Exposing Resource Consumption Attacks in Internet Multimedia Services

Zisis Tsiatsikas*, Georgios Kambourakis*, Dimitris Geneiatakis†
*Dept. of Inform. and Comm. Systems Engineering, University of the Aegean, Karlovassi, Greece
Email: {tzisis, gkamb}@aegean.gr

†Institute of the Protection and Security Citizen, Joint Research Center, Ispra, Italy
Email: dimitrios.geneiatakis@jrc.ec.europa.eu

Abstract

Attackers always find ways to elude the employed security mechanisms of a system, no matter how strong they are. Nevertheless, audit trails - which as a rule of thumb are kept by any service provider - store all the events pertaining to the service of interest. Therefore, audit trail data can be a valuable ally when it comes to the certification of the security level of a given service. This stands especially true for critical real-time services such as multimedia ones, which nowadays are on the rise. This work proposes a practical, simple to implement yet powerful solution based on the Hellinger Distance metric for conducting audit trail analysis destined to expose security incidents. Our solution relies on a set of different features existing in the app layer protocol for session handling in order to classify the analyzed traffic as intrusive or not. Taking the well-known Session Initiation Protocol (SIP) as an example, we thoroughly evaluate the effectiveness of the proposed detection scheme in terms of accuracy under various realistic scenarios. The outcomes reveal competitive detection rates in terms of false positives and negatives and can be used as a reference for future works in the field.

Keywords—Session Initiation Protocol, Hellinger Distance, DoS, Abnormal Traffic

I. Introduction

Security audits and certifications can improve the dependability of virtually any service as they are able to identify limitations in the current deployment and suggest appropriate ameliorations. Most of the existing methodologies devoted to secure auditing, focus mainly on the level of business procedure. This is for instance the case with risk analysis approaches such as CRAMM [1] and COBBIT [2], or standardization business-level procedures, including ISO 27001 [3] and others.

While the aforementioned solutions are sure to be beneficial with respect to service dependability, they do not capitalize on data gathered by the underlying services. This mainly refers to log file data collected by default by the network accounting mechanisms. This is largely due to the lack of the appropriate tools for examining the logging data. As a result, it might be mistakenly concluded that the provided service is secure, while in practice is indeed vulnerable to various security flaws. A characteristic example of such a situation is low volume Denial of Service (DoS) attacks, which undoubtedly remain hard to detect and repel.

This is imperative especially for real-time multimedia services that need to ensure high availability almost under any condition. That is, in contrast to legacy telecommunication services provided by Public Switch Telephone Network (PSTN) which relies on a closed network architecture, modern multimedia services are offered over the open Internet architecture. To do so, various signaling protocols, such as Session Initiation Protocol [4], H.323 [5], *etc.*, have been proposed to manage the multimedia session effectively. SIP stands out as it has been adopted as the standard signaling protocol to manage multimedia session in Next Generation Networks (NGNs). However, this SIP blooming along with the open nature of the Internet and the text-based nature of the protocol have given rise to security concerns for virtually any service relying on it. Indeed, in the literature, there is no dearth of works addressing various security issues pertaining to SIP [6]–[9].

Overall, it would be very beneficial if the existing security audit controls and certification methodologies could be seconded by appropriate tools that are able to automatically analyze the collected audit trails and determine in a formal way whether or not the provided service suffers a security incident, e.g., a flooding attack. To contribute in this direction, in this work we investigate the feasibility of exploiting the well-known Hellinger distance (HD) metric [10] to identify abnormal traffic in SIP-based multimedia services. Specifically, our solution takes as input the various headers of a SIP message (belonging to a corresponding set of messages) in an effort to classify it as intrusive or not. This is done following a training phase with normal traffic and the calculation of a threshold that is used afterwards to make decision on each message contained in an unknown set. Although a couple of other research works [11], [12] take advantage of the same metric for detecting resource consumption attacks, our approach is quite different in both the input it feds to the formula, and the types of attacks it is able to identify. Based on detailed evaluations employing diverse realistic scenarios the proposed solution performs well with respect to other state-of-the-art detection schemes. The results reveal that false positives can reach up to 8%, while false negatives are rather negligible. Note that while SIP is taken as a case study for demonstrating the applicability of our proposal, we argue that it can be of

