A First Order Logic Security Verification Model for SIP

Dimitris Geneiatakis[1], Costas Lambrinoudakis[1], Georgios Kambourakis[1], Aggelos Kafkalas[1] and Sven Ehlert[2]

[1] Department of Information and Communications Systems Engineering
University of the Aegean, Karlovassi, GR-83200 Samos, Greece

[2] Fraunhofer FOKUS, Berlin, Germany
Kaiserin-Augusta-Allee 31, 10589 Berlin, Germany

Abstract: It is well known that no security mechanism can provide full protection against a potential attack. There is always a possibility that a security incident may happen, mainly as a result of a new or modified attack that the employed countermeasures cannot handle or identify. It is therefore useful to perform a deferred analysis of logged network data, in an attempt to identify abnormal behavior/traffic that flags some type of security incident that has not been detected by the security countermeasures. Such an analysis of logged data for critical real time applications, like VoIP services, is certainly a valuable tool for enhancing the security level of the provided service.

In this paper we introduce a practical tool that can be employed for the analysis of logged VoIP data and thus validate the effectiveness of the security mechanisms and the conformance with the corresponding security policy rules. For the analysis of the data we capitalize on our security model for VoIP services [25] that is based on First Order Logic concepts, while the Protégé API and the Semantic Web Rule Language (SWRL) are also exploited. The proposed tool has been evaluated in terms of an experimental environment, while the results obtained confirm the validity of its operation and demonstrate its effectiveness.

***Index Terms*—Voice over IP, Session Initiation Protocol, Forensics, Security**

I. Introduction

Today the Internet is considered as the de-facto telecommunication network supporting delivery of sophisticated, but low cost, services and effective resource utilization [1]. Voice and telephony services are among the big variety of services offered through it. Internet-based voice services, known as Voice over IP (VoIP) or Internet Telephony, rely on an open public and distributed network architecture. This of course does not apply to Public Switch Telephone Network (PSTN) services, which rely on a closed, centralized network architecture. In this regard different protocols for session management (signaling protocols) and data transmission (multimedia protocols), that also take into account Internet specific characteristics, are required. Currently, organizations and companies deploy different protocols for session management, i.e. H.323 [2], Session Initiation Protocol (SIP) [3], Skinny Call Control Protocol (SCCP) [4] etc. The predominant protocol among them is SIP, as it has been adopted by various standardization organizations as the standard protocol for establishing multimedia sessions in both wireline and wireless world in the Next Generation Networks (NGN) era. On the other hand, for data transmission the Real Time Protocol (RTP) [5] is employed.

Until today many security flaws for VoIP services have been reported in the literature. These flaws have been identified for both signaling and transport protocols [6][7]. This is not the case for PSTN services, where security flaws are seldom exploited due to its closed architecture. Several security mechanisms [8]-[16] have been proposed not only for protecting VoIP services against possible attacks but also for enhancing the overall security level. Yet, to ensure that a certain level of security is maintained, the system behavior must be controlled and restrained by specific security policies. One might consider a security policy as a set of rules that regulates the nature and the context of actions that can be performed within a system according to specific roles and rules. Nevertheless the main problem with this approach is that it is quite difficult to verify whether a system implementation conforms to its policy. Besides, the identification of security flaws in a network service is currently performed mainly by penetration or security testing. Generally, security testing is considered as an important activity that helps not only to evaluate the security level, but also to identify security vulnerabilities, flaws and attacks that could be possibly launched against a network service. At the very early stages of security testing, domain experts were responsible for employing manually appropriate tests based on well-known attack signatures and patterns [18]. Nevertheless, over the last years various tools like Nessus (http://www.nessus.org), Retina (http://www.eeye.com/) etc, have been developed in order to automate the security testing procedure. Furthermore, a lot of effort has been put into developing appropriate security tests [19][20], during the various development phases of an information system. Usually these tests are based on predefined system specifications in order to validate its conformance with them.

As far as VoIP security services validation and identification of security flaws is concerned, very little has been made [21]-[24]. Specifically a first research work focusing on the security evaluation of SIP parsers using black box testing methods is presented in [21]. In [22] the authors propose a tool named Fuzzy Packet, which provides a wide range of features to manipulate any kind of SIP messages over a network through injection or capturing packets. The tool is able to assess the security level of a specific SIP based VoIP service. Moreover [23] and [24] introduce two different frameworks that could be utilized to perform VoIP-specific penetration tests.

To the best of our knowledge until today there is no published research work focusing on the verification of the security policy of VoIP services or on the analysis of VoIP logged data in order to identify security flaws and problems. In this paper we elaborate on our security model for VoIP services [25] which is able to verify in a formal way the employed security policy. Specifically, we introduce and evaluate a practical tool that can accurately identify any misbehavior by analyzing the network traffic log files. The proposed tool builds on top of our model by exploiting the Protégé API, and Semantic Web Rule Language (SWRL). Note that our model utilizes *First Order Logic (FOL)* concepts targeting on SIP based VoIP services. However, with minor modifications it can be also utilized with alternative signaling protocols.

