Detection and Prevention of SIP Flooding Attacks in Voice over IP Networks

Jin Tang, Yu Cheng and Yong Hao
Department of Electrical and Computer Engineering
Illinois Institute of Technology
Email: {jtang9, cheng, yhao4}@iit.edu

***Abstract*—As voice over IP (VoIP) increasingly gains popularity, traffic anomalies such as the SIP flooding attacks are also emerging and becoming into a major threat to the technology. Thus, detecting and preventing such anomalies is critical to ensure an effective VoIP system. The existing flooding detection schemes are inefficient in detecting low-rate flooding from dynamic background traffic, or may even totally fail when flooding is launched in a multi-attribute manner by simultaneously manipulating different types of SIP messages. In this paper, we develop an online scheme to detect and subsequently prevent the flooding attacks, by integrating a novel three-dimensional sketch design with the Hellinger distance (HD) detection technique. The sketch data structure summarizes the incoming SIP messages into a compact and constant-size data set based on which a separate probability distribution can be established for each SIP attribute. The HD monitors the evolution of the probability distributions and detects flooding attacks when abnormal variations are observed. The three-dimensional design equips our scheme with the advantages of high detection accuracy even for low-rate flooding, robust performance under multi-attribute flooding, and the capability of selectively discarding the offending SIP messages to prevent the attacks. Moreover, we develop an estimation freeze mechanism to protect the detection threshold from being polluted by attacks. Not only do we theoretically analyze the performance of the proposed detection and prevention techniques, but also resort to extensive simulations to thoroughly examine the performance.**

I. Introduction

Compared to the traditional public switched telephone network (PSTN), voice over IP (VoIP) is a much more economic technology, but with the tradeoff of more security concerns due to its open infrastructure mainly based on the session initiation protocol (SIP) [1] and the Internet protocol (IP). The SIP flooding attack is among the most severe of all because it is easy to launch and capable of quickly draining the resources of both networks and nodes. The attack disrupts perceived quality of service (QoS) and subsequently leads to denial of service (DoS). Furthermore, SIP is a transactional protocol and possesses multiple controlling message attributes; the flooding attacks can thus bear diverse forms and together initiate the multi-attribute attack. In order to achieve a secure VoIP system, an anomaly defense system is desired to detect the flooding attacks, classify the respective forms of them, and prevent the attacks from bringing damages to the services.

Detecting anomalies from network traffic can be modeled as distinguishing odd traffic behavior from normal behavior which is estimated based on history information. Such approaches resemble anomaly detection in the subject of statistics [2], where measurements of the investigated data form a time series for analysis. In the case of flooding attack, an intuitive choice for such measurements can be traffic volume/rate since an unreasonable volume/rate burst can well imply some malicious behavior on the network [3], [4]. However, one major limitation of volume/rate-based monitoring is that low-rate flooding can hardly be distinguished from the normal rate fluctuation due to randomness. Fortunately, besides just minor volume/rate changes, anomalies are likely to induce different probability distributions from the normal one, which reveals the presence of anomalies. The Hellinger distance (HD) [5] is a well-known metric to describe the deviation between two probability distributions, which has been used in [6] to implement a flooding detection system with good sensitivity. However, the scheme in [6] establishes a probability distribution by monitoring the relative proportions of four types of SIP messages associated with four SIP attributes within the total traffic. The detection method will become ineffective if the four attributes are proportionally flooded simultaneously. We refer to such an attack as *multi-attribute attack* in this paper. Also through investigation, we find that as there is a relatively large time difference between the BYE attribute and the other three attributes due to call holding times, dynamic normal traffic arrivals can severely undermine the effectiveness of the scheme in [6]. Moreover, the scheme does not address the important issues of how to protect the detection threshold from being polluted by attacks and how to subsequently prevent the attacks after detection.

In this paper, we develop a versatile scheme for detecting and preventing the SIP flooding attacks in VoIP networks, by integrating the sketch technique [3], [7] with the HD-based detection for a more effective and flexible solution. Sketch is capable of summarizing each of the incoming SIP messages into a compact and constant-size data set by random hash operations. Based on the sketch data set, we can establish a probability distribution for each SIP attribute independently, termed as *sketch data distribution*, which is the cornerstone of our design. Especially, we design a generic *three-dimensional sketch*: the sketch comprises multiple two-dimensional *attribute hash-tables* (one for each SIP attribute), and each attribute table consists of multiple *element hash-rows* (one associated with a different hash function). The three-

This work was supported in part by NSF grant CNS-1117687.

dimensional sketch design allows us to apply HD detection to examine the anomaly over each SIP attribute separately and therefore successfully resolve the multi-attribute attack. The multiple element hash-rows provision a voting scheme to improve detection accuracy. Also due to the separate examination on each attribute, the time difference between the attributes does not affect our scheme and we are able to maintain high detection accuracy under dynamic normal traffic arrivals. Furthermore, the multiple hash-row design with an attribute table can be leveraged to identify the offending SIP messages responsible for the flooding attack over the attribute under consideration. We can then selectively discard those messages to efficiently prevent the attack. In addition, we develop an *estimation freeze mechanism* that can protect the HD threshold estimation from being impacted by the attacks. A side benefit of the estimation freeze mechanism is that the durations of attacks can be identified.

We theoretically prove that our detection scheme can detect the flooding attack over a SIP attribute with high probability, assuming in an ideal case that the sketch data distribution associated with normal traffic could be accurately measured by a training data series. We also prove that when the HD indicates an attack, an entry in an element hash-row with a larger value than the estimated normal value must be associated with some offending SIP messages, which is the theoretical foundation for our prevention scheme design. Performance of the proposed techniques is validated through extensive simulations and comparisons to the existing SIP flooding detection solution. In summary, this paper has four-fold main contributions. (1) By exploiting the sketch technique, we decouple the probability model construction from the specific SIP attributes, which significantly enhances the flexibility of the HD-based detection. (2) We design a novel three-dimensional sketch, which equips our scheme with the advantages of high detection accuracy even for low-rate flooding attacks, robust performance under multi-attribute flooding attacks, and the capability of selectively discarding the offending SIP messages to efficiently prevent the attacks. (3) An estimation freeze mechanism is developed to protect the detection threshold from being impacted by attacks and determine the attack durations. (4) We thoroughly examine the performance of the proposed techniques through theoretical analysis and computer simulations.