Fig. 1. A typical SIP INVITE request message

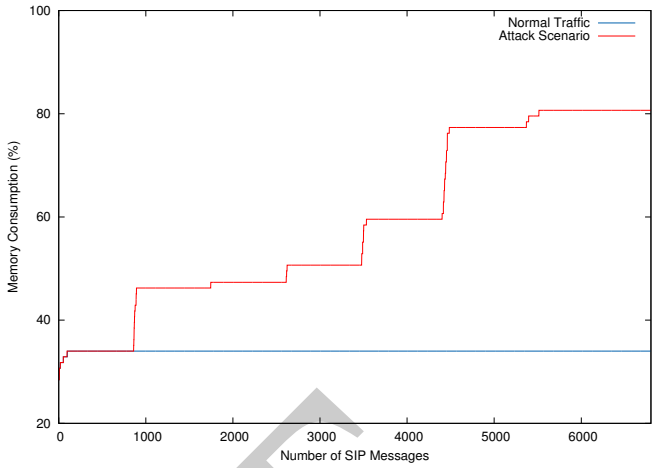

Fig. 2. Memory usage during an attack

benefit to similar protocols (services) as well.

The reminder of this paper is structured as follows. Section II provides an overview of SIP. Section III briefly describes the threat model. Section IV details on the proposed solution, while Section V offers offers an evaluation of the results. Finally, Section VII draws a conclusion and gives some pointers to future work.

II. Preliminaries

As the Internet dominates communication nowadays, telcos and other companies are trying to exploit its advantages to offer low cost voice multimedia communication services to their users. Very similar to legacy telecommunication systems the basis to do so is a signaling protocol in charge of managing the multimedia session. As already pointed out, nowadays, SIP seems to attract the major piece of attention. Intrinsically, SIP is designed to create, modify, and terminate voice and other more advanced multimedia sessions over the Internet. SIP is text-based with syntax similar to that of HTTP. SIP messages can be either a request or an acknowledgment to a corresponding request, consisting of the appropriate header fields and optionally a message body, depending on the nature of the request or response. An example of a typical SIP request message is given in Figure 1.

Whenever a user wishes to use a SIP service she should announce its presence by registering their current IP address to the registration service (registrar) through a SIP REGISTER message. After that, the user is able to initiate a session with other registered or interconnected User Agents (UA) by sending a SIP INVITE message to its local SIP proxy. After the call has been established, the two endpoints, namely the caller and callee are able to start the multimedia session with the help of Real-time Transport Protocol (RTP) [13]. At any time, either the caller or the callee may terminate the call by sending a SIP BYE message toward the other endpoint.

III. Threat Model

As noted in Section I, SIP based services are vulnerable to different types of attacks such as malformed message, flooding, SQL injection, or/and signaling ones [6]–[9], [14], [15]. Perhaps, the main reason behind this is the text-oriented nature of SIP and its simple syntax, allowing an attacker to easily manipulate the appropriate SIP headers to, say, trigger a DoS. For instance, an attacker might modify the first line of a SIP INVITE message depicted in Figure 1 by injecting various addresses (existing or not) in order to cause a DoS either to the provided service or to the end-user.

DoS attacks of high volume are sure to quickly attract the attention of the security guards. But what happens with other security incidents that are of low-volume and usually evade detection? Say, for example, that the aggressor sends out spoofed SIP requests having unresolved IP addresses, but does this at a slow pace and following a carefully designed wake/sleep strategy. Such stealthy incidents may gradually affect the availability of the provided service, mostly reflected to reduced bandwidth, which in turn causes users' dissatisfaction and reduced market share for the provider. To better highlight on this issue we executed a single DoS attack to a standard SIP server with a rate of only 10 INVITE message per second. Figure 2 shows the percentage of memory allocation at the server side during the attack. As observed, under normal operation, the memory consumption induced by the SIP server process is about 35%. However, as this simple attack unfolds, causes the corresponding percentage to increase to around 80%.

