This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

SIP Flooding Attack Detection with a Multi-Dimensional Sketch Design

Jin Tang, *Member, IEEE,* Yu Cheng, *Senior Member, IEEE,*
Yong Hao and Wei Song, *Member, IEEE*

Abstract—The session initiation protocol (SIP) is widely used for controlling multimedia communication sessions over the Internet Protocol (IP). Effectively detecting a flooding attack to the SIP proxy server is critical to ensure robust multimedia communications over the Internet. 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 detection scheme for SIP flooding attacks, by integrating a novel three-dimensional sketch design with the Hellinger distance (HD) detection technique. In our sketch design, each SIP attribute is associated with a two-dimensional sketch hash table, which summarizes the incoming SIP messages into a probability distribution over the sketch table. The evolution of the probability distribution can then be monitored through HD analysis for flooding attack detection. Our three-dimensional design offers the benefit 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 from bringing damages to the network. Furthermore, we design a scheme to control the distribution of the normal traffic over the sketch. Such a design ensures our detection scheme's effectiveness even under the severe distributed denial of service (DDoS) scenario, where attackers can flood over all the sketch table entries. In this paper, we not only theoretically analyze the performance of the proposed detection techniques, but also resort to extensive computer simulations to thoroughly examine the performance.

Index Terms—Session initiation protocol, flooding attack, multi-dimensional sketch, Hellinger distance.

1 Introduction

The session initiation protocol (SIP) [1] is the signalling protocol for controlling voice and video communications over the Internet protocol (IP). SIP is however designed with an open structure vulnerable to security attacks. The SIP flooding attack is among the most severe attacks 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 their respective forms, and prevent the attacks from bringing damages to the services.

- *J. Tang was with the Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, IL 60616. He is now with AT&T Labs. E-mail: jin.tang@att.com.*
- *Y. Cheng is with the Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, IL 60616. E-mail: cheng@iit.edu.*
- *Y. Hao was with the Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, IL 60616. He is now with Juniper Networks. E-mail:yhao@juniper.net.*
- *W. Song is with the Faculty of Computer Science, University of New Brunswick, Fredericton, NB, Canada E3B 5A3. E-mail: wsong@unb.ca.*

This work was supported in part by NSF grant CNS-1117687. A preliminary version [19] of this paper was presented at IEEE INFOCOM 2012.

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 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 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

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

the effectiveness of the scheme in [6]. Moreover, the scheme in [6] 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 defense scheme for detecting the SIP flooding attacks, by integrating the sketch technique [3], [7] with the HD-based detection. 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-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 within 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 prove a *detection theorem* that our detection scheme can detect the flooding attack over a SIP attribute with a high probability, assuming an ideal case that the sketch data distribution could be accurately measured and the normal distribution is unknown to attackers. We also prove a *location theorem* 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, with the entry being termed as an *abnormal entry*. The location theorem provides the theoretical foundation for our multi-dimensional sketch design to identify the offending SIP messages. Specifically, the abnormal entries in a row can indicate a set of suspicious attackers; the intersection of the suspicious sets crossing all the sketch rows identify those messages from attackers.

The basic multi-dimensional sketch design was presented in our preliminary work [19]. In the basic design, the sketch data distribution solely depends on the hash function. Such basic detection design performs effectively in the situation where the attackers occupy a limited key space (the SIP address space in our paper) and they cannot mimic the normal sketch distribution. In this paper, we enhance the sketch design, so that the flooding attacks can still be detected even in the very severe distributed denial of service (DDoS) scenario where powerful attackers flood over all the SIP address space to mimic the normal traffic distribution, termed as *all-space attack*. Our methodology is to control the sketch distribution of normal traffic with a *sketch distribution key* (SD-key). The sketch can set a target sketch distribution, which is independent of the hash function and kept as confidential secrets to the SIP server. When a normal user applies for the SIP service, an SD-key will be calculated to bond the hash output together with the confidential sketch distribution. Later when a normal user makes a SIP call, it needs to offer its SD-key to the server based on which its sketch entry is calculated. We will show that in the online operation the SD-key design can always shape the normal traffic into the target sketch distribution and effectively detect the flooding under the all-space attack.

Performance of the proposed techniques is validated through extensive simulations and comparisons to the existing SIP flooding detection solution. In summary, this paper has five-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) An SD-key design is developed to control the sketch data distribution and effectively detect flooding attacks in the challenging all-space DDoS scenario. (5) 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 2 reviews more related work. Section 3 describes the system model. Section 4 presents the proposed SIP flooding detection scheme and conducts the theoretical performance analysis. Section 5 presents the sketch key distribution design. Section 6 gives the performance evaluation results. Section 7 provides discussions on related issues. Section 8 concludes the paper.

2 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

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

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.

Generally, intrusion detection systems are classified into two major approaches, signature based and behavior based. The signature based approach profiles known attack patterns as signatures. Detection systems in this approach raise alert if the on-going traffic patterns match the profiled signatures. For example, Kreibich et al. [9] propose a system capable of automatically generating attack signatures based on pattern-detection techniques and packet header conformance checks at multiple levels over the network protocol hierarchy. Also, popular intrusion detection tools such as Snort [10] are developed following this approach. However, a major limitation of the approach is that it is not able to detect new anomalies. Rather than profiling known attacks, the behavior based approach builds models that represent normal behaviors on the network. Alarms are raised if the observed behaviors significantly deviate from the behaviors estimated by the model. The main advantages of this approach are that *a priori* knowledge of attack strategy is not required and new anomalies unknown before can be detected. Our detection scheme in this paper adopts the behavior based approach.

