available at www.sciencedirect.com

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Utilizing bloom filters for detecting flooding attacks against SIP based services

*Dimitris Geneiatakis**, *Nikos Vrakas*, *Costas Lambrinoudakis*

Laboratory of Information and Communication Systems Security, Department of Information and Communication Systems Engineering, University of the Aegean, Karlovassi, GR-83200 Samos, Greece

ARTICLE INFO

Article history:
Received 12 August 2007
Received in revised form
26 April 2009
Accepted 27 April 2009

Keywords:
Session Initiation Protocol (SIP)
Voice over IP (VoIP)
Flooding attacks
Denial of Service
Bloom filter
Security

ABSTRACT

Any application or service utilizing the Internet is exposed to both general Internet attacks and other specific ones. Most of the times the latter are exploiting a vulnerability or misconfiguration in the provided service and/or in the utilized protocol itself. Consequently, the employment of critical services, like Voice over IP (VoIP) services, over the Internet is vulnerable to such attacks and, on top of that, they offer a field for new attacks or variations of existing ones. Among the various threats–attacks that a service provider should consider are the flooding attacks, at the signaling level, which are very similar to those against TCP servers but have emerged at the application level of the Internet architecture. This paper examines flooding attacks against VoIP architectures that employ the Session Initiation Protocol (SIP) as their signaling protocol. The focus is on the design and implementation of the appropriate detection method. Specifically, a bloom filter based monitor is presented and a new metric, named *session distance*, is introduced in order to provide an effective protection scheme against flooding attacks. The proposed scheme is evaluated through experimental test bed architecture under different scenarios. The results of the evaluation demonstrate that the required time to detect such an attack is negligible and also that the number of false alarms is close to zero.

1. Introduction

The World Wide Web (WWW) has been designed to provide a mean for sharing information among specific communities, utilizing a ''unified'' open network as the Internet. At the early stages of Internet deployment no security considerations were taken into account. This fact has been exploited in several ways by many malicious users who have caused numerous incidents, one of them being Denial of Service (DoS) (i.e. they have stopped illegally the communication among the communicating parties). By the term of DoS is identified any attempt by malicious users trying to constitute the provided service unavailable to legitimate users. The existence of such a problem was firstly pinpointed in Gligor (1984), in which the malicious users focus on the vulnerabilities of the operating system. Generally DoS attacks could have two major forms. In the first one, the malicious user crafts very carefully a packet trying to exploit vulnerabilities in the implemented software (service or a protocol). Among the most known attacks of this category are the buffer overflow attacks while an incident of such an attack is the Ping of Death. In the second form, the malicious user is trying to overwhelm system's resources of the provided service-like memory, CPU or bandwidth, by creating numerous of useless well-formed requests. This type of attack is well known as flooding attack. The most known flooding attack against web servers is reported in Gibson

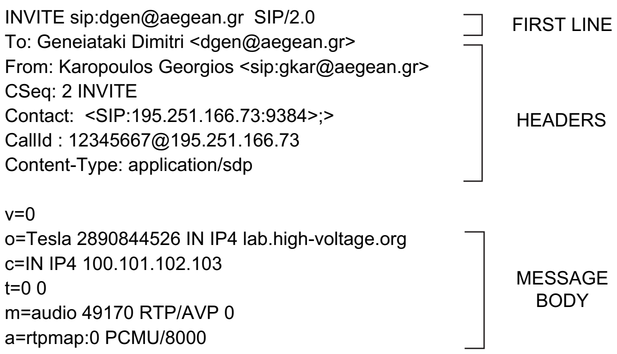
* *Corresponding author*. Tel.: +30 22730 82247; fax: +30 22730 82009.
E-mail addresses: dgen@aegean.gr (D. Geneiatakis), nvrak@aegean.gr (N. Vrakas), clam@aegean.gr (C. Lambrinoudakis).

0167-4048/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cose.2009.04.007

(2002). A detailed analysis of DoS attacks in Internet services can be found in Carl et al. (2006), Peng et al. (2007) and Mirkovic et al. (2004).