The remainder of the paper is organized as follows. Section II reviews more related work. Section III describes the system model. Section IV presents the proposed SIP flooding detection and prevention scheme. Section V gives the performance evaluation results. Section VI provides discussions on related issues. Section VII concludes the paper.

II. Related Work

In the context of anomaly detection, several studies are based on the classic time series forecasting analysis and outlier detection [8]. Sketch [7] is a technique to summarize high dimensional data and provide scalable and flexible input to the time series forecasting model. Krishnamurthy et al. [3] utilize sketch in detecting behavior changes. However, their approach is based on the traffic volume, and requires the operation of retrieving data values for given keys from sketch even in the normal condition. This can incur relatively high computational cost. In our scheme, we do not perform such operation.

Using the destination addresses to profile traffic is a common approach to address the DoS problem [9], [10]; even though the attackers can be distributed, their target is concentrated on the victim addresses. This causes the traffic at destination addresses to significantly deviate from the normal condition and thus the attack will be detected. However, such an approach is not practical in the SIP case as the victim of flooding is usually a proxy server. The messages can be sent to the proxy server no matter what addresses are in the SIP destination header. In our work, we use the source addresses to profile traffic. This allows us to both detect the flooding attacks and identify the offending messages efficiently.

Surveys of the SIP security issues can be found in [14], [15]. The schemes presented in [11], [12] work effectively to detect SIP flooding. In their work, SIP transactional models are built to detect deviations from normal behaviors. However, these schemes are customized specifically to the SIP protocol suite and can not be easily generalized to other flooding detection cases. Whereas in our scheme, we can use the attributes associated with protocols other than SIP as keys to profile traffic and have a generic method to detect flooding attacks.

III. System Model

A. SIP-based VoIP

VoIP [16] utilizes SIP [1] as the application-layer signaling protocol to establish, manage and terminate communication sessions. At the transport layer, SIP normally favors the user datagram protocol (UDP) over the transmission control protocol (TCP) due to the simplicity of UDP and the connection-oriented nature of SIP itself. There are three basic components in a SIP environment, which are user agent client (UAC), user agent server (UAS) and SIP proxy server. These components are identified using the SIP address, which has a similar form to an email address, typically containing a username and a host name, e.g., "sip:alice@iit.edu". Messages are exchanged between these components to perform ordinary SIP operations.

The SIP messages used to establish and terminate sessions are basically INVITE, 200 OK, ACK and BYE. They are also called the SIP methods or attributes. A UAC initiates a SIP session by sending out an INVITE. Intermediate proxies look over the destination SIP address in the message and forward it to the destined UAS who will respond with a 200 OK. An ACK message then finishes the three-way handshake to establish the session and media will go directly between the UAC and the UAS. When the session is finished after some time (call holding time), it will be terminated by a BYE message from either of the calling parties.

B. Threat Model

SIP is vulnerable to network anomalies such as the flooding attacks. These attacks can be easily mounted by utilizing various SIP traffic generators openly available on the Internet, e.g., SIPp [13]. The victim SIP proxy servers can be overwhelmed or even crushed by a large number of SIP messages within a short period of time.

SIP utilizes multiple methods/attributes to manage sessions. This provides possibilities for the attackers to take advantage of the vulnerabilities of these attributes to launch different forms of SIP flooding attacks. We describe some of these attacks below. We see that a general detection/prevention system is desired to defend these attacks.

1) INVITE Flooding: In this attack, thousands of INVITE messages are generated and transmitted to the victim proxy servers which can barely support all of them. Moreover, being a transactional protocol, SIP may require the intermediate proxy servers to maintain a state for each INVITE message when they are expecting the associated 200 OK. Thus the resources of these victim proxy servers could be exhausted almost in real time if the attack rate is high enough.

2) BYE Flooding: The BYE message is used to terminate SIP sessions. Therefore it can be utilized by the attackers to bring down ongoing VoIP phone calls. More severely, the attackers can just launch a brute force BYE flooding attack to prematurely tear down most ongoing sessions in a VoIP network without the knowledge of the SIP addresses of the legitimate users. Such flooding attacks will cause call drops over a big range of users immediately.

3) Multi-Attribute Flooding: Intelligent attackers can launch different forms of SIP flooding attacks together to the victim proxy servers in a distributed manner. In this case, not only will the resources of the proxy servers be exhausted, but all the ongoing sessions may also be torn down instantly at the same time, which makes the multi-attribute flooding attacks devastating to the VoIP service. Moreover, the attacks flood the four SIP attributes simultaneously and thus do not change the relative proportions of the attributes. Therefore the existing SIP flooding detection solution [6] based on observing significant deviations in such proportions will become ineffective against the multi-attribute flooding attacks.

C. Detection and Prevention System

Our flooding detection and prevention system monitors the SIP messages arriving at a proxy server. We implement it in a firewall module, which can be deployed without modifying the proxy server. The system operation is based on two techniques, sketch and Hellinger distance.

1) Sketch: The sketch data structure is a probabilistic data summarization technique. It builds compact and constant-size summaries of high dimensional data streams through random aggregation, by applying a hash function [17] to the data. Specifically, we consider that each data item consists of a key k_i and its associated value v_i, represented as $a_i = (k_i, v_i)$, for constructing a sketch. Data items whose keys are hashed to the same value will be put in the same entry in sketch and their values will be added up to obtain the value of that entry. In our scheme, we use the SIP address as the key, and the value associated with each key is set as 1 indicating one SIP attribute generated from that address.