The rest of the paper is structured as follows: Section II provides a brief description of SIP based VoIP services and an overview of the potential attacks against them. Section III presents and analyzes the proposed FOL model for security analysis in SIP based VoIP services. Section IV introduces a novel tool that capitalizes the proposed model for detecting deviations from the security policy adopted. The last section concludes the paper and gives pointers to future work.

II. Background

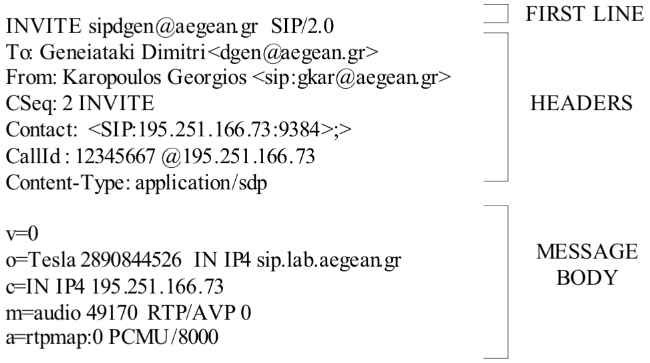
A. SIP Based VoIP Architecture

Beyond and above the utilization of existing Internet protocols, VoIP services require the development of specific protocols on the Internet application level for session management and media transmission. Regarding the former, SIP [3] is the predominant protocol designed for administrating multimedia sessions provided through the Internet. Particularly, SIP is an application layer signaling protocol for managing multimedia sessions among two or more participants over the Internet. Also SIP is a text-based protocol, which inherits its message structure from HTTP. Very briefly, a SIP message consists of the First Line, which designates whether a message is a request or a response. First Line is followed by other headers providing specific details that are required for message routing. Figure 1 depicts an example of a SIP INVITE message used to establish a multimedia session between a caller (gkar) and a callee (dgen). Generally, such messages are utilized for session administration. The exchanged messages between the two parties and the corresponding SIP proxy for establishing a session are illustrated in Figure 2. After successful session establishment the RTP or Secure RTP (SRTP) protocols [17] are utilized for media transmission.

B. Security Flaws in SIP Based VoIP Services

Security flaws and vulnerabilities in VoIP have been the subject of various research works [6],[7]. Most of them are specifically targeting on SIP, although, similar security flaws may arise in VoIP services employing alternative signaling protocols. Specifically and without loss of generality, security flaws in SIP based VoIP services could be classified into the following categories [6]:

- Malformed Message Attacks: In this class of attacks a malicious user crafts messages that are not compliant with the corresponding signaling protocol grammar specifications and sends them toward the provided service in order to cause a Denial of Service (DoS) or gain unauthorized access.
- Resource Consumption Attacks: In this category of attacks a malicious user sends a large number of well-structured messages in order to overwhelm system's resources causing a DoS.
- Signaling Attacks: In this last class of attacks the attacker injects into the network one or more messages in order to illegally modify a specific session's parameters causing that way either a DoS or gaining unauthorized access in the provided service.



```
INVITE sipdgen@aegean.gr  SIP/2.0
To: Geneiataki Dimitri<dgen@aegean.gr>
From: Karopoulos Georgios <sip:gkar@aegean.gr>
CSeq: 2 INVITE
Contact:  <SIP:195.251.166.73:9384>;>
CallId : 12345667 @195.251.166.73
Content-Type: application/sdp

v=0
o=Tesla 2890844526  IN IP4 sip.lab.aegean.gr
c=IN IP4 195.251.166.73
m=audio 49170  RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Figure 1. An Example SIP Request Message

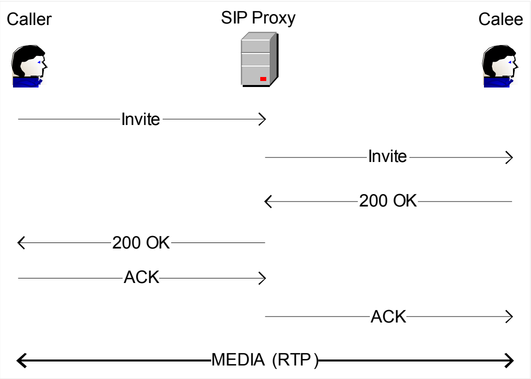


Figure 2. Session Establishment Procedure

III. A Security Model For VoIP Services [25]

In order to detect or / and prevent the aforementioned attacks, the VoIP community has proposed various security countermeasures [8]-[16]. However, as it is always the case, no security mechanism can provide 100% protection in terms of identifying or preventing an attack. There is always a possibility that a security incident may happen, mainly as a result of a new or modified attack that the employed countermeasures cannot handle / identify. As a result it is useful, if not necessary, to perform a deferred analysis of logged network data in an attempt to identify abnormal behavior/traffic that flags some type of security incident that has not been detected by the security countermeasures. It is important to stress that such an analysis can only be considered effective when done through automatic tools.

As presented in the next section, for the analysis of the network logged data we capitalize our security model for VoIP services that has been published in [25]. The specific model is based on FOL concepts and on the simple idea that any type of attack, irrespectively of the application that it aims at, utilizes the corresponding protocol's messages trying to cause a specific problem.

For completeness purposes we provide a very brief description of the model's main characteristics. It consists of two parts. The first part formalizes the SIP message structure. Specifically, the predicates *SIP_Message* (see Formula 1) and *SIP_Header* (see Formula 2) define specific rules based on SIP grammar specifications. These rules can be used to validate if a given SIP message conforms to specification. Formula 3 defines an additional rule for SIP message conformance i.e., any SIP message requires the existence of a first line and at least three SIP headers. Formula 4 designates that a first line should be a request or response, whereas, formula 5 shows that a request should not be a response and vice versa. Formula 6 represents the structure of a request, which consists of a specific SIP method (see formula 7) and the requested resource. Formulas 8 to 17 represent relations between specific "requirements", like the authentication and/or timestamp headers included in a SIP message and the SIP message itself. For instance, Formula 8 specifies that every SIP message m1 must be authenticated using a mechanism a1.

$$SIP\_Message(x)\ (1)$$

$$SIP\_Header(x)\ (2)$$

$$\forall x\ SIP\_Message(x) \Leftrightarrow \exists f\ FirstLine(f)$$

$$\wedge \exists^{\geq 3} h\ \ SIP\_header(h)\ (3)$$

$$\forall f\ FirstLine(f) \Rightarrow \mathrm{Re}quest(f) \vee \mathrm{Re}sponse(f)\ (4)$$

$$\forall x\, \mathrm{Re}quest(x) \Rightarrow \neg\, \mathrm{Re}sponse(x)\ (5)$$

$$\forall r\, \mathrm{Re}quest(r) \Leftrightarrow \exists m\ Method(m) \wedge \exists rs\, \mathrm{Re}source(rs)\ (6)$$

$$\forall m\ Method(m) = \{INVITE \vee REGISTER \vee OPTIONS\}\ (7)$$