Consequently, any application and/or service utilizing the Internet's open architecture is constituted susceptible to similar attacks. It is therefore reasonable for every critical real-time application to treat architectures like the Internet as hostile environments. This is, or at least should be, the case for Voice over IP (VoIP) services offered over the Internet. Furthermore, various researchers have already identified security vulnerabilities in those systems (Endler et al., 2005; Sisalem et al., 2005; Geneiatakis et al., 2006; Ming et al., 2008). On the contrary Public Switch Telephone Network (PSTN) security flaws are considered minimal, due to its closed network architecture. Consequently, in PSTN it is considered very difficult to launch any kind of attack provided that the attacker has no physical access to the network. This is not the case for VoIP based Internet services, since a vulnerability in the service/protocol can be exploited by utilizing various VoIP security tools like SIP swiss army knife (sipsak), PROTOS testing suite (Wieser et al., 2003) or by developing specific tools as demonstrated in Endler et al. (2005), Sisalem et al. (2005) and Geneiatakis et al. (2006), without requiring physical access to the medium. Additionally, a side effect of the interconnection between VoIP and PSTN is that PSTN infrastructures become vulnerable to VoIP attacks. Consider, for instance, that a VoIP service is used as an intermediate for launching a flooding attack against the PSTN infrastructure. Such a scenario was not feasible a few years ago.

It is therefore clear that the protection of VoIP systems against flooding attacks is crucial not only to ensure that the trust and security levels offered are similar to those offered by PSTN architectures, but also to protect the PSTN system itself against flooding attacks. In the work presented here we examine the case of flooding attacks against VoIP systems that employ the Session Initiation Protocol (SIP) (Rosenberg et al., 2002) for call management. For protecting SIP proxies' servers against flooding attacks, we propose a monitor system that is based on bloom filters combined with a new metric, named *session distance*. Our main focus is on SIP as it seems to overwhelm the other signaling protocols (H.323, MGCP) and it has been adopted by various standardization (e.g. 3GPP, IETF) organizations as the protocol to establish multimedia sessions at both wireline and wireless world in the Next Generation Networks (NGN) era.



```
INVITE sip:dgen@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 lab.high-voltage.org
c=IN IP4 100.101.102.103
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

FIRST LINE
HEADERS
MESSAGE BODY

Fig. 1 – A typical SIP-INVITE request.

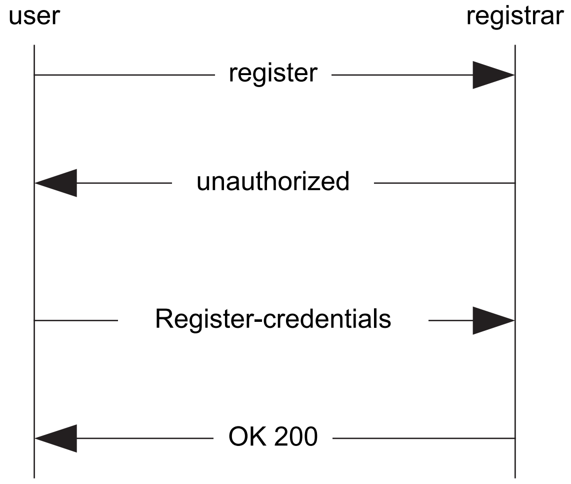


Fig. 2 – Registration procedure.

The rest of the paper is structured as follows. Section 2 provides background information on the SIP protocol. Section 3 presents various types of flooding attacks against SIP based VoIP systems while Section 4 demonstrates their consequences. Section 5 presents and evaluates a novel detection method for flooding attacks, which is based on a *bloom filter*. Finally Section 6 acquaints with related work and Section 7 concludes the paper giving some pointers for future work.

2. The Session Initiation Protocol

Session Initiation Protocol (SIP) is an application-layer signaling protocol for creating, modifying, and terminating multimedia sessions among one or more participants (Rosenberg et al., 2002). The general structure of the SIP protocol is inherited by Hyper Text Transfer Protocol (HTTP) (Fielding et al., 1999) and thus SIP messages are text based similar to the HTTP ones. Specifically, a SIP message can be either a request or the corresponding response, depending on the first line of the message, followed by the appropriate headers required to describe the request or the response, and the message body. The message body is optional and its existence depends on the request. Fig. 1 illustrates an example

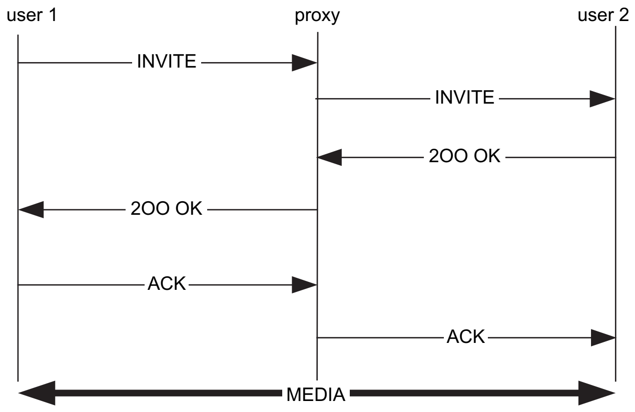


Fig. 3 – SIP multimedia connection establishment.