It is well-known that depending on the legislation of different countries, Telcos and service providers in general retain signaling data for a certain period of time for billing, auditing and network management and planning purposes. Therefore, these logs can be proved a valuable source of information toward identifying whether or not a given provider has been the target of a DoS. Generally, the analysis of such data could help one to (a) prove the security level of the provided services and investigate the related incidents, and (b) highlight the need of employing additional security protection measures to enhance service availability, say, due to attacks that managed to bypass the already deployed countermeasures.

IV. The Proposed Detection Scheme

A. Hellinger distance

Hellinger Distance is a well-known metric to quantify the similarity between two probability distributions, namely P and Q based on equation (1). According to theory, if two distributions are identical, then the HD value will be equal to zero, whereas in case where the distributions are dissimilar the HD value will tend to one (<=1). The distributions are defined according to equation (2), where p_i, q_i correspond to the probability occurrence of each symbol in the examined set.

$$HD = 1/\sqrt{2} \times \sqrt{\sum_{i=1}^{K} (\sqrt{p_i} - \sqrt{q_i})^2} \tag{1}$$

$$\begin{aligned} P &= \{p_1, p_2, ..., p_k\} \\ Q &= \{q_1, q_2, ..., q_k\} \end{aligned} \tag{2}$$

B. The detection service

To implement a detection service that relies on HD metric to identify traffic abnormalities one needs to choose certain parts of SIP message as the symbols of interest. By observing Figure 1 one can conclude that the most important parts of a SIP message are the first 6 lines (headers) corresponding to symbols S1 to S6. According to the literature, these symbols reflect the different types of SIP messages that an aggressor could craft in order to launch a resource consumption attack. For instance, a malicious entity could select to replay the same message or fabricate dissimilar SIP messages by modifying certain segments, including FROM, TO, Call-ID headers or even the First Line (corresponding to the fourteen different SIP methods) depending on the case. While a detailed analysis of SIP flooding attacks is out of the scope of this work the interested reader could refer to [16], [17].

After the symbols of interest have been selected, the detection service has to be fed with *P* distribution. That is, a training phase is required taking as input an attack-free log file. Note that P distribution can be updated less or more frequently depending on the provider's needs. One the other hand, the Q distribution is calculated on unknown traffic. Note that contrary to other approaches, P and Q distributions are generated for all the possible types of SIP methods. This has the advantage that it allows one not only to detect an attack incident but also identify its exact type (e.g., an INVITE or BYE flooding). To sum up, for every possible type of message there exists a pair of P and Q distributions. Depending on the case at hand, the audit trail file can be split into several message segments based on a predefined message window, say, equal to 1000. If so, P and Q have to be computed on the basis of this message window.

Specifically, to train the model we calculate the mean distribution value of every symbol included in the reference sample traffic. This procedure is given in algorithm 1. The main role of the training period is to compute the threshold, which is adjusted to the examined traffic by a parameter d, using equation (3). In fact, the parameter d coincides to a standard deviation metric, which is equal to the square root of the variance computed over a specific message type in the normal traffic set based on the mean value of HD.

$$Threshold = MeanHD + d \tag{3}$$

An unknown log file containing SIP transactions is examined for its conformity with the already determined normal model. That is, similar to the calculation of P, we reckon symbols distribution for the different requests and responses that formulate the corresponding Q distributions per message type. After that, P and Q distributions for the message type of interest are compared, as detailed in algorithm 2. In case that the HD value of the examined message is greater than the predefined threshold, then an alert is raised and the message is classified as malicious.