$$\forall m1, a1\ is\_auth(m1, a1) \Rightarrow authenticate(a1)\ (8)$$

$$\forall m1, a1\ is\_auth(m1, a1) \Rightarrow SIP\_message(m1)\ (9)$$

$$\forall m1, t1\ message\_sent(m1, t1) \Rightarrow SIP\_message(m1)\ (10)$$

$$\forall m1, t1\ message\_sent(m1, t1) \Rightarrow time(t1)\ (11)$$

$$\forall m1, e1\ create(m1, e1) \Rightarrow SIP\_message(m1)\ (12)$$

$$\forall m1, e1\ create(m1, e1) \Rightarrow Event(e1)\ (13)$$

$$\forall m, h\ has\_header(m, h) \Rightarrow SIP\_message(m)\ (14)$$

$$\forall m, h\ has\_header(m, h) \Rightarrow header(h)\ (15)$$

$$\forall m, f\ inc\_first\ln(m, h) \Rightarrow SIP\_message(m)\ (16)$$

$$\forall m, f\ inc\_first\ln(m, h) \Rightarrow SIP\_message(m)\ (17)$$

The second part of the proposed model corresponds to the definition of SIP security flaws in *FOL*. As already mentioned in Section II security flaws in SIP could be one of the following types: (a) malformed, (b) signaling and (c) flooding. Formulas 18 to 24 represent in *FOL* these types of attacks, which are independent from each other. Particularly, regarding SIP malformed attacks, Formula 25 defines this type of attack as the complement of a SIP message. This means that if the predicate *SIP_Message* evaluates in a false value the formula 25 will trigger an alarm for malformed message attack. Signaling attacks represented by Formulas 26 and 27 designate that such incidents might take two forms: (a) the existence of two or more identical SIP messages within different time frames is considered as a signaling attack (see Formula 26), and (b) according to the policy that requires authentication, any not authenticated message is also considered as a signaling attack (see Formula 27).

On the other hand, flooding attacks might take the form of single or multiple source attacks (see Formula 28). A flooding attack is characterized as single source if a target receives from another SIP node a number of messages that exceeds a given threshold (see Formula 29), or as multi source if the number of simultaneous single source attacks exceeds a specific threshold (see Formula 30).

$$\forall m\ SIP\_Attack\ (m) \Leftrightarrow Malformed\ (m) \vee Signalling\ (m)$$

$$\vee\ Flood\ (m)\ (18)$$

$$\forall m\ Malformed\ (m) \Rightarrow \neg Singalling\ (m)\ (19)$$

$$\forall m\ Malformed\ (m) \Rightarrow \neg Flood\ (m)\ (20)$$

$$\forall m\ Flood\ (m) \Rightarrow \neg Singalling\ (m)\ (21)$$

$$\forall m\ Flood\ (m) \Rightarrow \neg Malformed\ (m)\ (22)$$

$$\forall m\ Signalling\ (m) \Rightarrow \neg Flood\ (m)\ (23)$$

$$\forall m\ Signalling\ (m) \Rightarrow \neg Malformed\ \ (24)$$

$$\forall m\ \neg SIP\_Message(m) \Leftrightarrow Malformed\ (m)\ (25)$$

$$\forall m1, m2\ SIP\_Message(m1) \wedge SIP\_Message(m2)$$

$$\wedge\ SameAs(m1, m2) \Leftrightarrow Signalling\ (m1)\ (26)$$

$$\forall m\ SIP\_Message(m) \wedge \neg Authentica\ te(m)$$

$$\Leftrightarrow Singalling\ (m)\ (27)$$

$$\forall m\ Single(m) \vee Multi\ (m) \Leftrightarrow Flood\ (m)\ (28)$$

$$\forall m\ Single(m) \Leftrightarrow Number\_of\ (m) > thrshlds$$

$$\wedge\ directed\ (m, t) \wedge source\_is(m, s)\ (29)$$

$$\forall m\ Multi\ (m) \Leftrightarrow Number(Single(m)) > thrhsldm\ \ (30)$$

$$\forall m\ Malformed\ (m) \Rightarrow SIP\_Attack\ (m)\ (31)$$

$$\forall m\ Signalling\ (m) \Rightarrow SIP\_Attack\ (m)\ (32)$$

$$\forall m\ Flood\ (m) \Rightarrow SIP\_Attack\ (m)\ (33)$$

$$\forall m\ Single(m) \Rightarrow Flood(m)\ (34)$$

$$\forall m\ Multi\ (m) \Rightarrow Flood\ (m)\ (35)$$

$$\forall a, m\ Attack\ \_utilize\ (a, m) \Rightarrow SIP\_Attack\ (a)\ (36)$$

$$\forall a, m\ Attack\ \_utilize\ (a, m) \Rightarrow SIP\_message(m)\ (37)$$

$$\forall a, t\ attack\_t\arg et(a, t) \Rightarrow SIP\_Attack\ (a)\ (38)$$

$$\forall a, t\ attack\_t\arg et(a, t) \Rightarrow t\arg et(t)\ (39)$$

$$\forall\ a1, c1\ attack\ \_cause(a1, c1) \Rightarrow SIP\_Attack\ (a1)\ (40)$$

$$\forall\ a1, c1\ attack\ \_cause(a1, c1) \Rightarrow consequence(c1)\ (41)$$

Formulas 31 to 35 show the relationship among the predicates *Signaling, Flood* and *Malformed* with the general predicate *SIP_Attack*. The remaining Formulas 36 to 41 represent the relationship among the different resources that are utilised during an attack incident. For instance, Formula 37 indicates which SIP message 'm' has been utilized in the SIP attack incident 'a', while Formula 39 identifies the target 't' of the corresponding attack.