Using the destination addresses to profile traffic is a common approach to address the DoS problem [11], [12]. 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 effectively detected. However, such an approach is not practical in the SIP case where the victim under the 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 field. In our work, we use the source SIP addresses to profile traffic. This allows us to both detect the flooding attacks and identify the offending SIP messages efficiently.

Surveys of the SIP security issues can be found in [17], [18]. A hash-based mechanism to protect both SIP proxy server and user agent against various SIP-based attacks is proposed in [13]. The schemes presented in [14], [15] work effectively to detect the SIP flooding DoS attacks. 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 cannot 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 thus have a generic method to detect other flooding attacks.

3 System Model

3.1 SIP basics

A voice or video communication session 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". The SIP addresses for legitimate users are normally provisioned by the voice/video service providers. Messages are exchanged between the basic 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, it will be terminated by a BYE message from either of the calling parties.

3.2 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 [16]. 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 against these attacks.

3.2.1 INVITE Flooding

In this attack, thousands of INVITE messages are generated and transmitted to the victims 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.

3.2.2 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

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

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.

3.2.3 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 also all the ongoing sessions may 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.

3.3 Sketch and Hellinger Distance

Our flooding defense 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.

3.3.1 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 [21] to the data. Specifically, we consider that each data item consists of a key k_i and its associated value v_i [20], 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. As legitimate users use SIP addresses provisioned by the service provider to communicate with each other, it is not easy for different users to use the same addresses as well as for some nomadic users to keep on changing their addresses as they want. Therefore, it is reliable to use SIP address as the hash key.

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].

3.3.2 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

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

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.

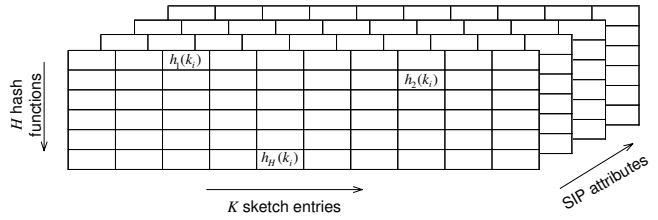


Fig. 1. Illustration of a three-dimensional sketch design.

4 Detection Scheme Design

4.1 Three-Dimensional Sketch

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 [21], 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 are indexed as a data stream. The data stream then passes

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

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

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

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

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

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

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

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.

4.2 Threshold under Attack

4.2.1 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.

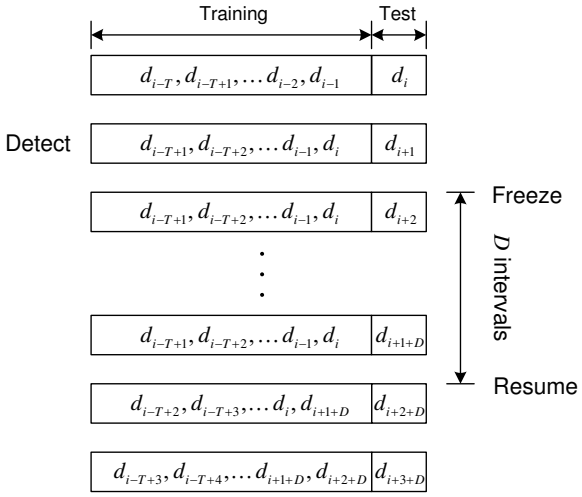


Fig. 2. Sliding window in estimation freeze mechanism.

In order to properly model the behavior, we adopt the exponential weighted moving average (EWMA) method [23] 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

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

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

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

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

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

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.

4.2.2 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},

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

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 *exceeds threshold* **then**
  t_1 = time of HD exceeding threshold;
  t_2 = t_1;
  freeze training set;
  freeze threshold;
  while $HD >$ *threshold* **do**
    test set proceeds;
    calculate HD between test set and freezed training set;
    t_2 = t_2 + d;
  end
  D = t_2 - t_1;
  else
    training set proceeds;
    test set proceeds;;
    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 cannot 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.

4.3 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 sketch probability distribution could be accurately measured from the training set, we can set the threshold for detection as 0. We have the following theorem.

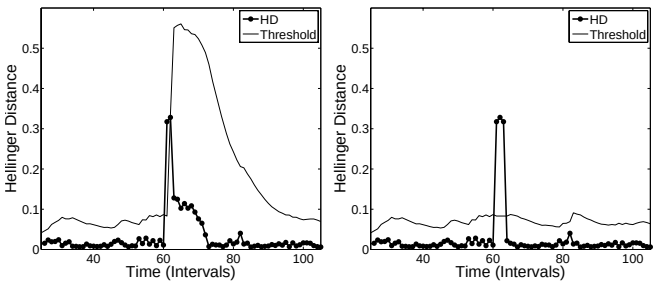


Fig. 3. Comparison of thresholds under attack.

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

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

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

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, the attackers cannot 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

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

$\frac{m_i}{M}$, 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 a high probability. □

Theorem 1 demonstrates the ideal performance under accurate distribution modeling. In practice, since random aggregation of sketch brings information loss and normal traffic itself is dynamic, the normal probability distribution may change over time. The distribution estimated by the training data is just an approximation to the real distribution. The discrepancy in estimation might lead to false alarms or missed detections. In an attribute hash-table, each element hash-row registers attacks independently when its associated HD exceeds the detection threshold. Considering the possible detection errors, in one detection circle certain rows may register attacks whereas others may not. However, if most rows agree on an attack, it is highly likely that the attack actually occurs. Correspondingly, if only a small portion of rows find one attack, we can probably consider it as a false alarm. Thus, 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.