Algorithm 1: Obtain Theoretical Messages

```
   Input: Segmented-Attack-Free-File
   Output: TheorMessages
 1 Normalization;
 2 while (SegmentedFile ≠ NULL) do
 3     Line ← ReadLine();
 4     if Line is equal to FirstLine then
 5         TypeOfMessage = ExtractTypeOfMessage();
 6         TypeOfMessageCounter++;
 7     else
 8         Occurences ← ExtractOccurences(Line);
 9         switch(TypeOfMessage);
10         TheorMessages[TypeOfMessage][NumberOfHeader++]+ ←
           Occurences;
11     end
12 end
13 TheorMessages[TypeOfMessage][NumberOfHeader]/ ← TypeOfMessageCounter;
14 while (TheorMessages[TypeOfMessage][SipHeaders] ≠ NULL) do
15     Occurences ← ExtractOccurences(SipHeaders);
16     Normalization+ ← Occurences;
17 end
18 THeoreticalMessages[TypeOfMessage][SipHeaders]/Normalization;
```

Algorithm 2: Compute Hellinger Distance

```
   Input: TheorMessages, ExMessage
   Output: HellingerDistance
 1 DistributionEx;
 2 NormalizationEx;
 3 SipHeaders ← ExtractSipHeaders(ExMessage);
 4 while (SipHeaders ≠ NULL) do
 5     Occurences ← ExtractOccurences(SipHeaders);
 6     NormalizationEx+ ← Occurences;
 7 end
 8 DistributionEx ← Occurences(SipHeaders)/NormalizationEx;
 9 while (SipHeaders ≠ NULL) do
10     Sum+ ← (sqrt(DistributionEx[SipHeaders]) −
       sqrt(TheorMessages[Type(ExMessage)][SipHeaders]))^2;
11 end
12 HellingerDistance ← 0.5 * Sum
```

V. Evaluation

The detection accuracy of the proposed scheme has been tested under fifteen different scenarios briefly described in Table I. We simulated distinct patterns for both legitimate and attack traffic using *sipp v.3.2*[1] and *sipsak*[2] tools respectively.

[1] http://sipp.sourceforge.net/

[2] http://sipsak.org/

TABLE I. DESCRIPTION OF SCENARIOS

Scenario Number	Description
SN1	It simulates 30 legitimate users establishing 2 calls/sec. This is an attack-free scenario.
SN1.1, SN1.2, SN1.3	These sub-scenarios use the background traffic of SN2 and simulate multiple sources of SIP INVITE flood attack with rates of 50, 175, 350 calls/sec respectively.
SN2	This attack-free scenario simulates 30 legitimate users establishing 5 calls/sec.
SN2.1, SN2.2, SN2.3	These sub-scenarios use the background traffic of SN1 and simulate a single source SIP INVITE flood attack with a rate of 20, 40, 80 calls/sec respectively.
SN3	This last attack-free scenario incorporates 50 legitimate users establishing 20 calls/sec.
SN3.1	It relies on background traffic of SN3 and simulates 16 single source SIP INVITE floods each one with 266 calls/sec.
SN4	This attack-free scenario incorporates 50 legitimate users establishing 120 calls/sec.
SN4.1	It relies on the background traffic of SN4 and simulates 24 single source SIP INVITE floods each one with 800 calls/sec.
SN5	It simulates 50 legitimate users establishing 120 calls/sec. This scenario contains no attack traffic.
SN5.1, SN5.2	These sub-scenarios employ the background traffic of SN5 and simulate a single source SIP INVITE flood attack of 400, 1200 calls/sec respectively.

Also, for the needs of the experiments, and in order to reproduce realistic call rate conditions, we employed an exponential inter-arrival time distribution (λ = 100) for legitimate traffic similar to that used in evaluating SIP server performance [18].