IV. Inspecting VoIP Data For Attack Identification

In order to apply and test the proposed model in a real environment, we have embedded it into the ontology model already described in [28]. This decision was based on the fact that ontologies could be utilized for providing a common understanding of concepts within a specific domain. Also our *FOL*-driven ontology provides strict logic rules, thus avoiding inconsistencies and ambiguities. Figure 3 depicts the abstract ontology representation of the proposed model. The "Sip_message" part of the ontology is in accordance with Formulas 1 to 17, while the "Sip_attack" part corresponds to Formulas 18 to 41.

The general procedure that was followed in order to inference about the validity of the VoIP logged data and the effectiveness of the employed security mechanism, is depicted in Figure 4. More specifically, the procedure consists of 4 steps. During the fist step (0) raw-data from SIP transactions is collected or captured from the network. Employing the ontology descriptions these raw data are transformed to semantic ones during steps 1,2. The last step uses semantic data as input to infer about the validity of SIP messages, taking into account the corresponding security policy.

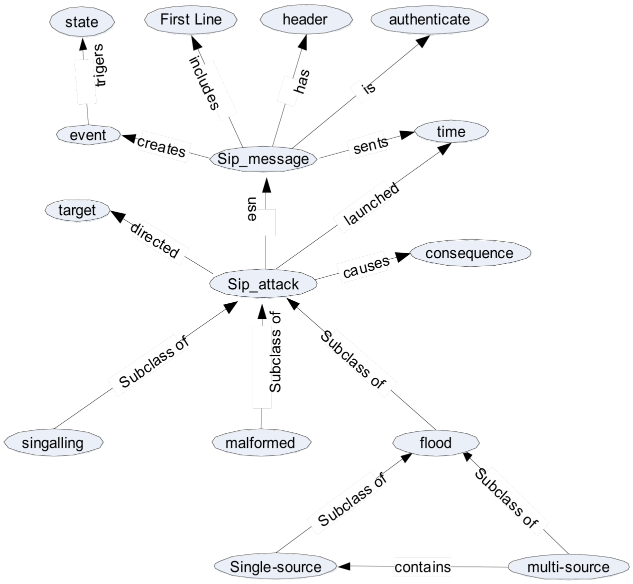


Figure 3. The Abstract Ontology Representation of the Proposed Model

The model has been ported into Web Ontology Language (OWL) [26] exploiting also the Protégé API (http://protege.stanford.edu/) and Semantic Web Rule Language (SWRL) [27]. This gives more expressiveness in the model and reasoning over the ontology, thus developing an accurate validation mechanism. For demonstration purposes SIP raw data have been transformed into the appropriate structure, as mandated by the model, in order to inference about the validity of demo messages according to a specific security policy. As a result any data not compliant with the FOL model could not be embedded as part of the model and is characterized automatically as malicious. At this point we assume that an administrator of an SIP realm has employed the following security policy:

- *Sec-Policy-Guide-1*: SIP Messages longer than 140 bytes should not be processed (dropped).
- *Sec-Policy-Guide-2*: A single user is not allowed to simultaneously send identical messages in the VoIP service.
- *Sec-Policy-Guide-3*: A single user is not allowed to send the same message within a time frame of 10 seconds.

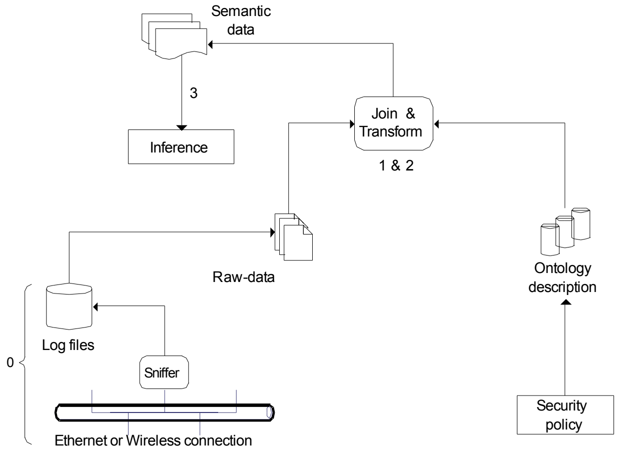
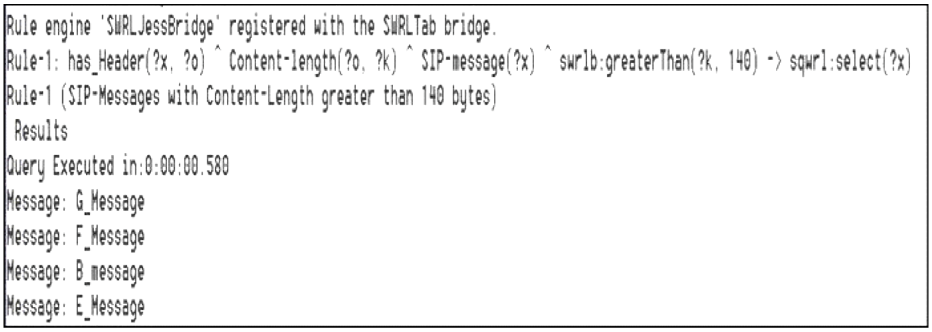


Figure 4. General Procedure to Inference about the Validity of SIP based VoIP Service Data

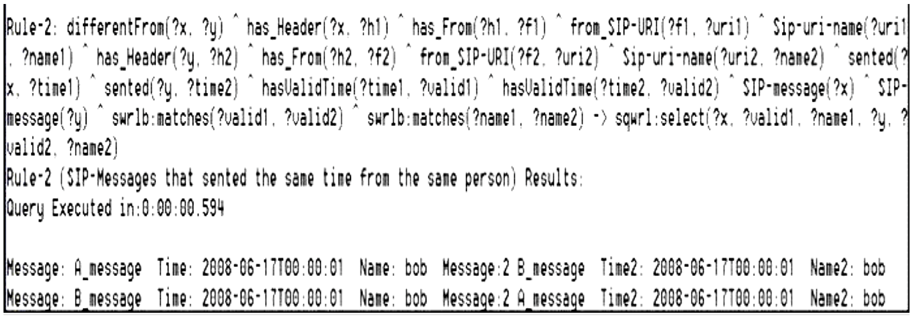
In order to identify any inconsistencies or misbehaviors in the SIP logged data, according to the aforementioned security policy, we employed three SWRL queries as illustrated in Table 1. Specifically, *Query-1* identifies all the SIP messages that are longer than 140 bytes. This is achieved by utilizing the information of Content-Length header of the transformed data which contains the overall SIP message length. Figure 5 shows the results of inference of *Query-1*on the demo data i.e., four messages are found to violate the *Sec-Policy-Guide-1*.

The remaining queries namely *Query-2* and *Query-3* are used to identify any inconsistencies according to *Sec-Policy-Guide-2* and *Sec-Policy-Guide-3*correspondingly. *Query-2* detects duplicate messages, which actually have been sent by the same user. This is done by correlating the "received" time of a SIP message and the data included in the "from" header of the demo message. Figure 6 depicts the inference results on the demo data, in which user bob has been identified for sending the same message at the same time. Similarly, *Query-3* searches for same messages that have been sent during a time frame of 10 seconds. Figure 7 shows the result of the execution of Query-3 in which user Alice has been identified as the sender of 3 messages during the time window of 10 seconds.



