has published more than 30 research articles in reputed conferences of information sciences.

Weihong Han is an Associate Professor at NUDT, Changsha. She obtained her PhD from National University of Defense Technology(CHINA). Her major is network security and database. She has published more than 20 research articles in reputed conferences of information sciences.

Yan Jia is a Professor at NUDT, Changsha. She obtained her PhD from National University of Defense Technology(CHINA). Her major is network security, database and AI. She has published more than 40 research articles in reputed conferences and international journals of information sciences.