Appl. Math. Inf. Sci. 6 No. 2S pp. 579S-585S (2012)

Applied Mathematics & Information Sciences
An International Journal

A Prevention System for Spam over Internet Telephony

Ming-Yang Su[1] and Chen-Han Tsai[2]
[1,2] Department of Computer Science and Information Engineering, Ming Chuan University, Taoyuan, Taiwan
Corresponding author: Email: minysu@mail.mcu.edu.tw

Received June 22, 2010; Revised March 21, 2011; Accepted 11 June 2011
Published online: 1 January 2012

Abstract: Internet Telephony is realized by way of instant messaging and the transmitted data are voice rather than text packets, therefore it is more difficult to identify and/or prevent Spam over Internet Telephony (SPIT) than SPAM – junk e-mail. With the rising popularity of Voice over IP (VOIP), SPIT becomes a serious issue on security nowadays. This paper proposed an effective SPIT defense system, which can automatically collect the characteristics of an Internet Telephony and perform k-Nearest Neighbor classification to determine whether it is a SPIT. If convicted of being SPIT, the proposed system will immediately update the built-in blacklist in the SIP server to prevent SPIT from the same sender subsequently. The call connection protocol adopted in this study was Session Initiation Protocol (SIP). In the experiments, malicious software named *Spitter* was used to generate SPIT. The experimental results suggest that the system can get the overall accuracy up to 93.65%.

Keywords: VOIP, Spam over Internet Telephony (SPIT), K-Nearest Neighbor (kNN), Spitter, Session Initiation Protocol, Overall Accuracy.

1 Introduction

With the rapid development of Internet technology and advances in multimedia compression technology, Voice over Internet Protocol (VoIP) has achieved a series of connections from the earliest PC to PC, to the present PC to phone or phone to phone. Through the Internet, VoIP can transmit voice instantly and allow end users around the world to communicate face to face by video conferencing. It reduces communication costs and provides enterprises with integrated services of voice and video applications. VoIP has become one of the prerequisites of enterprises despite some security issues it entails.

E-mail useless and even distressing to users is called SPAM. Most of the contents are by nature advertising. Spam over Internet Telephony (SPIT) is to transmit useless voice information to VOIP users in real time. The advertiser may prepare voice recordings to broadcast using VoIP services through controlled Zombies. SPIT is far harmful than SPAM as users cannot prevent being disrupted as SPIT is delivered by instant calling software while the users can choose free time to receive email. Frequent SPIT callings can sure continuously disrupt the users at work. Regarding research on SPIT defense, the following approaches have been proposed in literature.

The authors of [1, 2, 3] basically proposed to analyze SIP URI, and then filter by using the black, white and gray lists. The authors of [4][5] adopted the Reputation System: each SIP social network user has individual rating values, by which the community users can determine to trust the other or not and provide a reference for different communities. For example, the Yahoo Auction website, after each transaction is completed, the buyer and seller will give the other party a rating value, by which other buyers or sellers can determine to transact or not. The authors of [6][7] prevented SPIT by authentication systems. As SIP is an open communications protocol, it relies on the From header of the SIP message to recognize the

identity of the caller when establishing a connection, however, its contents are prone to distortion and hiding of true identity. Internet Engineering Task Force (IETF) in 2002 released the P-Asserted-ID in RFC 3325, providing that the callee can determine the true identity of the caller within a small network or between small networks. However, such authentication mechanism cannot be implemented in case of large networks. To more effectively address the identity authentication issue, in August 2006, IETF proposed a more effective technology in RFC 4474. However, the call setup is very time-consuming as the caller and the callee have to confirm each other's identity on the authentication server.

The authors of [2, 8] proposed to install a Turing test module on the SIP server to request the caller to answer a few simple questions in order to discriminate the caller a machine (program) from the true person. The commonly used three methods are as follows. 1. Caller is requested to press some keystrokes such as "press 123", the callee can be connected only if the input is correct. 2. Caller will first hear a voice message and requested to answer a few simple mathematical calculations such as "1 + 1 = ?" the callee can be connected only if the input is correct. 3. The management server assigns a number for authentication subject to the control of the callee. At the connection establishment stage, the caller should send the notification to the other party and input the number for calling. The disadvantage of Turing test is the difficulty in implementation, and the inconvenience for the caller to answer questions before calling.