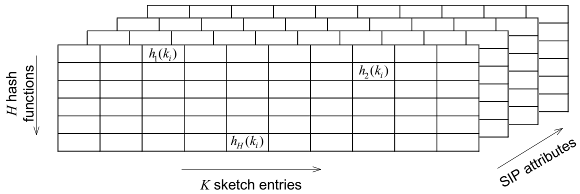

Fig. 1. Illustration of a three-dimensional sketch design.

Using sketch makes our scheme scalable. No matter how many users exist in the VoIP network, sketch is able to derive a constant-size traffic summary. More importantly, sketch allows us to construct a probability distribution based on the sketch entries, with no need to investigate the correlation among different SIP attributes as described in [6].

2) Hellinger Distance: The Hellinger distance (HD) is used to measure the distance between two probability distributions [5]. To compute HD, suppose that we have two histogram distributions on the same sample space, namely, $P = (p_1, p_2, \cdots, p_n)$ and $Q = (q_1, q_2, \cdots, q_n)$. The HD between the two distributions is defined as follow

$$H^2(P, Q) = \frac{1}{2} \sum_{i=1}^{n} (\sqrt{p_i} - \sqrt{q_i})^2. \qquad (1)$$

It is not difficult to see that the HD will be up to 1 if the two probability distributions are totally different and down to 0 if they are identical. This property provides a good approach to quantify the similarity of two data sets in either normal or anomalous situations. Recall that we aim to build an anomaly detection system which needs a statistical model to represent the normal traffic condition and raises alarms when abnormal variations are observed. The property of HD makes it well suited to this role. A low HD value implies that there is no significant deviation in the current traffic observations and a high HD is a strong indication that anomalies have happened.

IV. Detection and Prevention Scheme Design

In this section we describe our scheme to detect and prevent the SIP flooding attacks. The scheme is based on integrating the two techniques introduced in Section III, sketch and Hellinger distance.

A. Three-Dimensional Design

The SIP flooding attack can bear different forms and thus induce changes in multiple SIP attributes. We must be able to isolate the changes across the attributes, then discriminate the diverse attack forms and cope with the multi-attribute attack.

Fig. 1 gives an illustration of our three-dimensional sketch design. The sketch comprises multiple two-dimensional attribute hash-tables, each of which is built for a SIP attribute. We build four such tables for the four SIP attributes investigated. An attribute hash-table consists of H element hash-rows, each of which is associated with a different hash function and has K entries. We construct the hash functions using independent random seeds [17], and therefore they are independent from each other. The hash functions are kept secret because the seeds are not known to others. The three-dimensional sketch design allows us to separately summarize each of the SIP attributes. In the following, we first discuss how to calculate an HD based on each hash-row, and then describe the operation in the context of three-dimensional sketch.

We divide time into discrete intervals and each interval is of a constant length d. The messages associated with a certain SIP attribute under consideration is indexed as a data stream. The data stream then passes through two periods: a training period and a test period. The training period contains T consecutive time intervals and the test period is the $(T+1)$th interval. We build two sketches, one for the training period and the other for the test period. The SIP address of each message is used as key for the data to be put into the sketch. Such two sketches can generate two probability distributions for HD analysis.

Based on the training set, we obtain a sketch data distribution P. Suppose that the values of the K entries are $n_1, n_2, \cdots, n_K$, and we denote $N = \sum_{i=1}^{K} n_i$. Then we define the distribution P as

$$P = (\frac{n_1}{N}, \frac{n_2}{N}, \cdots, \frac{n_K}{N}). \tag{2}$$

Similarly, we obtain a distribution Q based on the sketch for the test period. Suppose that the values of the K entries of the test sketch are $m_1, m_2, \cdots, m_K$, with $M = \sum_{i=1}^{K} m_i$. We can have the distribution Q as

$$Q = (\frac{m_1}{M}, \frac{m_2}{M}, \cdots, \frac{m_K}{M}). \tag{3}$$

The Hellinger distance of the above two distributions is then calculated as

$$H^2(P,Q) = \frac{1}{2}\sum_{i=1}^{K}(\sqrt{\frac{n_i}{N}} - \sqrt{\frac{m_i}{M}})^2. \tag{4}$$

We monitor the data stream by tracing the HD. Assume that there is no attack in the first training set, which initially represents the normal condition. To calculate the HD, we obtain the “test” distribution Q from the current time interval and the ”training” distribution P from the immediately preceding T time intervals. We continue this operation and move the test and training periods forward respectively at each time interval, as long as the HD is smaller than a threshold. Such a sliding window mechanism better estimates the pattern of the data stream than directly analyzing two consecutive individual time intervals. It can well reflect the dynamics of the evolving traffic and smooth sudden fluctuations in normal traffic.

All the H hash-rows in an attribute hash-table independently monitor the data stream associated with a certain SIP attribute, following the same operation as described above. Similarly, in the three-dimensional sketch, the four attribute hash-tables investigate the four SIP attributes separately and are prepared for the attack detection.