Legacy IDS error assessment metrics, namely False Positive (FP) and False Negative (FN) [19], have been employed to measure the effectiveness of the detection engine. The first one pertains to messages classified as abnormal but they belong to the legitimate traffic, while the latter involves messages classified as normal but they belong to abnormal traffic. Table II summarizes the FP and FN results for all the scenarios. As observed, the FP fluctuates between 0.2% and 7.6%, whereas FN reaches a maximum value of 0.002%.

To exemplify the results obtained above, Figures 3 to 6 depict a fragment of HD distribution for scenarios SN1, SN1.1, SN2, SN2.3, SN3, SN3.1, and SN4, SN4.1 accordingly. Note that for easy reference and comparison the figures also include HD distribution for the corresponding attack-free traffic scenarios (*i.e.,* SN-1, SN2, SN3, SN4). This is to better conceptualize the fluctuations (increment in our case) exhibited in HD line between normal and intrusive traffic. Taking Figure 3 and SN1.1 as an example, one can easily observe that HD values remain as low as ≈ 0.01, while for SN1.1 the HD fluctuates between ≈ 0.04 and ≈ 0.25. A similar situation is depicted in Figures 5 and 6. Specifically, HD line for SN3.1 and SN4.1 abruptly reaches ≈ 0.25 when the attack is initiated. Note that in all the attack scenarios, normal and attack traffic take

TABLE II. SUMMARY OF EVALUATION METRICS

SN	Traffic (Calls)		FP		FN		Stats (HD)	
	Rec.	Attack	Inst	%	Inst	%	Mean	St. Dev.
SN1	1426	-	-	-	-	-	0.002	0
SN1.1	12000	5574	120	1	0	0	-	-
SN1.2	13000	7238	87	0.6	0	0	-	-
SN1.3	24530	19622	120	0.4	0	0	-	-
SN2	3598	-	-	-	-	-	0.041	0.061
SN2.1	11000	6516	566	5.1	2	0	-	-
SN2.2	14000	9327	592	4.2	0	0	-	-
SN2.3	15409	10802	1179	7.6	1	0	-	-
SN3	12435	-	-	-	-	-	0.161	0.155
SN3.1	667047	563705	5864	0.8	959	0.001	-	-
SN4	2505	-	-	-	-	-	0.025	0.028
SN4.1	178438	168073	8808	4.9	0	0	-	-
SN5	2004	-	-	-	0	0	0.018	0.021
SN5.1	261999	195050	1468	0.5	752	0.002	-	-
SN5.2	667769	601798	1463	0.2	788	0.001	-	-

turns. Naturally, this sudden increase in HD value is due to the attack traffic contained in scenario SN1.1, which in turn is translated into excessive symbol repetition, thus exceeding the predefined threshold. Nevertheless, if the attacker manages to generate traffic that has congruent characteristics to that of normal traffic she may be able to evade detection. On the opposite, however, the impact of the attack is anticipated to be much smaller in terms of probability of occurrence. Also, taking an INVITE method as an example, the aggressor is not able to randomly generate headers of the following format: INVITE sip:x@y:port, where x the username and y the host (domain or IP). This is because the IP used must correspond to an existent address of the internal network. Otherwise, due to packet filtering rules, the packet will be most likely dropped at the perimeter. If the attacker uses only legitimate IPs to perform the flooding attack, then due to the limited sip URI space, she is not in position to sent too many identical messages.

Generally, the occasional resemblance in terms of probability of occurrence between normal and attack messages is the most prevalent cause of FP alarms. More specifically, FPs are mainly the result of arbitrary device retransmissions or repetitive patterns in user's (call) behavior (e.g., a user calls another one very often). On the other hand, attack messages that appear seldom during the attack incident may generate an FN.