The authors of [1] used a cryptographic puzzle to require the callers' machines to consume a lot of CPU cycles to work out the answer, and generally interacted with Turing test. Besides, they also proposed another paid method to defense SPAM/SPIT. When user A sends a mail to user B, user A has to pay 10 cents USD to user B as guarantee. When user B receives the mail and confirms as non-SPAM, the deposited guarantee should be refunded to user A. Such a transaction procedure is rather complex and troublesome and difficult in implementation. [5] proposed a SPIT defense mechanism named the Socio-Technical Defense by using various caller-callee relationships including presence, rate limiting, black and white lists, trust, social networks and reputation to calculate whether the calling is SPIT. On the other hand, the authors of [9] used SPIT level which is computed according to the appearances of SIP messages, to judge whether a call is a SPIT or not.

This paper designed and implemented a system to prevent SPIT instantly. The proposed system is installed on the SIP server computer. When a connection request comes into the SIP server, if the caller is not on the built-in blacklist of the SIP server, the callee will be connected. After the end of calling, the proposed system will search relevant information in the database for kNN classification with the caller as the index to determine whether the calling is SPIT or not. If yes, the SIP server blacklist would be updated immediately to prevent the caller from sending SPIT again. The remainder of this paper is organized as follows: Section 2 briefly introduces VoIP-related Session Initiation Protocol and Real-time Transport Protocol; Section 3 introduces in detail the proposed system architecture and feature selection about a SIP call; Section 4 elaborates on the kNN classification effect in the design, finally, Section 5 is the conclusions of this paper.

2 Backgrounds

At present, SPAM, mostly advertising by nature, has caused great distress to individuals and businesses. Despite the development of various anti-SPAM technologies, the SPAM has been constantly changing and developing in pattern to penetrate various types of filtering technologies. One of them is SPIT which uses VOIP to transmit voice advertising. This section briefly introduces the Session Initiation Protocol and Real-time Transport Protocol (RTP)/ Real-time Transport Control Protocol (RTCP) used in VoIP.

Session Initiation Protocol (SIP) was appeared in RFC2543, and revised in RFC3261, in 1999 and 2002 respectively, as a communication protocol defined on the application layer for TCP and UDP protocols. SIP is a client-server structure, by which the user sends the request message and the server sends the appropriate response message upon the reception of the request message. Such a message exchange is applicable in the wide area network communications. VOIP adopts the independent operation on message and data. The SIP is responsible for the message exchanges of connection establishment. After that, the RTP/RTCP is responsible for transmitting voice packets.

SIP network mainly consists of four components including user agent (UA), SIP server, redirect server, and registrar server. UA is the end user, which can play a double role of UAC and UAS. When a UA is the caller, it plays the role of

UAC; if it is the callee, it plays the role of UAS. SIP server is the SIP network operational center. When UAC sends the connection request, the request will be directed to SIP server as UAC does not know the IP address of the callee, and then the SIP Server will pass the request to the destined UAS. Redirect server is to separate the logical position and the actual address of a user. Registrar server is the server to accept registration request and maintain the SIP URI (Uniform Resource Identifier) and IP addresses of all users. The Registrar server maintains the user identity authentication so that a user can inquire other users only after correct logging. There are 6 types of SIP request messages including the Invite, Ack, Bye, Cancel, Register, and Option. There are also 6 types of SIP response messages including 1XX notice response, 2XX success response, 3XX redirect response, 4XX request failure, 5XX server failure, and 6XX global failure.

When the caller sends the Invite message, the Session Description Protocol (SDP), appeared in RFC 2327, 3266, and 4566, information including the media format, address, communication port desired by the caller is also transmitted. The callee can determine to accept or reject the Invite message according to the media capabilities it can provide. SIP UA first should register on the Register Server. The registration process is important as it forbids one user to register more than one numbers at one time. After registration, the call setup process can be started as shown in Figure 1. UAC will send the Invite message to SIP server, which will first respond '100 Trying' to notify UAC of the reception of the Invite message, and attempts to pass the message to UAS. After receiving the start ringing, UAS responds '180 Ringing'. Upon receiving the response, SIP server will redirect to UAC to notify the ringing and waiting for response. When the UAS picks up the microphone, it will respond '200 OK' to UAC, which sends back Ack message to UAS to complete the establishment of connection, and then to start the talking by using RTP. RTP transmission does not need SIP server until the end of calling.