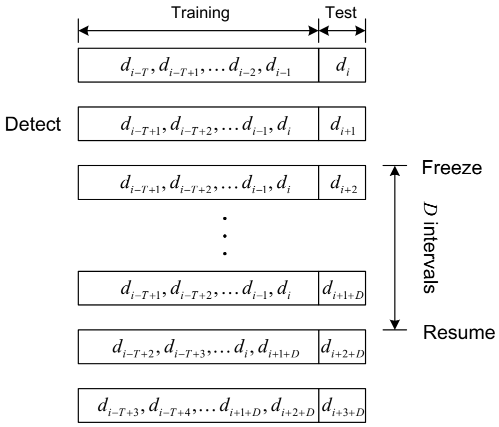
Training
Test
$d_{i-T}, d_{i-T+1}, \ldots d_{i-2}, d_{i-1}$
d_i
Detect
$d_{i-T+1}, d_{i-T+2}, \ldots d_{i-1}, d_i$
d_{i+1}
Freeze
$d_{i-T+1}, d_{i-T+2}, \ldots d_{i-1}, d_i$
d_{i+2}
D intervals
$d_{i-T+1}, d_{i-T+2}, \ldots d_{i-1}, d_i$
d_{i+1+D}
Resume
$d_{i-T+2}, d_{i-T+3}, \ldots d_i, d_{i+1+D}$
d_{i+2+D}
$d_{i-T+3}, d_{i-T+4}, \ldots d_{i+1+D}, d_{i+2+D}$
d_{i+3+D}

Fig. 2. Sliding window in estimation freeze mechanism.

B. Threshold under Attack

1) Detection Threshold: As we want to utilize HD to model the traffic behavior along time, a detection threshold is needed to reflect the normal condition and be the actual indicator of anomalies. Since normal traffic behaviors also fluctuate over time and the distribution obtained based on sketch may even not be stationary, the HD in the normal condition will be non-zero and may dynamically change. In order to properly model the behavior, we adopt the exponential weighted moving average (EWMA) method in our scheme to compute a dynamic threshold.

Let h_n denote the value of HD in the current time interval n. To smooth its fluctuation, we calculate an estimation average, H_n, of h_n as

$$H_n = (1-\alpha) \cdot H_{n-1} + \alpha \cdot h_n. \tag{5}$$

Next, to have an estimate of how much H_n deviates from h_n, we compute the current mean deviation S_n as

$$S_n = (1-\beta) \cdot S_{n-1} + \beta \cdot |H_n - h_n|. \tag{6}$$

Then given values of H_n and S_n, we derive the estimated threshold H_{n+1}^{Thre} following

$$H_{n+1}^{Thre} = \lambda \cdot H_n + \mu \cdot S_n, \tag{7}$$

where λ and μ are multiplication factors used to set a safe margin for the threshold. Due to the ability of HD to accurately monitor the difference between two probability distributions, proper values of these two parameters may greatly reduce false alarms. The parameters α, β, λ and μ are all tunable parameters in the model. We set the initial values of them according to previous research [6] and tune them in our experiments to achieve desirable detection accuracy.

2) Estimation Freeze Mechanism: When the HD obtained from a certain element hash-row exceeds the threshold, an attack detection is registered. After this, if we continue the update according to (5), (6), (7), the threshold will be polluted by the attack as the attacking traffic will be taken into account in estimating the threshold. To avoid this from happening, we freeze the threshold and keep it as a constant as long as the HD is above it. Also, to prevent the attacking traffic from entering the training set and thus keep the HD high only during attacks, we modify the sliding window mechanism. As shown in Fig. 2, after an attack detection is registered at the $(i+1)$th time interval d_{i+1}, we freeze the current training set and only let the test set proceed to the next time interval. This "one freezing one proceeding" action only ends when the HD goes below the threshold and the normal sliding window is then resumed. Overall, the above operations are illustrated in Algorithm 1, termed by us as the "estimation freeze mechanism“. As a side benefit of the mechanism, we can determine the attack duration D because the HD is above the threshold all through the attack and immediately comes down right afterwards.

Algorithm 1: Estimation Freeze Mechanism

```
Input: SIP attribute stream
Output: Duration of the anomaly $D$
$D$ = 0;
$d$ = time interval length;
anomaly starting time $t_1$ = 0;
anomaly ending time $t_2$ = 0;
if $HD$ surpasses threshold then
    $t_1$ = time of $HD$ surpassing threshold;
    $t_2$ = $t_1$;
    freeze training set;
    freeze threshold;
    while $HD >$ threshold do
        proceed test set;
        calculate $HD$ between test set and freezed
        training set;
        $t_2$ = $t_2$ + $d$;
    end
    $D$ = $t_2$ - $t_1$;
else
    proceed training set;
    proceed test set;
    update threshold;
end
end
return $D$;
```

We illustrate a comparison between two thresholds under attack in the same traffic condition in Fig. 3. The left one is estimated directly from HD without our estimation freeze mechanism whereas the right one is obtained using the mechanism. We see that without freezing the threshold goes all the way up with HD when the attack is detected. It is even much higher than HD after the detection and can not reflect the normal traffic condition. Obviously such a threshold mechanism loses track of the attack after the initial detection. On the contrary, using our estimation freeze mechanism, the threshold remains low and HD keeps high after the attack is detected. Together they also explicitly determine the duration of the attack. This provides a very clear indication of the entire attack.

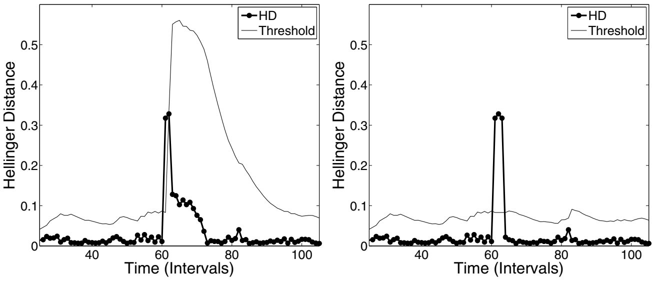

Fig. 3. Comparison of thresholds under attack.

C. Attack Detection

As described above, to actually detect possible attacks, the HD associated with a certain hash-row will be computed between the sketch data distribution constructed from the testing set and that constructed from the training set. In an ideal case, assuming that the normal probability distribution could be accurately measured from the training set, the threshold for detection can be set as 0. We have the following theorem.