```
Rule engine 'SWRLJessBridge' registered with the SWRLTab bridge.
Rule-1: has_Header(?x, ?o) ^ Content-length(?o, ?k) ^ SIP-message(?x) ^ swrlb:greaterThan(?k, 140) -> sqwrl:select(?x)
Rule-1 (SIP-Messages with Content-Length greater than 140 bytes)
 Results
Query Executed in:0:00:00.580
Message: G_Message
Message: F_Message
Message: B_message
Message: E_Message
```

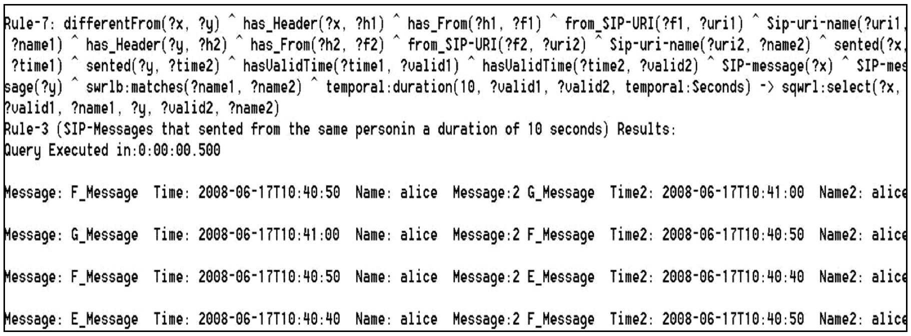
Figure 5. Demonstration of Query 2 Inference