As discussed in the previous section, the threshold for all the scenarios - represented as a flat line in each of the figures - is calculated as the sum of the last two columns of Table II for all normal traffic scenarios. So, depending on the uniformity of the messages included in each scenario with their mean value, the threshold is expected to set a boundary above which a message is categorized as suspicious. For example, for scenario SN1 the threshold is nearly zero while for SN3 is 0.316. Lastly, the scarce upward peaks observed in the normal traffic (blue) line belong to messages that differ considerably from all the others contained in the normal traffic set. That is, their headers appear very frequently causing HD to suddenly spring up. As already pointed out, this may happen due to, say, device retransmissions. On the other hand, the downward pointing peaks observed in the HD line belong to normal messages that are interposed between the attack ones.

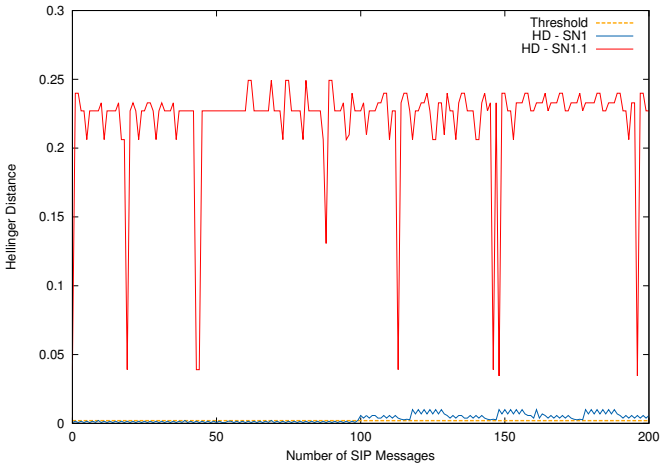

Fig. 3. A fragment of HD values for scenarios SN1 and SN1.1

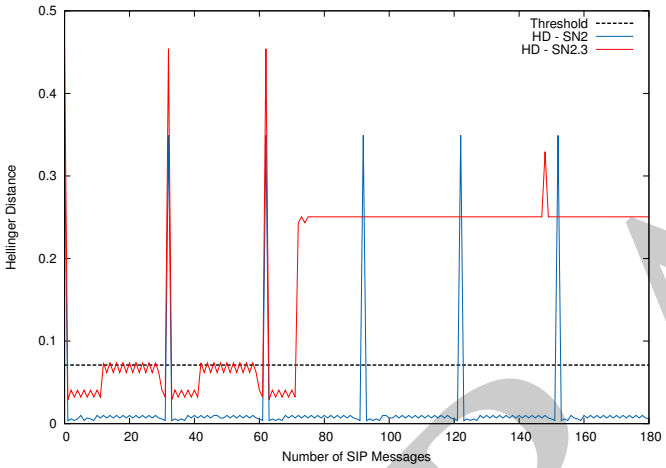

Fig. 4. A fragment of HD values for scenarios SN2 and SN2.3

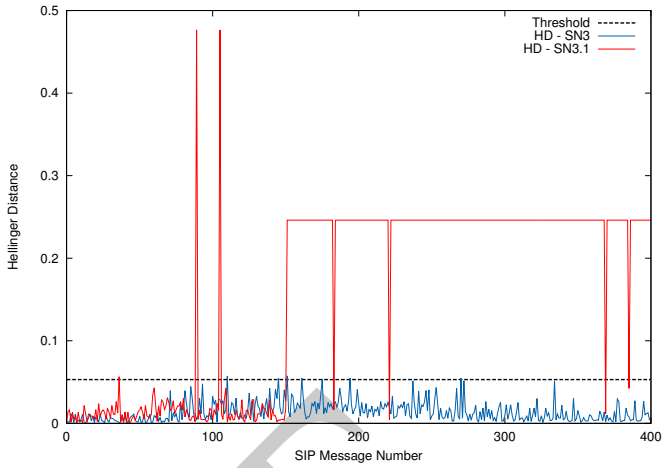

Fig. 5. A fragment of the HD for scenarios SN3 and SN3.1

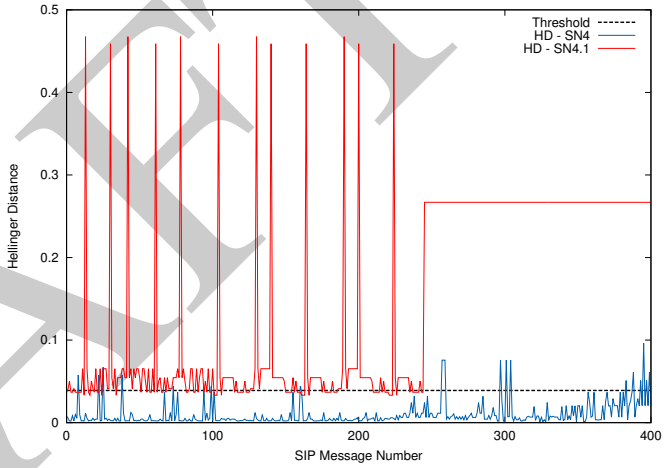

Fig. 6. A fragment of the HD values for scenarios SN4 and SN4.1

VI. Related Work

This section provides a brief overview of similar to ours solutions that rely on HD metric to detect abnormal of suspicious network traffic. Other works on VoIP security in general are considered out of scope and have been intentionally neglected. The authors in [11] contributed a proposal that relies on HD metric to classify the incoming traffic. Specifically, they built a real-time monitor to detect deviations between specific app layer traffic attributes, namely requests and responses. So, if the number of requests exceeds by far the number of responses (or the opposite) an attack alert is raised.The performance of the proposed framework using different call generation rates was evaluated. According to their results based on four different scenarios, the detection accuracy ranges between 80% and 100%. Similarly, the work in [12] builds a solution that combines HD metric with the sketch data structure [20]. The authors examine the differences among distributions originating from specific requests and responses, that is INVITE, 2OO OK. They report a false alarm percentage that fluctuates between zero to nearly 8% or more for DDoS detection depending on the configuration. As already discussed in the previous sections, our approach differs from the above mentioned proposals in that not only allows for the detection of an attack incident with high certainty, but also makes possible the identification of its exact type.