Theorem *1:* A flooding attack over a SIP attribute can be detected with high probability by computing the HD between sketch data distributions, assuming that the normal probability distribution could be accurately measured from the training set.

Proof: Consider an element hash-row in the attribute hash-table under investigation. Suppose that the hash-row has K entries. The total volume of normal traffic in the testing set is M, which is distributed into the K entries according to $M = \sum_{i=1}^{K} m_i$, with m_i denoting the volume counted by the ith entry. Assume that there is a flooding attack with total volume M' added over the normal traffic in the testing set, which is distributed to K' $(\leq K)$ entries according to $M' = \sum_{i=1}^{K'} m'_i$. Let p_i denote the probability mass of entry i, and $p_i = \frac{m_i}{M}$ in the normal situation. Assume that the entry is contaminated by the attacking traffic, the probability mass will then be $p'_i = \frac{m_i+m'_i}{M+M'}$. Assume that the training set can accurately monitor the normal probability distribution and the testing set is consistent with such a distribution. The performance of the HD-based detection is then determined by the relation between p_i and p'_i as

$$|p_i - p'_i| = \left|\frac{m_i}{M} - \frac{m_i + m'_i}{M+M'}\right| = \left|\frac{m_i M' - m'_i M}{M(M+M')}\right| \\ = \left|\frac{\frac{m_i}{M} - \frac{m'_i}{M'}}{1 + M/M'}\right|. \tag{8}$$

Given a threshold of 0, the attacker needs to set the distribution of the flooding traffic exactly as the normal distribution to avoid being detected. A significant benefit of utilizing the sketch data distribution is that the hash functions used by the detection system will be kept secret to users. Therefore, it is hard for the attackers to estimate the normal sketch data distribution even if they can monitor the raw user data distribution. Furthermore, the detection system can dynamically change the sketch hash functions for a higher level of security. If the attacker attempts to guess the normal sketch data distribution $\frac{m_i}{M}$, the probability of guessing the correct value will be low, because the value of $\frac{m_i}{M}$ in a given entry can be considered as a *continuous* random variable. In other words, our detection system can detect the attack with high probability. ■

Theorem 1 demonstrates the ideal performance under accurate distribution modeling. Practically, since random aggregation of sketch brings information loss and normal traffic itself is dynamic, the normal probability distribution may change over time. Thus we can not monitor it that ideally in practice and detection accuracy may be impacted. However, the analysis shows us that attacks will indeed disturb the probability distribution obtained from the test set and as a result cause HD to rise.

In an attribute hash-table, each element hash-row registers attacks independently when its associated HD exceeds the detection threshold. To increase detection confidence and assure high accuracy, we apply a voting procedure: if at least z percent of the H rows in an attribute hash-table register attacks, a flooding attack alarm is finally raised.

D. Attack Prevention

After detecting the flooding attack, the next step is to identify the offending SIP messages and discard them to prevent the attack from reaching the proxy servers. In order to achieve this, we first identify the anomalous sketch entries that contain the offending messages in each row. Assuming that the normal probability distribution could be accurately measured from the training set, we have the following theorem.

Theorem *2:* In a flooding attack context, when the HD-based detection indicates an attack, there must exist entries in a sketch hash-row for the testing set which has a larger probability mass than that in the corresponding entry for the training set, and such entries are definitely associated with certain offending SIP messages.

Proof: In the normal situation, we assume that the normal probability distribution could be accurately measured from the training set and the testing set is consistent with the distribution. Thus, we have $\frac{m_i}{M} = \frac{n_i}{N}$. In the context under attack, the probability mass deviation in an entry i is

$$p_i' - \frac{n_i}{N} = p_i' - \frac{m_i}{M} = \frac{\frac{m_i'}{M'} - \frac{m_i}{M}}{1 + M/M'} \quad (9)$$

according to (8). When the HD detection indicates an attack, there must exist entries where $p_i' \neq \frac{n_i}{N}$. Moreover, in such entries, we must have $p_i' > \frac{n_i}{N}$ for some of them and $p_i' < \frac{n_i}{N}$ for others; otherwise the condition that $\sum_{i=1}^{K} p_i' = 1$ could not be maintained. In those entries with $p_i' > \frac{n_i}{N}$, the item associated with offending messages $\frac{m_i'}{M'}$ must exist. However, the entries with $p_i' < \frac{n_i}{N}$ may not include offending messages. The reason is that the attacking traffic might only occupy a subset of the entries in a hash-row, i.e., $K' < K$. In the leftover $K - K'$ entries, $m_i' = 0$ and offending messages are not included. ■

According to Theorem 2, we mark entries whose probability increases as possible anomalous entries. Suppose that we have p_i as the probability mass of the ith entry in one row from the training sketch set and q_i as the probability mass of the same entry from the test set. Then, if the condition

$$\sqrt{p_i} - \sqrt{q_i} < 0 \quad (10)$$

satisfies, we mark this ith entry as a suspicious entry. We use square roots of p_i and q_i as we have already obtained the value of every $\sqrt{p_i} - \sqrt{q_i}$ when calculating HD. Thus this operation would not incur much more computational cost to our scheme.

Let U_j denote the set of SIP messages that are mapped to the suspicious entries of the jth row in an attribute hash-table. We then tag these messages in U_j as offending message candidates. Certainly there will be normal SIP messages among these candidates because sketch hashes multiple users to one entry. However, since each row in a table independently performs random aggregation, offending messages and certain normal messages which are hashed to the same entry in one row are not likely to be hashed to one entry in other rows. Thus, we identify the offending message set U over all the H rows in a table through

$$U = \bigcap_{j=1}^{H} U_j. \quad (11)$$

This intersection of candidates filters out normal messages in the suspicious entries. As a result, the set U are finally believed to just include the offending SIP messages.