```
Rule-2: differentFrom(?x, ?y) ^ has_Header(?x, ?h1) ^ has_From(?h1, ?f1) ^ from_SIP-URI(?f1, ?uri1) ^ Sip-uri-name(?uri1
, ?name1) ^ has_Header(?y, ?h2) ^ has_From(?h2, ?f2) ^ from_SIP-URI(?f2, ?uri2) ^ Sip-uri-name(?uri2, ?name2) ^ sented(?
x, ?time1) ^ sented(?y, ?time2) ^ hasValidTime(?time1, ?valid1) ^ hasValidTime(?time2, ?valid2) ^ SIP-message(?x) ^ SIP-
message(?y) ^ swrlb:matches(?valid1, ?valid2) ^ swrlb:matches(?name1, ?name2) -> sqwrl:select(?x, ?valid1, ?name1, ?y, ?
valid2, ?name2)
Rule-2 (SIP-Messages that sented the same time from the same person) Results:
Query Executed in:0:00:00.594

Message: A_message  Time: 2008-06-17T00:00:01  Name: bob  Message:2 B_message  Time2: 2008-06-17T00:00:01  Name2: bob
Message: B_message  Time: 2008-06-17T00:00:01  Name: bob  Message:2 A_message  Time2: 2008-06-17T00:00:01  Name2: bob
```

Figure 6. Demonstration of Query 2 Inference



```
Rule-7: differentFrom(?x, ?y) ^ has_Header(?x, ?h1) ^ has_From(?h1, ?f1) ^ from_SIP-URI(?f1, ?uri1) ^ Sip-uri-name(?uri1,
 ?name1) ^ has_Header(?y, ?h2) ^ has_From(?h2, ?f2) ^ from_SIP-URI(?f2, ?uri2) ^ Sip-uri-name(?uri2, ?name2) ^ sented(?x,
 ?time1) ^ sented(?y, ?time2) ^ hasValidTime(?time1, ?valid1) ^ hasValidTime(?time2, ?valid2) ^ SIP-message(?x) ^ SIP-mes
sage(?y) ^ swrlb:matches(?name1, ?name2) ^ temporal:duration(10, ?valid1, ?valid2, temporal:Seconds) -> sqwrl:select(?x,
?valid1, ?name1, ?y, ?valid2, ?name2)
Rule-3 (SIP-Messages that sented from the same personin a duration of 10 seconds) Results:
Query Executed in:0:00:00.500