4.4 Attack Prevention

After detecting the flooding attack, the next step is to identify the offending SIP messages and selectively discard them to prevent the attack from reaching the proxy servers and causing damages. 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:* (Location Theorem) 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

$$p'_i - \frac{n_i}{N} = p'_i - \frac{m_i}{M} = \frac{\frac{m'_i}{M'} - \frac{m_i}{M}}{1 + M/M'} \tag{9}$$

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 remaining $K - K'$ 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

$$\sqrt{p_i} - \sqrt{q_i} < 0 \tag{10}$$

satisfies, we mark this ith entry as a suspicious entry. We use square roots of p_i and q_i since we have already obtained the value of every $\sqrt{p_i} - \sqrt{q_i}$ when we calculate HD. Therefore 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

$$U = \bigcap_{j=1}^{H} U_j. \tag{11}$$

This intersection of candidates filters out normal messages in the suspicious entries. As a result, the set U is 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 also effectively prevent the attacks from reaching the proxy servers and subsequently causing damages.

5 Sketch Distribution Key Design

The detection theorem in Section 4.3 indicates that the sketch based detector is effective under the condition that the attackers cannot get information about the normal sketch distribution. The theorem however does not tell how to ensure such a condition. The basic detection scheme design presented in Section 4 works fine in scenarios where the number of attackers is not very large and the attacking traffic is only hashed into some of the sketch entries. In a large scale system with many

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

users, if the SIP address space served as input to the sketch hash functions is large enough, the output sketch distribution will become dependent only on the hash function and basically independent of the input traffic. In on-line operations, both normal and attacking flows will be processed by the sketch hash function. The traffic independent behavior implies a way for attackers to avoid being detected — flooding over a large enough space to approach the normal sketch distribution. We term such an attack as *all-space attack*. The all-space attack is indeed possible if the attackers are powerful enough to launch a large-scale DDoS attack.

In this section, we enhance the sketch design, so that the flooding attacks can still be detected even in the very severe all-space DDoS scenario. Our methodology is to control the sketch distribution of normal traffic with a *sketch distribution key* (SD-key). The sketch can set a target sketch distribution, which is independent of the hash function and kept as confidential secrets to the SIP server. When a normal user applies for the SIP service, an SD-key will be calculated to bond the hash output together with the confidential sketch distribution. Later when a normal user makes a SIP call, it needs to offer its SD-key to the server based on which its sketch entry is calculated.

Specifically, in a SIP-based multimedia application, a legitimate user needs to first register with a proxy server to get the service. The SIP proxy server sets a *target sketch data distribution* for the normal traffic. This target distribution is kept as secrets and known to the server only. Note that the target sketch distribution should not be a trivial uniform distribution. When the registration request comes, the proxy server determines the position of the sketch entry for this user according to the target sketch data distribution, denoted as a_i. At the same time, the server also calculates the hash value of the user's SIP address $b_i = h(k_i)$. Then in response to the registration request, the proxy server will send a sketch distribution key in the value of $c_i = a_i - b_i$ back to the user. When the user later sends a SIP controlling message to the proxy sever, it must include c_i in the messages. After receiving the messages, the proxy server will hash the included SIP address to obtain b_i again, and the final position of these messages in the sketch table will be determined by the value of $c_i + b_i$, which in fact equals to a_i.

As a result of the above operation, if a user l is legitimate and knows its correct associated c_l value from registration, the position of the messages from this user in sketch will simply be a_l. However, an attacker k in a DDoS attack did not go through the registration process and thus has no SD-key for the target sketch distribution. One way for the attacker is to use some random number c'_k to send to the proxy server and its position in the sketch table will be a random position $c'_k + b_k$, which very possibly gives a uniform distribution rather than the target sketch distribution. Thus, the deviation brought by the attacks from normal traffic in the sketch distribution can be identified by HD. Another way for attacking is

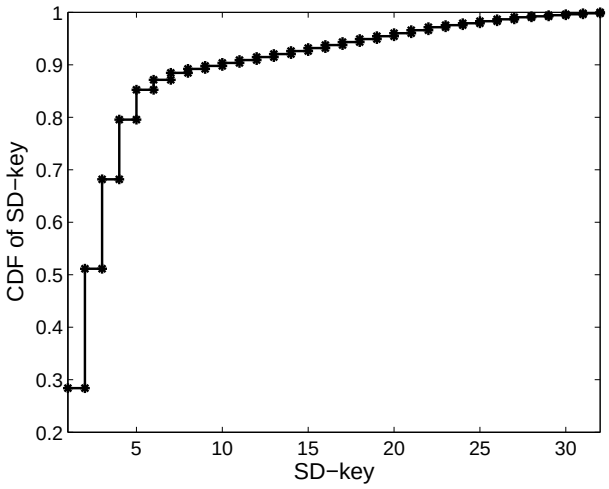


Fig. 4. SD-key distribution.

an *SD-key playback attack*. In this attack, the attackers can record those SD-keys carried by normal users and then randomly insert a copied SD-key into the attacking SIP message. Note that the SD-key has the property to bond the target sketch distribution with the normal user's SIP address together. If the playback attacker k'_i uses the SD-key from user k_i, the sketch entry for this playback attacker will be $a_i - h(k_i) + h(k'_i)$, which is again an arbitrary random value. Here, we assume that the SIP address of a normal user is protected with certain authentication schemes and cannot be spoofed by the attackers in the registration process. This ensures that only legitimate users can get the SD-keys associated with their SIP addresses in registration. If attackers do spoof SIP addresses in flooding attacks, they will not be able to evade detection due to the strength of our SD-key mechanism. The attackers must gather the SD-key for every SIP address that they spoof to get the target distribution, and they do not have the means to achieve it. The effectiveness of this enhanced sketch distribution will be demonstrated in Section 6.