Once the offending messages are identified, they will be immediately discarded and only normal SIP messages can go through. This ensures that the proxy servers will only serve normal messages, and effectively prevents the attacks from reaching the proxy servers and subsequently causing damages.

V. Performance Evaluation

We evaluate the performance of the proposed SIP flooding detection and prevention scheme through simulation. The analysis focuses on the INVITE flooding case first since other SIP attributes can be addressed in a similar way. We also investigate the advantage of our scheme over the detection scheme in [6] where the effectiveness of the scheme [6] can be severely undermined by the combination effect of dynamic normal traffic arrivals and call holding time. Then we extend our discussion to the cases of distributed denial of service (DDoS) attack and multi-attribute attack.

A. Normal Traffic Behavior

In the normal condition, the average call generating rate is uniformly distributed from 25 to 75 per second with a mean of 50. The senders of the messages are chosen from 100,000 users. The numbers of messages from each sender are long tail distributed which more likely resembles the real life situation. The reason for this is that in an operational VoIP network, users do not fairly make phone calls with the same frequency. Most calls come from a relatively small number of heavy

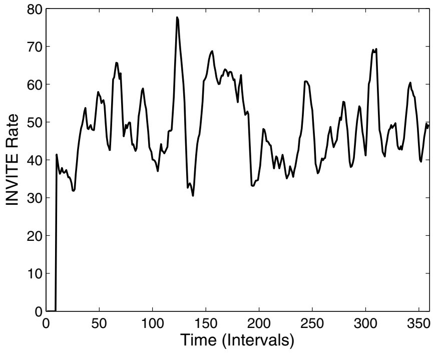

Fig. 4. Dynamic traffic rate.

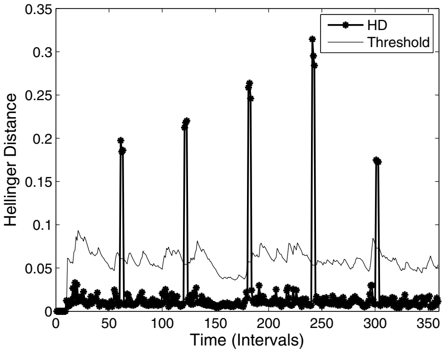

Fig. 5. Detection of flooding attacks.

users, whereas the rest lighter users who are the majority do not make phone calls that frequently. Therefore we model the user SIP addresses using the Pareto distribution [18]. Also, to properly model the BYE messages, we set the normal call holding time I to 60 seconds the same as in [6].

We parse INVITE messages from the trace data. As in [6], to achieve higher detection accuracy and lower computational cost, we set the length of a time interval d to 10 seconds. Also, as longer training set better captures the pattern of the traffic whereas shorter training set responds quicker to change, in order to find a good balance between them, the number of time intervals in a training set T is set to 10.

We build two sketches for the training set and the test set respectively and calculate the Hellinger distances between their related element hash-rows along time as described in Section IV. We find that the HDs are mostly distributed around 0.01 when we choose $K = 32$ and $H = 5$. These low HD values show the similarity of the training set and the test set when the traffic behaviors are normal.

B. Ineffectiveness of Rate Based Approach

In the flooding attack experiment, we use the normal traffic described above as background and mix it with the flooding traffic from an attacking source. In Fig. 4, we show the dynamics of traffic rates when there are five attacks of 50 INVITEs per second from a single attacker mixed with the normal traffic. The durations of the attacks are all 30 seconds. We see that there is hardly a sign of abnormal behaviors in the figure since the normal traffic itself has fluctuation as well.

C. Flooding Attack Detection and Prevention

1) Detection: We then apply our scheme to detect the same five attacks of 50 INVITEs per second as described above. We set the initial values of the parameters in the scheme according to previous research [6] and empirically get their final values as $\alpha = 0.125$, $\beta = 0.25$, $\lambda = 5$, $\mu = 1$ to achieve desirable detection accuracy. Fig. 5 shows the dynamics of the HD obtained from a hash-row and the associated threshold. The five spikes clearly identify the five flooding attacks. Other rows may not have the same detection accuracy due to different aggregations of INVITE messages, but as we set $z = 80\%$, the voting procedure finds an agreement among the five rows and raises attack alarms accurately. Also in Fig. 5, due to the "estimation freeze mechanism" applied, we can see that the HD remains high and threshold keeps constant during attack. They together precisely determine the duration of an attack, which lasts for 3 time intervals, i.e., 30 seconds. Both the HD and the threshold evolve with the dynamics of the traffic and thus preserve the ability to detect attacks online. Whereas in [6], the threshold does not react accordingly under attack and remains low as if it is always estimated from normal traffic. Compared to our threshold mechanism, theirs is not able to accurately reflect the online traffic situation.

We repeat the experiments and change the attack rates accordingly. The flooding rates vary from 15 per second to 500 per second. The purpose of choosing such a wide range is to see that despite effectively detecting high rate flooding, our scheme is even capable of identifying low rate attacks which can hide in the normal traffic and still preserves high accuracy. The detection results are shown in Table I. We can see that our scheme is able to detect the attacks with 100% accuracy when the attack rate is as low as 35 per second. Also, even when the attack rate comes down to 15 per second, we are still able to detect 88% of the attacks.

TABLE I
DETECTION RESULTS

Flooding Rate	Number of Experiments	Detection Probability
15	50	88%
35	50	100%
50	50	100%
75	50	100%
100	50	100%
500	50	100%

We then compare our scheme to the detection scheme in [6]. In their scheme, probability distributions are derived by monitoring the relative proportions of the four SIP attributes within the same period of time. However, as BYE comes after a relatively long lag, i.e., the call holding time I, compared to the other three attributes, its number within a certain period of time is correlated to the number of the other

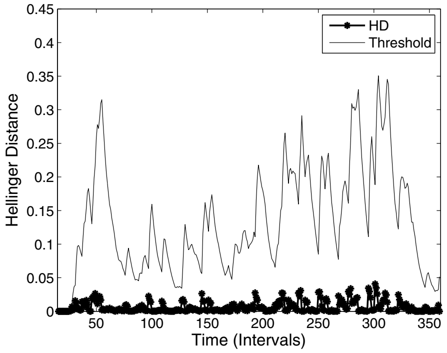

Fig. 6. Limitation of existing detection scheme [6].