SIP message is transmitted between UA and SIP Server. It is vulnerable to attack without encryption. To avoid tampering of the message contents, some researches proposed the mechanisms to provide the SIP security such as using S/MIME to safeguard the message body dominated by SDP, or applying the IPsec or SSL/TLS mechanism to provide a safe channel between UA and SIP server, and so forth [10, 11, 12, 13, 14, 15, 16].

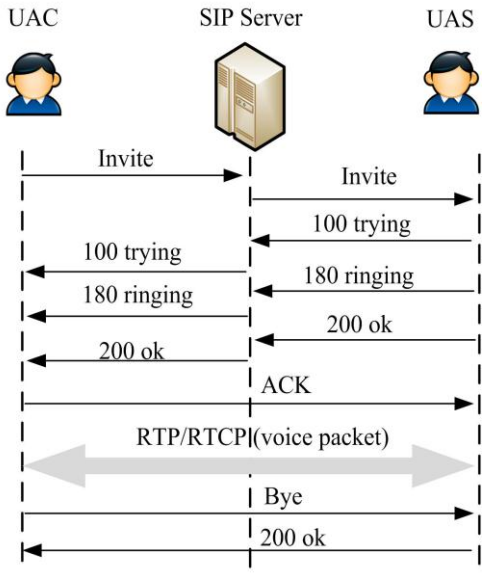


Figure 1: SIP connection establishment flowcharts.

Real-time Transport Protocol (RTP) and Real-time Transport Control Protocol (RTCP) formulated in RFC 1889 and 3550 provide real-time unicast, multicast and broadcast transmission services. RTP is based on the UDP protocol, while the transmission process is monitored by RTCP. RTP can be widely applied in various types of multimedia applications such as audio on-demand, video on-demand, Internet telephony and video conference. RTCP is to sniff service quality and transfer conferee-related information. RTCP packets do not pack voice or television data, but pack the sender and/or receiver statistics, including the number of transferred packets, the number of lost packets, and the jittering of the network traffic. According to the RTCP information, the sender may change transmission rate, or the receiver can determine the network problem, if exists, as local, regional or global. The combination of RTP and RTCP can maximize the transmission efficiency, hence, is particularly appropriate for real-time transmission.

3 The Proposed System

SPIT is a kind of real time interference. It is difficult to recognize SPIT at the connection establishment stage, because the SIP message information is very little. Hence, this study proposes a mechanism to analyze the SIP features after the calling. Once a certain user registered with SIP server is identified as a sender of SPIT, the

proposed system will automatically update the SIP server built-in blacklist, to prevent the user from sending SPIT again. The proposed system is shown in Figure 2. It mainly uses the kNN algorithm for judgment, requesting features from the two parts: 1) use a JNETPCAP [17] program to sniff SIP messages to retrieve requested features; 2) use a JDBC program to read the calling information from the SIP server logs. For the convenience in reading log files and timely updating of the SIP server's built-in blacklist, we installed the proposed system with SIP server on the same machine.