6 Performance Evaluation

We evaluate the performance of the proposed SIP flooding defense scheme in this section. VoIP signaling traces are simulated and analyzed using Matlab. All detection results are based on the enhanced sketch design with the SD-key mechanism. 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 affected by the combination effect of dynamic normal traffic arrival and call holding time. Then we extend our discussion to the cases of DDoS attack and multi-attribute attack.

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

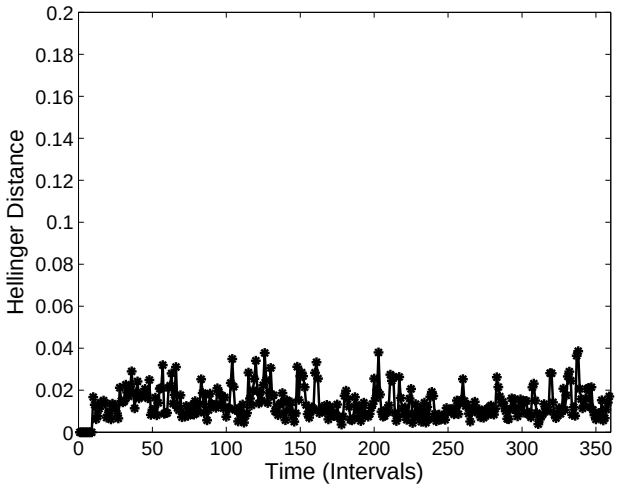


Fig. 5. Hellinger distances under normal traffic.

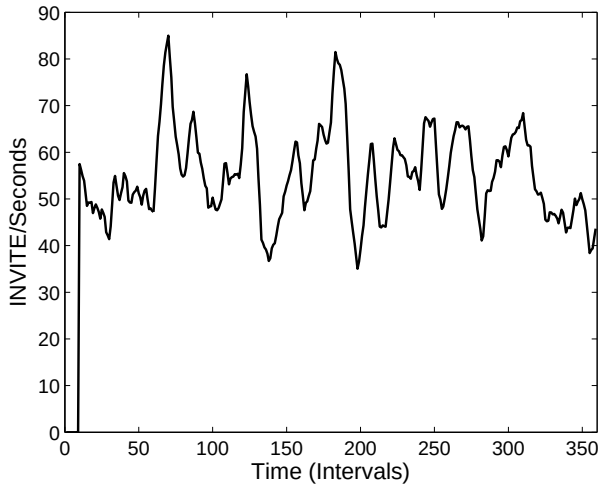


Fig. 6. Dynamic traffic rate.

6.1 Normal Traffic Behavior

In the normal condition, the average call generating rate is uniformly distributed from 25 per second to 75 per second with a mean of 50 per second. The senders of the messages are randomly chosen from 100,000 uniform users. Also, to properly model the BYE messages, we set normal call holding times to I, where I either follows a log-normal distribution to reflect the long tail characteristic of real VoIP call holding times [25] or has a constant duration of 60 seconds according to [6]. We will discuss the effect of I later.

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 a longer training set better captures the pattern of the traffic whereas a shorter training set responds quicker to change, in order to find a good balance between them, the training period is set as 10 consecutive intervals, i.e., $T = 10$.

We build two enhanced sketches for both the training set and the test set and calculate the Hellinger distances between their related element hash-rows along time as described in Section 4. The CDF of the SD-key distribution used to shape the sketch distribution of normal traffic is illustrated in Fig. 4. We will maintain this SD-key distribution for all the attacking cases in our following experiments. Also, as shown in Fig. 5, in the normal traffic condition, the HDs are mostly distributed around 0.015 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.

6.2 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. 6, we show the dynamics of traffic rates when there are five attacks of 60 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. Comparatively, we will see how our scheme responds to these attacks in next section.

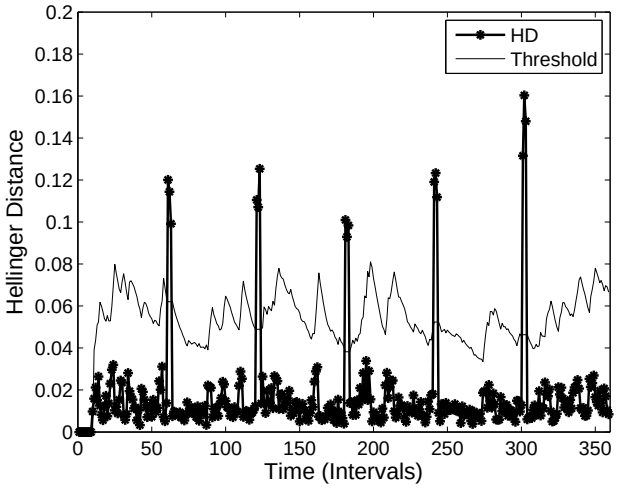


Fig. 7. Detection of flooding attacks.