Message: F_Message  Time: 2008-06-17T10:40:50  Name: alice  Message:2 G_Message  Time2: 2008-06-17T10:41:00  Name2: alice

Message: G_Message  Time: 2008-06-17T10:41:00  Name: alice  Message:2 F_Message  Time2: 2008-06-17T10:40:50  Name2: alice

Message: F_Message  Time: 2008-06-17T10:40:50  Name: alice  Message:2 E_Message  Time2: 2008-06-17T10:40:40  Name2: alice

Message: E_Message  Time: 2008-06-17T10:40:40  Name: alice  Message:2 F_Message  Time2: 2008-06-17T10:40:50  Name2: alice
```

Figure 7. Demonstration of Query 3 Inference

Query Number	SWRL Query
Query-1	SIP-message(?x) ^ has_Header(?x, ?o) ^ Content-length(?o, ?k) ^ swrlb:greaterThan(?k, 140) → sqwrl:select (?x)
Query-2	SIP-message(?x) ^ SIP-message(?y) ^ differentFrom(?x, ?y) ^ has_Header(?x, ?h1) ^ has_From(?h1, ?f1) ^ from_SIP-URI(?f1, ?uri1) ^ Sip-uri-name(?uri1, ?name1) ^has_Header(?y, ?h2) ^ has_From(?h2, ?f2) ^ from_SIP-URI(?f2, ?uri2) ^ Sip-uri-name(?uri2, ?name2) ^ sented(?x, ?time1) ^ sented(?y,?time2)^hasValidTime(?time1, ?valid1) ^ hasValidTime(?time2, ?valid2) ^ swrlb:matches(?valid1, ?valid2) ^ swrlb:matches(?name1, ?name2) → sqwrl:select(?x, ?valid1, ?name1, ?y, ?valid2, ?name2)
Query-3	SIP-message(?x) ^ SIP-message(?y) ^ differentFrom(?x, ?y) ^ has_Header(?x, ?h1) ^ has_From(?h1, ?f1) ^ from_SIP-URI(?f1, ?uri1) ^ Sip-uri-name(?uri1, ?name1) ^ has_Header(?y, ?h2) ^ has_From(?h2, ?f2) ^ from_SIP-URI(?f2, ?uri2) ^ Sip-uri-name(?uri2, ?name2) ^ sented(?x, ?time1) ^ sented(?y, ?time2) ^ hasValidTime(?time1, ?valid1) ^ hasValidTime(?time2, ?valid2) ^ swrlb:matches(?name1, ?name2) ^ temporal:duration(10, ?valid1, ?valid2, temporal:Seconds) → sqwrl:select(?x, ?valid1, ?name1, ?y, ?valid2, ?name2)

Table 1. Examples of Inference Queries in SWRL

Summarizing, the presented practical tool is capable of analyzing the logged VoIP data effectively in order to identify security inconsistencies, by exploiting our security model. Although the analysis of the logged data is performed off-line, the processing time, as depicted in Figures 5 to 7, is about 500 ms (searching in 100 records).

V. Conclusions and Future Work

The analysis of the logged data for critical real time applications like VoIP could be considered as a valuable tool for enhancing and improving the security level of the provided service. Furthermore, malicious messages that may bypass the underlying security mechanisms can not be identified or recognized through some different method. In this work we introduce a novel system that is able to analyze logged data in order to validate the accuracy of the employed mechanisms and the conformance with the corresponding security policy. In our future work we would like to consider a more holistic view for our model, incorporating Internet architecture protocols like UDP and IP, as well as modeling the corresponding security flaws. This will give us the opportunity to develop a multilevel formal model for automatically validating security policies in the Internet architecture.

References

[1] Varshney, U., Snow, A., McGivern, M., and Howard, C. "Voice over IP", Communications of the ACM, January 2002.

[2] Hersent, O.; Petit J.; Gurle D., IP Telephony: Deploying Voice-over-IP Protocols, Wiley, March 2005

[3] Rosenberg, J.; Schulzrinne H.; Camarillo, G.; Johnston, A.; Peterson, J.; Spark, R.; Handley, M.; Schooler E., “Session Initiation Protocol”, RFC 3261, June 2002.

[4] http://www.cisco.com/en/US/tech/tk652/tk701/tk589/tsd_technology_support_sub-protocol_home.html

[5] Schulzrinne H.; Casner S.; Frederick R.; Jacobson, V., RTP: A Transport Protocol for Real-Time Applications, July 2003

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

[7] Sisalem, D.; Kuthan, J.; Ehlert, S., "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

[8] Geneiatakis D., Lambrinoudakis C., "A Lightweight Protection Mechanism against Signaling Attacks in a SIP-Based VoIP Environment", Telecommunication Systems, Vo 36, No 4, pp. 153-159, Springer, February 2008,

[9] Geneiatakis D., Kambourakis, G., Lambrinoudakis, C., Dagiouklas, A., Gritzalis S., "A framework for protecting SIP-based infrastructure against Malformed Message Attacks", Computer Networks, Vo. 51, No. 10, pp. 2580-2593, 2007, Elsevier

[10] Cao, F., Jennings, C., "Providing response identity and authentication in IP telephony", in the proceedings of The First International Conference on Availability, Reliability and Security, April 2006

[11] Yang, Chou-Chen., Wang, Ren-Chiun., Liu, Wei-Ting., "Secure authentication scheme for session initiation protocol", Computers & Security, Vol.24, Iss.5, August 2005.

[12] Nagpal, S., Yardeni, E., Schulzrinne, H., and Ormazabal, G. "Secure SIP: A Scalable Prevention Mechanism for DoS Attacks on SIP-based VoIP Systems." In Principles, Systems and Applications of IP Telecommunications (IPTComm2008), July 2008

[13] Fiedler, J., Kupka, T., Ehlert, S., Magedanz, T., Sisalem, D., "VoIP Defender: Highly Scalable SIP-based Security Architecture", in the proceeding of Principles, Systems and Applications of IP Telecommunications (IPTComm2007) Conference, July 2007.

[14] Chen, E.Y., "Detecting DoS attacks on SIP systems", in the proceedings of 1st IEEE Workshop on VoIP Management and Security, April 2006.

[15] Sengar, H., Wang, H., Wijesekera, D., and Jajodia, S. 2008. Detecting VoIP Floods using the Hellinger Distance. IEEE Transactions on Parallel and Distributed Systems 19, 6 (June 2008), 794-805.

[16] Zhang, G., Ehlert S., Magedanz, T., Sisalem D., "Denial of Service Attack and Prevention on SIP VoIP Infrastructures Using DNS Flooding", in the proceeding of *Principles, Systems and Applications of IP Telecommunications (IPTComm2007) Conference,* July 2007.

[17] Baugher, M., McGrew, D., Naslund, M., Carrara, E., Norrman K., "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[18] Diana Senn, David Basin, Germano Carroni, Firewall Conformance Testing, in proceedings of Testing of Communicating Systems, May 2005

[19] Haralambos Mouratidis, Paolo Giorgini, Security Attack Testing (SAT)--testing the security of information systems at design time, Information SystemsVolume 32, Issue 8, , December 2007, Pages 1166-1183

[20] Yliès Falcone, Jean-Claude Fernandez, Laurent Mounier, Jean-Luc Richier, "A Test Calculus Framework Applied to Network Security Policies", in proceedings of Formal Approaches to Software Testing and Runtime Verification, 2006

[21] Wieser C., Laakso M., Schulzrinne H., “Security testing of SIP implementations”, available on line in: http://compose.labri.fr/documentation/sip/Documentation/Papers/Security/Papers/462.pdf, 2003

[22] Abdelnur, H.; Cridlig, V.; State, R.; Festor, O., "VoIP security assessment: methods and tools," in the proceedings of 1st IEEE Workshop on VoIP Management and Security, pp. 29-34, 3 April 2006

[23] Abdelnur, H.; State, R.; Chrisment, I.; Popi, C., "Assessing the security of VoIP Services," Integrated Network Management, 2007. IM '07. 10th IFIP/IEEE International Symposium on, vol., no., pp.373-382, May 21 2007-Yearly 25 2007.

[24] Theodore Tryfonas, Iain Sutherland, Ioannis Pompogiatzis, "Employing penetration testing as an audit methodology for the security review of VoIP: Tests and examples", vol. 17, Iss. 1, pp:61-87 Internet Research, 2007

[25] Geneiatakis D., Lambrinoudakis C. and Kambourakis G., "An Ontology Based-Policy for Deploying Secure SIP- based VoIP Services", to appear in Computer and Security, 2008, Elsevier, available on line: doi:10.1016/j.cose.2008.07.002

[26] McGuinness, D. L.; Frank van Harmelen, OWL Web Ontology Language, http://www.w3.org/TR/owl-features/

[27] Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, Mike Dean, SWRL: A Semantic Web Rule Language Combining OWL and RuleML, http://www.w3.org/Submission/SWRL/

[28] Geneiatakis, D., Lambrinoudakis, C., "An Ontology Description for SIP Security Flaws", Computer Communication, Computer Communication, Vo. 30, No. 6, pp. 1367-1374, 2007, Elsevier