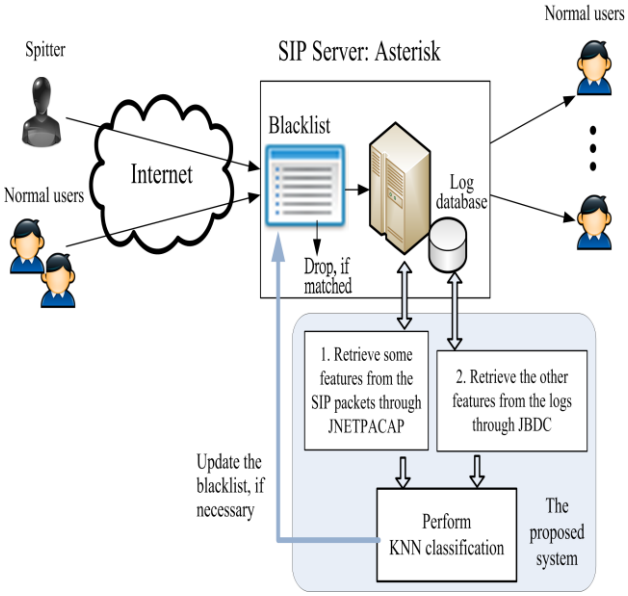


Figure 2: The proposed system's architecture

In Figure 2, the server hardware consists of Intel Pentium®4 CPU 3.40GHz, 768MB RAM, and the operating system is Elastix, the SIP server software is Asterisk [18]. the proposed system was implemented by JAVA, the database software is the MySQL attached with Elastix 1.6 Version. A total of 6 brands of Internet Telephony software are used including: X Lite, 3CX Phone, Mizu Phone, SIP Communicator, Zoiper, and Kapanga. The malicious software to send junk calls used in this study was *Spitter* [19]. It can edit and record voice data in advance (.CALL files), and send to the end users in great number. On the other hand, the normal instances of calling in this study were collected from the actual human VoIP calls.

For a phone call, the proposed system will collect 23 features as listed in Table 1 to judge it as SPIT or a normal call. All the features come from the logs of SIP-server log and SIP packet on line, with no feature coming from the voice packet. We note that a call's information would be recorded in the log only if it is accomplished according to SIP protocol, i.e., the callee pick up the phone. If a call is abandoned by the caller or no answering from the callee, this call's information will not be recorded in the log. Therefore, as shown in Figure 2, the proposed system uses a JNETPCAP [17] program to sniff online SIP packet in addition to using a JDBC program to inquire the records of SIP server's log. The information retrieved by the JNETPCAP program will also be stored in a database for future use. The column on the right side of Table 1 represents the feature coming from the SIP server log or feature coming from the online SIP packet.

Table 1: All 23 features used by the proposed system.

	Features by long term (24 hrs)	from
1	# calls from distinct caller	packet
2	# calls to distinct callee	packet
3	Total time (minutes) of all incoming calls	log
4	Total time (minutes) of all outgoing calls	log
5	Total duration (minutes)=feature3+feature4	log
6	# calls by the same caller and callee	log
7-11	# 1xx, 2xx, 4xx, 5xx, 6xx, respectively	packet
12	# INVITE packet	packet
13	# BYE packet	packet
14	Time (min.) since the last call from the caller	log
	Features by short term (1 hr)	
15	Total time (minutes) of all incoming calls	log
16	Total time (minutes) of all outgoing calls	log
17	# calls by the same caller and callee	log
18	# INVITE packet	packet
19	# Response packet, such as 1xx, 2xx, …	packet
20	#Request packet, such as INVITE,BYE, …	packet
21	# BYE packet	packet
22	Call rate	log
23	# incoming calls	log

The features as shown in the table can be mainly divided into two categorizes: long-term and short-term. The long-term features require longer time of observation, set as 24 hours in this study, while the short-term features require shorter time of observation, set as 1 hour in this study. In case of an incoming call through SIP server, if the caller is not on the blacklist, the caller is then connected to the callee through the server. After the end of the call, the proposed system will search in the logs/database for calling information with the caller as an index, to form an instances corresponding to the 23 features in Table 1. Then, perform kNN classification to determine whether the call is SPIT. If yes, update the SIP server blacklist immediately to directly reject the connection request of this caller on the SIP server next time.

Table 2 shows a normal call instance and a SPIT instance. Before applying kNN classification, data normalization should be done to prevent dominated impact of some excessively large feature values. Some of the features as shown in Table 3.1 are number of calls, for example, Feature 1 and Feature 2, which should be divided by the total number of calls within the period (long term or short term). Some features were time, for example, Feature 3 and Feature 4, which should be divided by the total minutes of calls within the period (e.g., the total minutes of the long-term features are 24*60, and the total minutes of the short-term features are 60). Some features were the number of messages, for example, Feature 7 and Feature 8, which should be divided by the total number of messages in the period at normalization.

Table 2: SIP features by instance

(a) A normal instance

F1	F2	F3	F4	F5	F6	F7	F8
15	20	10	7	17	2	12	18
F9	F10	F11	F12	F13	F14	F15	F16
1	0	0	4	3	2	3	7
F17	F18	F19	F20	F21	F22	F23	
1	4	12	20	10	2	1	

(b) A SPIT instance

F1	F2	F3	F4	F5	F6	F7	F8
0	60	0	25	25	2	80	40
F9	F10	F11	F12	F13	F14	F15	F16
10	4	5	60	30	0	0	20
F17	F18	F19	F20	F21	F22	F23	
1	30	40	45	0	20	0	

4 Experimental Results

The design of this research was based on the kNN classification In KNN classification, each instance with n features is represented by a point in the n-dimensional space. Assume there are some classified sample instances in the space, for the instance x of unknown class, kNN algorithm will find out k instances at the shortest distance from x, and classify x as the class of the majority. For example, as shown in Figure 3, k = 3, x would be classified as of class A as two of the three closets points (the majority) were of the class A.