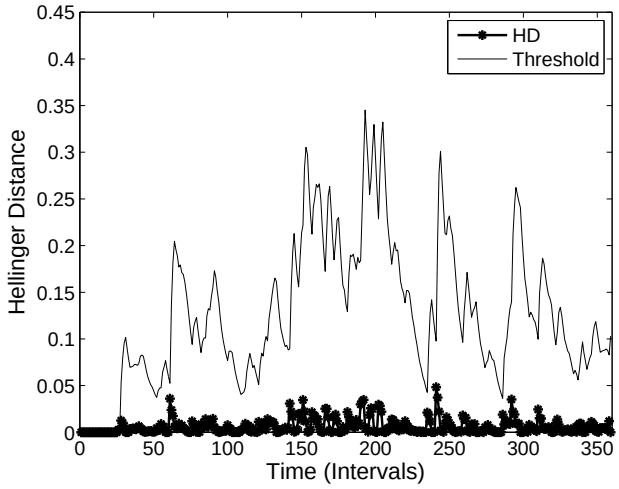
6.3 Effectiveness of Sketch-Based Detection

6.3.1 Detection

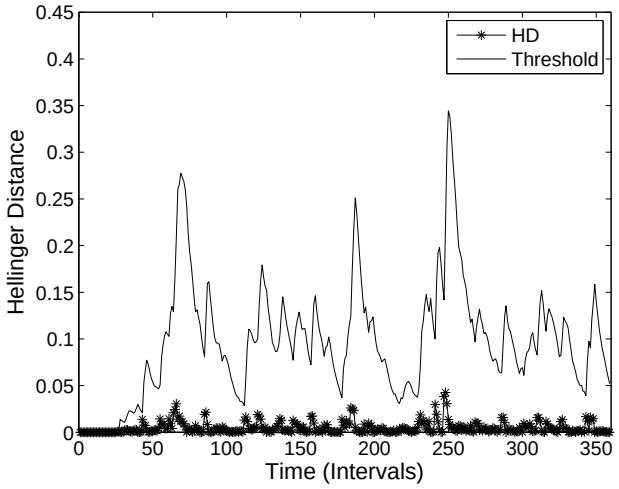
We apply our scheme to detect the same five attacks of 60 INVITEs per second as described above. We set the initial values of the parameters in the scheme according to previous research [6] and empirically get their final values as $\alpha = 0.125$, $\beta = 0.25$, $\lambda = 4$, $\mu = 1$ to achieve desirable detection accuracy. Fig. 7 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

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing



(a) Dynamic traffic rate and constant I.



(b) Dynamic traffic rate and lognormal I.

Fig. 8. Limitation of existing detection scheme [6].

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. 7, due to the "estimation freeze mechanism" applied, we can see that the HD remains high and the 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 experiment for several times and change the attack rates accordingly. The flooding rates vary from 60 per second to 500 per second. The purpose of choosing such a wide range is to see that, in addition to 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 durations of the attacks are all 30 seconds. The detection results are shown in Table 1. We can see that our scheme with enhanced sketch is able to detect the attacks with 100 percent accuracy when the attack rate is as low as 60 per second.

In [6], 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 three attributes which arrived I seconds earlier. 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. 8(a) illustrates the HD and the associated threshold calculated based on the scheme in [6] from the same traffic condition where we obtain Fig. 7. The normal average call generating rate is uniformly distributed from 25 per second to 75 per second with a mean of 50 per second and there are five attacks of 60 INVITEs per second mixed with the normal traffic. The call holding time I is set to a constant of 60 seconds in this case. We see that the HD is relatively large even when traffic is normal before and between attacks, with its mean value around 0.005. Also, the five instances of attacks are not detected, as the attacks cannot bring larger deviation compared to the normal traffic. We find that for the scheme in [6] to be more effective, the standard deviation of the normal traffic rate needs to be small. We have similar observations in Fig. 8(b), where all the setting is the same as Fig. 8(a) except that I is set to follow a lognormal distribution.

Through investigation, we learn that dynamic traffic arrivals can severely affect 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

TABLE 1
Detection Results

Flooding Rate	Number of Experiments	Detection Probability
60	50	100%
75	50	100%
100	50	100%
500	50	100%

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

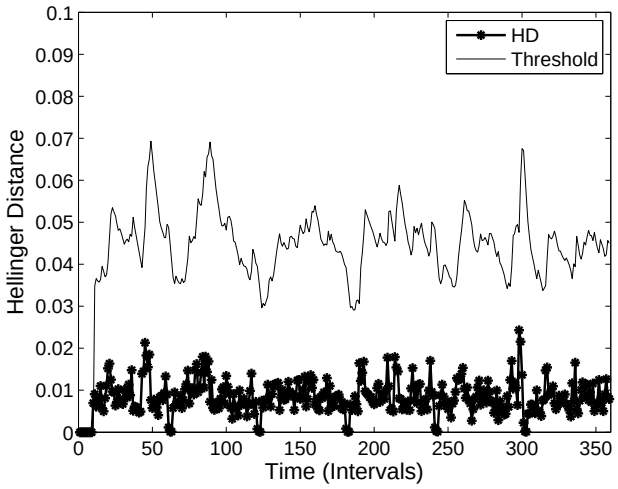


Fig. 9. Ineffectiveness of the basic scheme with hash operations but without SD-key mechanism against DDoS attacks.

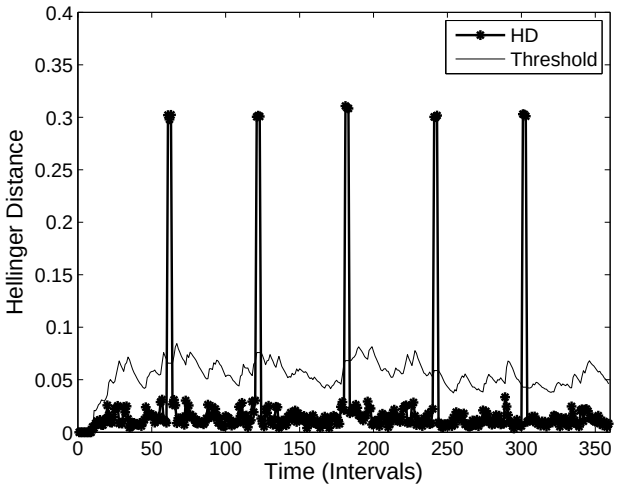


Fig. 10. Detection of the DDoS attacks.