VII. Conclusions

Without doubt, audit trails are rich of valuable information that can be exploited to certify and improve the security of virtually any app layer service. The paper at hand introduces a framework destined to the detection of resource consumption attacks in SIP VoIP services. While our solution relies on the Hellinger distance metric, it combines different symbols that usually exist in a multimedia service audit trail to tell if a service suffers a resource consumption attack. The outcomes reveal that our approach can be used effectively to assist security auditing certification, making the basic assumption

that an attack free audit trail exists. In the case of SIP, the construction of such a log file can be decisively seconded by the billing service, because these logs are supposed to be accurate and valid [21]. All the results reported in the context of this work correspond to off-line testing. Our intention is to expand the solution to work in real-time by inspecting each incoming message on the fly. This requires the development of a software module destined to SIP proxies. The applicability of the results of this work to similar services and protocols are also of great interest.

ACKNOWLEDGEMENTS

This paper is part of the 5179 (SCYPE) research project, implemented within the context of the Greek Ministry of Development-General Secretariat of Research and Technology funded program "Excellence II / Aristeia II", co-financed by the European Union/European Social Fund - Operational program "Education and Life-long Learning" and National funds.

REFERENCES

[1] "The ccta risk analysis and management method (cramm) user guide," 1993.

[2] L. Al Omari, P. H. Barnes, and G. Pitman, "Optimising cobit 5 for it governance: examples from the public sector," in *Proceedings of the ATISR 2012: 2nd International Conference on Applied and Theoretical Information Systems Research (2nd. ATISR2012)*. Academy of Taiwan Information Systems Research, 2012.

[3] "ISO27001: Information Security Management System (ISMS) standard," *Online: http://www.27000.org/iso-27001.htm*, Oct. 2005.

[4] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "Sip: Session initiation protocol," United States, 2002.