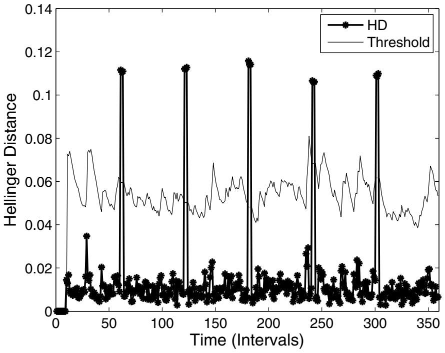

Fig. 7. Detection of DDoS attacks.

three attributes which arrived I seconds ago (there are time differences between the other three attributes, but they are much smaller compared to I). Thus if the normal traffic arrival rate is dynamic, the probability distribution derived from the relative proportions of the four SIP attributes within the same period of time will certainly have great fluctuations and result in large deviation between the training set and the test set even under the normal condition. Fig. 6 illustrates the HD and the associated threshold calculated based on the scheme in [6] from the same traffic condition where we obtain Fig. 5. We see that the five instances of attacks are not detected, as the attacks can not bring larger deviation compared to the normal traffic. We figure that for the scheme in [6] to be more effective, the standard deviation of the normal traffic rate needs be small.

Through above investigation, we learn that dynamic traffic arrivals can severely undermine the effectiveness of the scheme in [6] as BYE needs to arrive later due to call holding times. Comparatively, our scheme establishes probability distributions and detects attacks over each attribute independently, which eliminates the dependency on the correlation between different attributes. Call holding times do not affect our scheme and high detection accuracy is achieved even under dynamic traffic arrivals. Therefore, our detection scheme is more effective and flexible than the scheme in [6].

2) Prevention: For attack prevention, our scheme accurately identifies all the offending INVITE messages from the single attacker and can thus drop the messages to prevent the attacks from damaging VoIP services. There are two facts contributing to this high accuracy. First, all the offending messages are aggregated to just one suspicious entry in each element hash-row. Second, the intersection of the five suspicious entries respectively from the five element hash-rows filter out the involved normal messages and identify the offending ones.

D. DDoS Attack Detection and Prevention

1) Detection: In the case of the DDoS attack, numerous attackers initiate flooding to a proxy server simultaneously. To test our scheme against such attacks, we launch five DDoS from 300 attackers through simulation. Fig. 7 shows that the attacks cause obvious deviation in HD along time. The five spikes of HD in the figure clearly identify the attacks. We repeat the experiments varying attacker numbers and rates. The results show neither missed detection nor false alarm. The principle behind this high detection accuracy is that sketch randomly aggregates data in the H element hash-rows and deploys the same number of totally independent probability distributions. Also, the numbers of messages from each sender are long tail distributed. These make it very hard for the attackers to capture the pattern of every sketch data distribution deployed from the normal traffic in a large VoIP network.

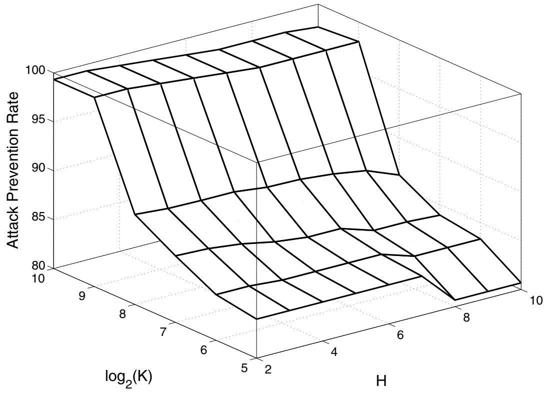

Fig. 8. DDoS prevention rate varying K and H.

2) Prevention: For the following attacker prevention, numerous offending SIP messages are identified by our scheme. However, there are still some missed offending messages that we are not able to identify. This should be due to the fact that the K used by us is smaller than the attacker number. In this case, every entry in one sketch row contains offending messages, thus by applying (10) we can only find part of them.

We then investigate the attack prevention rate for the DDoS attack. We vary the values of K and H to find out how the two parameters affect the prevention rate. The results are illustrated in Fig. 8. Clearly, when K increases, the prevention rate increases accordingly. As K becomes larger than the attacker number 300, we achieve almost 100% accuracy. Changing the value of H does not affect the result much. However, a larger H leads to lower prevention rate since more rows tend to have less consensus. Also, a small H can cause more false positives. Thus we figure that an H of 5 is a good enough value.

E. Multi-Attribute Attack

We generate distributed multi-attribute flooding attacks through simulation. There are ten attack occasions in this experiment. In each occasion, attackers flood the four SIP attributes simultaneously to a proxy server. Results of the experiment show that our scheme successfully identifies the ten attack occasions of each SIP attribute. We build three-dimensional sketch data sets to separately address each attacking attribute. Thus our scheme is able to naturally discriminate the different forms of SIP flooding no matter which attribute is being used to launch the attacks.

VI. Discussion

In this section we discuss some limitation of the proposed detection scheme and a possible solution to the issue. Flooding attacks can bear various forms to evade detection. One such special form is the stealthy flooding. Under such attack circumstances, intelligent and patient attackers start with no rush from a low initial rate. And then they will continue to periodically increase the attack rate in a slow pace. This stealthy attack does not cause sudden directly observable changes in traffic. However, it can bring damages to the network in a long time scale even though initially the attack may seem harmless.