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 robust than the scheme in [6].

6.3.2 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 the VoIP services. There is no missed identification in each attack occasion. There are two facts contributing to this high accuracy. First, all the offending messages are aggregated to just one suspicious entry in each of the element hash-rows. Second, the intersection of the five suspicious entries respectively from the five element hash-rows is enough to filter out all the involved normal messages and identify the offending ones.

6.4 DDoS Attack Detection

6.4.1 Detection

In the case of the DDoS attack, numerous attackers in a VoIP network initiate flooding to a SIP proxy server simultaneously. To test our scheme against such attacks, we launch five DDoS attackers from 300 attackers with addresses uniformly chosen from the whole address space.

Fig. 9 shows the results of the basic detection scheme as in [19] (without the SD-key mechanism) against DDoS attacks, where HD is always less than the threshold and the attacks cannot be detected. The reason for the ineffectiveness is that attack messages can occupy every entry of a sketch table just like messages from the normal users and thus bring little change to the overall traffic distribution. Fig. 10 shows the results of our scheme with

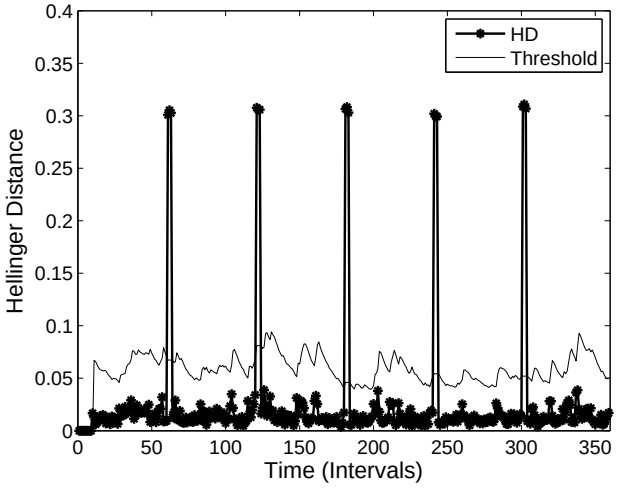


Fig. 11. Detection of the SD-key playback attack.

traffic distribution obtained from the enhanced sketch with the SD-key mechanism. The attacks cause obvious deviation in HD along time. The five spikes of HD in the figure clearly identify the attacks. We run the experiment multiple times with varying attacker numbers and rates. The results show neither missed detection nor false alarm. Note that this is achieved when we set the attribute hash-table size $K = 32$. We then further decease the value of K to test how performance may change with less overhead. As a result, we see that the false alarm rate rises to 1.67% when $K = 16$ and 7.81% when $K = 8$. The principle behind this high detection accuracy is that the enhanced sketch is able to utilize the number c_i to differentiate normal users and attackers, as the correct value of c_i is only known to normal users. This makes it difficult for the attackers to capture the pattern of every distribution deployed from the normal traffic in a large VoIP network. The detection performance drops as K gets smaller. When K

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

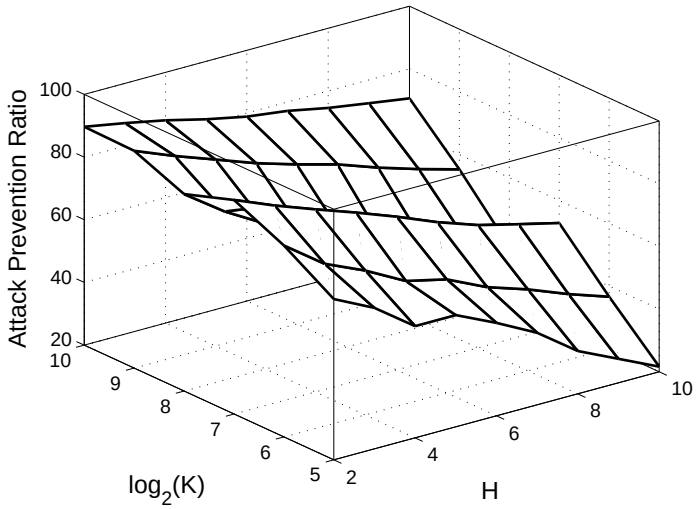


Fig. 12. DDoS prevention rate varying K and H.

is smaller, on average the probability mass distributed to one entry is larger. Let p_i be the probability that a SIP address will be hashed into the ith entry. Let X_j be an indicator, where $X_j = 1$ if the jth message is hashed into the ith entry, and $X_j = 0$ otherwise. X_j then follows the Bernoulli distribution with the probability p_i, which has the variance $var(X_j) = p_i(1 - p_i)$. It is not difficult to see that when $p_i < 0.5$ (which is normally the case over a sketch with more than two entries), the variance $var(X_j)$ increases with p_i. For online detection, given N messages, the estimated probability mass in the ith sketch entry is $(\sum_{j=1}^{N} X_j)/N$, with the variance of $(1/N)var(X_j)$ which increases with p_i. Therefore, with the larger variance due to a larger p_i, the estimated distribution is easier to deviate from the real value, and thus leads to a larger false alarm rate.