[5] C. K. Yeo, S. C. Hui, I. Y. Soon, and L. M. Ang, "H.323 compliant voice over ip system," *Int. J. Comput. Appl. Technol.*, vol. 16, no. 4, pp. 143–153, Jul. 2003.

[6] S. Ehlert, D. Geneiatakis, and T. Magedanz, "Survey of network security systems to counter sip-based denial-of-service attacks," vol. 29, no. 2, 2010, pp. 225 – 243.

[7] D. Geneiatakis, T. Dagiuklas, G. Kambourakis, C. Lambrinoudakis, S. Gritzalis, K. Ehlert, and D. Sisalem, "Survey of security vulnerabilities in session initiation protocol," *Communications Surveys Tutorials, IEEE*, vol. 8, no. 3, pp. 68–81, rd 2006.

[8] A. D. Keromytis, "A comprehensive survey of voice over ip security research," *IEEE Communications Surveys and Tutorials*, vol. 14, no. 2, pp. 514–537, 2012.

[9] D. Geneiatakis and C. Lambrinoudakis, "A lightweight protection mechanism against signaling attacks in a sip-based voip environment," *Telecommunication Systems*, vol. 36, no. 4, pp. 153–159, 2007.

[10] M. Nikulin, "Hellinger distance."

[11] 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. [Online]. Available: http://dx.doi.org/10.1109/TPDS.2007.70786

[12] J. Tang, Y. Cheng, Y. Hao, and W. Song, "Sip flooding attack detection with a multi-dimensional sketch design," *Dependable and Secure Computing, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.

[13] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "Rtp: A transport protocol for real-time applications," United States, 2003.

[14] D. Geneiatakis, G. Kambourakis, C. Lambrinoudakis, T. Dagiuklas, and S. Gritzalis, "Sip message tampering: The sql code injection attack," in *Proceedings of 13th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2005), Split, Croatia*, 2005.

[15] D. Geneiatakis, G. Kambourakis, C. Lambrinoudakis, T. S. Gritzalis, "A framework for protecting a sip-based infrastructure against malformed message attacks," *Comput. Netw.*, vol. 51, no. 10, pp. 2580–2593, Jul. 2007. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2006.11.014

[16] D. Geneiatakis, N. Vrakas, and C. Lambrinoudakis, "Utilizing bloom filters for detecting flooding attacks against SIP based services," *Computers & Security*, vol. 28, no. 7, pp. 578–591, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.cose.2009.04.007

[17] S. Ehlert, "Denial-of-service detection and mitigation for SIP communication networks," Ph.D. dissertation, Berlin Institute of Technology, 2009. [Online]. Available: http://opus.kobv.de/tuberlin/volltexte/2010/2496/

[18] R. Krishnamurthy and G. Rouskas, "Evaluation of sip proxy server performance: Packet-level measurements and queuing model," in *Communications (ICC), 2013 IEEE International Conference on*, June 2013, pp. 2326–2330.

[19] G. Gu, P. Fogla, D. Dagon, W. Lee, and B. Skorić, "Measuring intrusion detection capability: an information-theoretic approach," in *Proceedings of the 2006 ACM Symposium on Information, computer and communications security*, ser. ASIACCS '06. New York, NY, USA: ACM, 2006, pp. 90–101.

[20] B. Krishnamurthy, S. Sen, Y. Zhang, and Y. Chen, "Sketch-based change detection: Methods, evaluation, and applications," in *Proceedings of the 3rd ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '03. New York, NY, USA: ACM, 2003, pp. 234–247. [Online]. Available: http://doi.acm.org/10.1145/948205.948236

[21] D. Geneiatakis, G. Kambourakis, and C. Lambrinoudakis, "A mechanism for ensuring the validity and accuracy of the billing services in ip telephony," in *Trust, Privacy and Security in Digital Business*, ser. Lecture Notes in Computer Science, S. Furnell, S. Katsikas, and A. Lioy, Eds. Springer Berlin Heidelberg, 2008, vol. 5185, pp. 59–68. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-85735-8_7