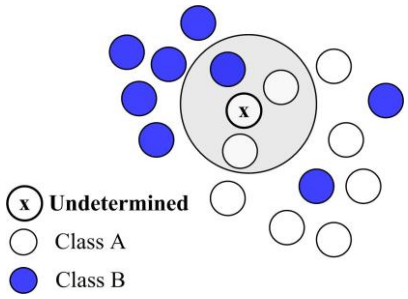


Figure 3: KNN diagram, k=3

The distance between two points in kNN generally refers to the Euclidean distance. Assume each instance has n features, the computation of Euclidean distance, dist(X, Y), between two points X = (x_1, x_2 , ..., x_n) and Y = (y_1, y_2 , ..., y_n) is as follows:

$$dist(X,Y)=\sqrt{(x_1-y_1)^2+(x_2-y_2)^2+...+(x_n-y_n)^2} \quad (4.1)$$

This study set up the SIP server and collected each call's features according to Table 1 of nearly one month. A total of 524 normal instances were collected and labeled as "+". In addition, the malicious software *Spitter* [19] generated 524 SPIT instances labeled as "-". According to the kNN classification design, the total ±524 instances should be divided into the sample dataset and test dataset. This study performed four experiments in case of different sizes of the sample dataset. We randomly selected ±10, ±30, ±50, ±100 instances as the sample dataset, and the rest ones as the instances of the test dataset to get the results as shown in Table 3. The overall accuracy (OA) is computed as (TP+TN)/(TP+FP+TN+FN). A false positive (FP) is a normal call being misclassified to SPIT; this term is also named false alarm. A false negative (FN) is a SPIT being misclassified to normal call. A true positive (TP) is a SPIT being

correctly classified to abnormal, and a true negative (TN) is a normal call being correctly classified to normal.

Table 3: Feature un-weighted classification results

#Sample: /#Test:	TP	FP	TN	FN	OA
± 10 ± 514	94.93 %	26.33 %	73.67 %	5.07%	84.30 %
± 30 ± 494	92.39 %	5.08%	94.92 %	7.61%	93.65 %
± 50 ± 474	85.30 %	2.67%	97.33 %	14.70 %	91.31 %
± 100 ± 424	88.58 %	4.32%	95.68 %	11.42 %	92.13 %

5 Conclusions

SPIT refers to undesirable Internet telephony; generally, it refers to pre-recorded voice data broadcast by Zombie using VoIP services. This paper designed and implemented on the Asterisk SIP server a system to prevent SPIT. The proposed system uses features of long-term and short-term categories; the long-term features were set for 24 hours and the short-term ones were set for 1 hour. When a connection request comes into the SIP server, the caller will be connected to the callee if found not on the SIP server's built-in blacklist. At the end of the call, PSPIT will search the database for calling records of previous 24 hours including the long-term and short-term features with the caller as the index to judge whether the call is SPIT. If yes, update the SIP server's built-in blacklist immediately to make the caller unable to send SPIT through the SIP server again. This study adopted the kNN algorithm to distinguish SPIT from normal calls. According to the experiments, the proposed system can get the best overall accuracy at 93.65%.

Acknowledgements

This work was partially supported by the National Science Council with contracts NSC 98-2221-E-130-007- and 99-2628-E-130-003-.

References

[1] Juergen Quittek, Saverio Niccolini, Sandra Tartarelli, and Roman Schlegel. On Spam over Internet Telephony (SPIT) Prevention. *IEEE Communications Magazine*. Vol.46, No.8, (2008), 80-86.

[2] Roman Schlegel, Saverio Niccolini, Sandra Tartarelli, and Marcus Brunner. Spam over Internet Telephony (SPIT) Prevention Framework. *IEEE on Global Telecommunications Conference*. (2006), 1-6.

[3] Dongwook Shin, Jinyoung Ahn, Choon Shim. Progressive Multi Gray-Leveling: A Voice Spam Protection Algorithm. *IEEE Network*. Vol.20, No.5, (2006), 18-24.