We further test the performance of our scheme against the SD-key playback attack. Fig. 11 shows our results and the attacks are clearly identified. This shows that matching legitimate c_i with the attackers' addresses does not affect the performance of our scheme.

6.4.2 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 is because we can only retrieve attacking SIP messages from the entries with a probability mass larger than the estimated value, according to the location theorem.

We then investigate the missed identification problem in the DDoS prevention operation. We vary the values of K and H to find out how they affect the prevention rate, while we always maintain the SD-key distribution for normal traffic as shown in Fig. 4. The results are illustrated in Fig. 12. We can see the trend that missed identification decreases when K increases. When we maintain the same distribution for normal traffic, the new entries in sketch resulting from the increase of K have the value of 0 in the normal condition and will be filled with attacking messages in the attacking condition. These attacking messages can easily be identified according to the location theorem. Also as K increases, there will be more attacking messages being filled in the increased new entries, resulting in more of these messages being identified. Moreover, we observe that a larger H here leads to more missed identification since more rows tend to have less consensus.

6.5 Multi-Attribute Attack

We generate distributed multi-attribute flooding attacks through simulation. There are ten attack occasions in this experiment. In each occasion, attackers send a large number of messages of the four SIP attributes, namely INVITE, 200 OK, ACK and BYE simultaneously. 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.

6.6 Computational Cost

The computational cost of the two key components in our scheme, i.e., the sketch operation and the computation of Hellinger distance, has crucial impact for real-time detection of the flooding attacks. In the enhanced scheme, the sketch operation includes computing the hash value [21] and calculating $c_i + b_i$ to determine the sketch entry position using the SD-key. Also, the computation of Hellinger distance includes the update of the thresholds as well as the estimation freeze mechanism besides actually calculating HD. The calculation of the SD-key, however, is performed in SIP registration and is only calculated once before users initiate VoIP calls. Thus our detection scheme does not require online SD-key calculation, and the SD-key calculation will then not directly affect the computational cost of attack detection. Between the two key components, the sketch operation needs to be done on every SIP message in the data stream, and the computation of Hellinger distance is performed at the end of each time interval. In our evaluation, we consider the CPU time needed to perform the sketch operations and Hellinger distance computations respectively on SIP messages during attacks within one time interval, i.e., 10 seconds, to see how our scheme performs in attack detection.

Table 2 shows the CPU time needed to detect the DDoS attacks. DDoS attack 1 is the same as described in Section 6.4, and DDoS attack 2 has double of the attacker number. The computer used to perform the experiments is a laptop PC with the CPU frequency of 2.4 GHz and memory of 4 GB. The combining CPU time

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

TABLE 2
Computational Cost - CPU Time

Operations	DDoS attack 1	DDoS attack 2
Sketch	0.2142	0.4202
Hellinger distance	0.4181	0.4004
Total	0.6323	0.8206

for the two operations are 0.6323 and 0.8206 seconds respectively for the two DDoS attacks. From the results we can see that even using a normal PC the overhead incurred by our detection scheme is not very high. Also to note is that even with the increase of attack intensities, i.e., more attackers or higher attacking rate, only the sketch operations will increase the needed CPU time, and the computation of HD will remain relatively stable. This is another good property of our detection scheme. Obviously, the CPU time for computing the hash value and calculating $c_i + b_i$ to process each SIP message in sketch is consistent. The increased CPU time for the sketch operations is due to processing a greater number of SIP messages in each time interval under more intense DDoS attacks.

Another aspect regarding the computational cost is memory. For attack detection, such cost is the memory used to maintain the sketch tables. In our three-dimensional design, we build two $4 * H * K$ sketch tables for the SIP messages from the training set and testing set respectively. If we use 32-bit integers to represent the value of one sketch entry, the memory cost for one three-dimensional sketch would be $128 * H * K$ bits. When we use $K = 32$ and $H = 5$ as specified in Section 6.1, the memory cost for two such sketch tables will be $40,960$ bits, which is very reasonable considering the memory and processing power of current computers.

The focus of this paper is on attack detection. However, the sketch-based technique also demonstrates its potential in attack prevention in previous sections. In our current design, each sketch entry needs to be further associated with a table to store the SIP addresses hashed into this entry. More efficient operations for attack prevention is a challenging research topic. We will address it in our future work.

7 Discussion

In this section we discuss some limitation of the proposed detection scheme and also one possible solution to the issue. Flooding attacks can bear various forms in order to evade detection. One special form of the attacks 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 following 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 detection schemes proposed in this paper and in [6] are not able to effectively address 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 caused 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 and have obtained some preliminary results [26].

8 Conclusion

The SIP flooding attack has become a major threat to the VoIP networks. In this paper, we propose an online VoIP flooding detection and prevention scheme by integrating two techniques, i.e., sketch and Hellinger distance. We first utilize sketch to build constant-size compact summaries of the SIP signaling message flows. The 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. Knowing that the original hash operation of sketch has limitations in detecting DDoS attacks, we further enhance sketch by utilizing information known only to normal users to establish the traffic distribution. 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 messages and subsequently prevent the attack. Since we establish probability distributions for each SIP attribute independently, our scheme is fully effective to the multi-attribute attack and is able to discriminate different forms of SIP flooding. We evaluate the performance of our scheme by conducting computer simulations. The experimental results show that the scheme preserves high accuracy on both attack

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2014.2302298, IEEE Transactions on Dependable and Secure Computing

detection and prevention. In our future work, we will further develop our scheme to improve its attack prevention rate against large scale DDoS attacks and more comprehensively evaluate the scheme using extensive VoIP traffic traces. Also, we will further 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," *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," *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, no. 6, pp. 13-35, Jan. 1993.