Unfortunately, both the schemes in this paper and in [6] can not efficiently detect the stealthy flooding attack. The reason is that the attacking rate only increases slightly or even keeps the same in consecutive time intervals, thus it can hardly cause significant deviation between the two probability distributions obtained from the training set and the test set. As a result, the attack does not bring significant changes to HD over time and is only able to slowly prompt the threshold higher rather than driving HD to exceed the threshold.

To effectively detect the stealthy flooding attack, we should quickly identify the deviation from normal traffic brought by the attack. This means that we need to extract more detailed information from the directly observed traffic which only seems to change slowly. Such thoughts inspire us to resort to wavelet analysis, a signal processing technique which is able to decompose the observed traffic measures into different levels and enable observations on these more detailed levels to identify the deviation. We are currently working on this issue to detect the stealthy attack [19].

VII. Conclusion

In this paper, we propose an online VoIP flooding detection and prevention scheme by integrating two techniques, sketch and Hellinger distance. Our three-dimensional sketch design is capable of summarizing each SIP attribute separately and deploying associated probability distributions. Based on these distributions, the Hellinger distance is utilized to monitor the normal traffic behaviors and detect attacks if any abnormal variations are observed. Also, the "estimation freeze mechanism" presented shows its ability to both maintain the information about normal behavior under attack and determine the durations of the flooding attacks. A voting procedure is applied to assure the detection accuracy. Moreover, we utilize the random aggregation property of sketch and the consensus between all the rows to selectively discard the offending SIP message and subsequently prevent the attack. As we establish probability distributions for each SIP attribute independently, our scheme is fully effective to the multi-attribute attack. Our experimental results show that the proposed scheme preserves high accuracy on both attack detection and prevention. In our future work, we will further develop our scheme to improve its attack prevention rate against large number DDoS attacks and more comprehensively evaluate the scheme using extensive VoIP traffic traces. Also, we will work on to address the issue of quickly and accurately detecting the stealthy flooding attack based on the idea of wavelet analysis.

References

[1] J. Rosenberg, H. Schulzrinne and G. Camarillo, "SIP: session initiation protocol," IETF RFC 3261, Jun. 2002.

[2] V. Barnett and T. Lewis, *Outliers in Statistical Data (3rd ed.)*, Wiley, 1994.

[3] B. Krishnamurthy, S. Sen, Y. Zhang and Y. Chen, "Sketch-based change detection: methods, evaluation, and applications," in *Proc. ACM SIGCOMM Conference on Internet Measurement*, 2003.

[4] R. Schweller, Z. Li, Y. Chen, Y. Gao, A. Gupta, Y. Zhang, P. Dinda, M. Kao and G. Memik, "Reverse hashing for high-speed network monitoring: algorithms, evaluation, and applications," in *Proc. IEEE INFOCOM*, 2006.

[5] G. Yang and L. Le Cam, *Asymptotics in Statistics: Some Basic Concepts (2nd ed.)*, Wiley, 2006.

[6] H. Sengar, H. Wang, D. Wijesekera and S. Jajodia, "Detecting VoIP floods using the Hellinger distance," *IEEE Trans. Parallel Distrib. Syst.*, vol. 19, no. 6, pp. 794-805, Jun. 2008.

[7] A. Gilbert, S. Guha, P. Indyk, S. Muthukrishnan and M. Strauss, "Quicksand: quick summary and analysis of network data," DIMACS, Tech. Rep. 2001-43, 2001.

[8] C. Chen and L. Liu, "Forecasting time series with outliers," *Journal of Forecasting*, vol. 12, iss. 6, pp. 13-35, Jan. 1993.

[9] A. Lakhina, M. Crovella and C. Diot, "Mining anomalies using traffic feature distributions," in *Proc. ACM SIGCOMM*, 2005.

[10] H. Sengar, D. Wijesekera, H. Wang and S. Jajodia, "VoIP intrusion detection through interacting protocol state machines," in *Proc. IEEE International Conference on Dependable Systems and Networks*, 2006.

[11] S. Ehlert, C. Wang, T. Magedanz and D. Sisalem, "Specification-based denial-of-service detection for SIP voice-over-IP networks," in *Proc. the Third International Conference on Internet Monitoring and Protection*, 2008.

[12] E. Chen, "Detecting DoS attacks on SIP systems," in *Proc. 1st IEEE Workshop on VoIP Management and Security*, 2006.

[13] SIPp, [Online.] Available: http://sipp.sourceforge.net/.

[14] D. Sisalem, J. Kuthan and S. Ehlert, "Denial of service attacks targeting a SIP VoIP infrastructure: attack scenarios and prevention mechanisms," *IEEE Network*, vol. 20, no. 5, pp. 26-31, Sept.-Oct. 2006.

[15] D. Geneiatakis, T. Dagiuklas, G. Kambourakis, C. Lambrinoudakis, S. Gritzalis, S. Ehlert and D. Sisalem, "Survey of security vulnerabiliteis in session initiation protocol," *IEEE Communication Surveys & Tutorials*, vol. 8, no. 3, pp. 68-81, 3rd. Qtr. 2006.

[16] Y. Cheng, X. Ling, W. Song, L. Cai, W. Zhuang, and X. Shen, "A cross-layer approach for WLAN voice capacity planning," *IEEE J. Select. Areas Commun.*, vol. 25, no. 4, pp. 678-688, May 2007.

[17] M. Thorup and Y. Zhang, "Tabulation based 4-universal hashing with applications to second moment estimation," in *Proc. the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2004.

[18] W. Reed, "The Pareto, Zipf and other power laws," *Economics Letters*, vol. 74, no. 1, pp. 15-19, Dec. 2001.

[19] J. Tang and Y. Cheng, "Quick detection of stealthy SIP flooding attacks in VoIP networks," in *Proc. IEEE ICC*, 2011.