[4] Christoph Sorge and Jan Seedorf. A Provider-Level Reputation System for Assessing the Quality of SPIT Mitigation Algorithms. *IEEE International Conference on Communications*. (2009), 1-6.

[5] Prakash Kolan and Ram Dantu. Socio-Technical Defense Against Voice Spamming. *ACM Transactions on Autonomous and Adaptive Systems*. Vol.2, No.1, (2007), 1-44.

[6] Francisco J Puente, Octavio J Salcedo Parra, and José Camacho. Anti-Spit Mechanism based on Identity SIP. *IEEE International Conference on New Trends in Information and Service Science*. (2009), 591-596.

[7] Feng Cao and Cullen Jennings. Providing Response Identity and Authentication in IP Telephony. *The First International Conference on Availability, Reliability and Security*. (2006).

[8] J. Quittek, S. Niccolini, S. Tartarelli, M. Stiemerling, M. Brunner, and T. Ewald. Detecting SPIT Calls by Checking Human Communication Patterns. *IEEE International Conference on Communications*. (2007), 1979-1984.

[9] Bertrand Mathieu, Saverio Niccolini, and Dorgham Sisalem. SDRS: A Voice-over-IP Spam Detection and Reaction System. *IEEE Security & Privacy*. Vol.6, No.6, (2008), 52-59.

[10] Rainer Baumann, Stephane Cavin, and Stefan Schmid. Voice Over IP - Security and SPIT. *Swiss Army, FU Br 41, KryptDet Report*, (2006), 1-34.

[11] He Guang-Yu, Wen Ying-You, and Zhao Hong. SPIT Detection and Prevention Method Based on Signal Analysis. *The Third International Conference on Convergence and Hybrid Information Technology*. (2008), 631-638.

[12] Angelos Nakulas, Lambros Ekonomou, Stavroula Kourtesi, Georgios P. Fotis, and Emmanouil Zoulias. A Review of Techniques to Counter Spam and Spit. *Proceedings of the European Computing Conference*. (2009), 501-510.

[13] Hyung-Jong Kim, Myuhng Joo Kim, Yoonjeong Kim, and Hyun Cheol Jeong. DEVS-Based modeling of VoIP spam callers' behavior for SPIT level calculation. Simulation Modelling Practice and Theory. Vol.17, No.4, (2009), 569-584.

[14] David Butcher, Xiangyang Li, and Jinhua Guo. Security Challenge and Defense in VoIP Infrastructures. *IEEE Transactions on Systems, Man, and Cybernetics, Part C:*

Applications and Reviews. Vol.37, No.6, (2007), 1152-1162.

[15] Dimitris Gritzalisa and Yannis Mallios. A SIP-oriented SPIT Management Framework. *Computers & Security*. Vol.27, No.5-6, (2008), 136-153.

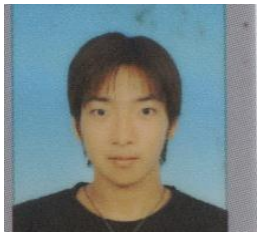
[16] S. Dritsas, V. Dritsou, B. Tsoumas, P. Constantopoulos, D. Gritzalis. OntoSPIT: SPIT management through ontologies. Computer Communications, Vol.32, No.1, (2009), 203-212.

[17] jNetPcap OpenSource, http://jnetpcap.com/.

[18] Asterisk: The Open Source Telephony Projects, http://www.asterisk.org/.

[19] Hacking VOIP Exposed, http://www.hackingvoip.com/sec_tools.html.

Ming-Yang Su received his B.S. degree from the Department of Computer Science and Information Engineering of Tunghai University, Taiwan in 1989, and received his M.S. and Ph.D. degrees from the same department of the National Central University and National Taiwan University in 1991 and 1997, respectively. He is an IEEE member, and currently a professor of the Department of Computer Science and Information Engineering at the Ming Chuan University, Taiwan. His research interests include network security, intrusion detection/prevention, malware detection, mobile ad hoc networks, and wireless sensor networks.



Chen-Han Tsai received the M.S. degree in 2011, from the Department of Computer Science and Information Engineering of Ming Chuan University, Taoyuan, Taiwan. His research interests are in the areas of network security, SIP security, and intrusion detection/prevention.