[9] C. Kreibich, and J. Crowcrowft, "Honeycomb: Creating Intrusion Detection Signatures Using Honeypots," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 1, pp. 51-56, Jan. 2004.

[10] Snort, [Online.] Available: http://www.snort.org/.

[11] A. Lakhina, M. Crovella and C. Diot, "Mining Anomalies Using Traffic Feature Distributions," *Proc. ACM SIGCOMM*, 2005.

[12] H. Sengar, D. Wijesekera, H. Wang and S. Jajodia, "VoIP Intrusion Detection Through Interacting Protocol State Machines," *Proc. IEEE International Conference on Dependable Systems and Networks*, 2006.

[13] R. Farley and X. Wang, "VoIP Shield: A Transparent Protection of Deployed VoIP Systems from SIP-based Exploits," *Proc. IEEE Network Operations and Management Symposium*, 2012.

[14] S. Ehlert, C. Wang, T. Magedanz and D. Sisalem, "Specification-based Denial-of-Service Detection for SIP Voice-over-IP Networks," *Proc. the Third International Conference on Internet Monitoring and Protection*, 2008.

[15] E. Chen, "Detecting DoS Attacks on SIP Systems," *Proc. 1st IEEE Workshop on VoIP Management and Security*, 2006.

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

[17] 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.

[18] 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.

[19] J. Tang, Y. Cheng and Y. Hao, "Detection and Prevention of SIP Flooding Attacks in Voice over IP Networks," *Proc. IEEE INFOCOM*, 2012.

[20] S. Muthukrishnan, "Data Streams: Algorithms and Applications," *Proc. the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2003.

[21] M. Thorup and Y. Zhang, "Tabulation Based 4-Universal Hashing with Applications to Second Moment Estimation," *Proc. the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2004.

[22] W. Stevens, *TCP/IP Illustrated Volume-1: The Protocols (1st ed.)*, Addison-Wiley, 1994.

[23] J. Kurose and K. Ross, *Computer Networking: A Top-Down Approach (4th ed.)*, Addison Wiley, 2007.

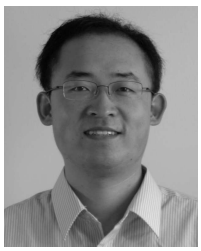
[24] SIP Express Router, [Online.] Available: http://www.iptel.org/ser/.

[25] F. Gustafson and M. Lindahl, "Evaluation of Statistical Distributions for VoIP Traffic Modelling," *University Essay from University West, Department of Economics and IT*, 2009.

[26] J. Tang and Y. Cheng, "Quick Detection of Stealthy SIP Flooding Attacks in VoIP Networks," *Proc. IEEE ICC*, 2011.

Jin Tang received the B.S. degree in Computer Science from Fudan University, Shanghai, China, in 2004, the Master's degree in Information Technology and Management from Illinois Institute of Technology, Chicago, IL, USA, in 2007, and the Ph.D. degree in Computer Engineering from Illinois Institute of Technology, Chicago, IL, USA, in 2012. He is now with AT&T Labs. His current research interests include wireless network security, intrusion detection and security in VoIP applications. He received a Best Paper Award from IEEE ICC 2011.

Yu Cheng received the B.E. and M.E. degrees in Electrical Engineering from Tsinghua University, Beijing, China, in 1995 and 1998, respectively, and the Ph.D. degree in Electrical and Computer Engineering from the University of Waterloo, Waterloo, Ontario, Canada, in 2003. Since August 2006, he has been with the Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, Illinois, USA, now as an Associate Professor. His research interests include next-generation Internet architectures and management, wireless network performance analysis, network security, and wireless/wireline interworking. He received a Postdoctoral Fellowship Award from the Natural Sciences and Engineering Research Council of Canada (NSERC) in 2004, and a Best Paper Award from the conferences QShine 2007 and ICC 2011. He received the National Science Foundation (NSF) CAREER award in 2011. He served as a Co-Chair for the Wireless Networking Symposium of IEEE ICC 2009, a Co-Chair for the Communications QoS, Reliability, and Modeling Symposium of IEEE GLOBECOM 2011, and a Technical Program Committee (TPC) Co-Chair for WASA 2011. He is an Associated Editor for IEEE Transactions on Vehicular Technology.



Yong Hao received the B.E. and M.E. degrees in Electrical Engineering from Huazhong University of Science and Technology, Wuhan, Hubei, China, in 2003 and 2007, respectively, and the Ph.D. degree in Computer Engineering from Illinois Institute of Technology, Chicago, IL, USA, in 2012. He is now with Juniper Networks. His current research interests include network security, cryptography, wireless network and vehicular ad hoc networks.

Wei Song received her Ph.D. degree in electrical and computer engineering from the University of Waterloo, Canada, in 2007. Since 2008, she has worked as a postdoctoral research fellow at the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley. In July 2009, she joined the Faculty of Computer Science, University of New Brunswick, as an Assistant Professor. She received a Harrison McCain Foundation Young Scholars Award in 2010, a Top 10% Award from IEEE Workshop on Multimedia Signal Processing (MMSP) 2009, and a Best Paper Award from IEEE WCNC 2007. Her current research interests include the interworking of cellular networks and wireless local area networks (WLANs), resource allocation for heterogeneous wireless networks, cooperative wireless networking, and cross-layer design for multimedia service provisioning.

1545-5971 (c) 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.