Importance-Based Security Level Verification in Web Services

Pham Phuoc Hung, Aziz Nasridinov, Jeongyong Byun
Department of Computer and Multimedia, Dongguk University
e-mail :hung205a2@yahoo.com, aziz_nasridinov@yahoo.com, byunjy@dongguk.ac.kr

웹 서비스에서 중요도 기반 보안수준 검증

밤복흥, 아지즈, 변정용
동국대학교 컴퓨터멀티미디어학부

Abstract

There are some cases when SOAP message, where WS-Security and WS-Policy elements are included, may consist of a sensitive and important data. In these cases, the message is highly recommended to be secured. The question exists of how to quickly identify that SOAP message satisfies security requirement and security level of a SOAP message. In this paper, we propose a technique called Bit-Stream which depends on the importance of SOAP elements to automatically identify the vulnerabilities and risks while offering advice for higher security.

1. Introduction

SOAP message sometimes contains the important and sensitive data. So, it is necessary to apply security for web service depending on the company's policy. In these cases, the need for secure communication in service-oriented architectures leads to several new specifications enabling security aspects for web service messages [1]. The question is how to identify which SOAP is not satisfied security requirement quickly and how to determine the security level of a SOAP message quickly and exactly in order to prevent vulnerabilities against the original goals, easily auditing and reviewing the security critical part of the system sources. In this paper, we propose a technique called Bit-Stream to automatically identify the vulnerabilities and risks while offering advices for higher security such as suggesting corrective actions. It is a rule-based technique which includes WS-Security policies for detecting typical error in SOAP and determining the security level of a SOAP message.

Advantages of this technique are:

- It's easy to recognize which element in SOAP message is not valid while assisting developer to understand and to improve their web service system.
- The importance of each element is different so we can quickly see where the most necessary element to correct the SOAP message is.
- It's convenient to validate the security level of a SOAP message more quickly and exactly with a specific value.

This paper is organized in following way. In Section 2 related studies are discussed. Section 3 illustrates the Motivating scenario and Section 4 presents system design. Section 5 presents Analysis and the paper will end with conclusion and future work.

2. Related Studies

Similar classification and functional relationships were explored in various discovery working groups.

In [2], authors propose approach with a tool giving a simplified view to its users, who are configuring secure Web Services for their systems. However, the tool has not been integrated with an actual development process.

In [3],[4] researchers describe the design of an advisor for web services security configurations, the tool to identify vulnerabilities automatically. The main point of this research is to find the element is not suggested in SOAP request. But the disadvantage of this approach is that we cannot identify security level from the security report immediately and exactly. It costs time to analysis the report result and hard to remember. On contrast, our system currently supports conveniently, quickly and more exactly to validate the security level of a SOAP message with a specific value.

3. Motivating Scenario

Suppose that Auto Repair Shop (ARS) connect to the Retailer Shop (RS) to buy some parts of car. To buy a part, they have to follow the steps as shown in figure 1:

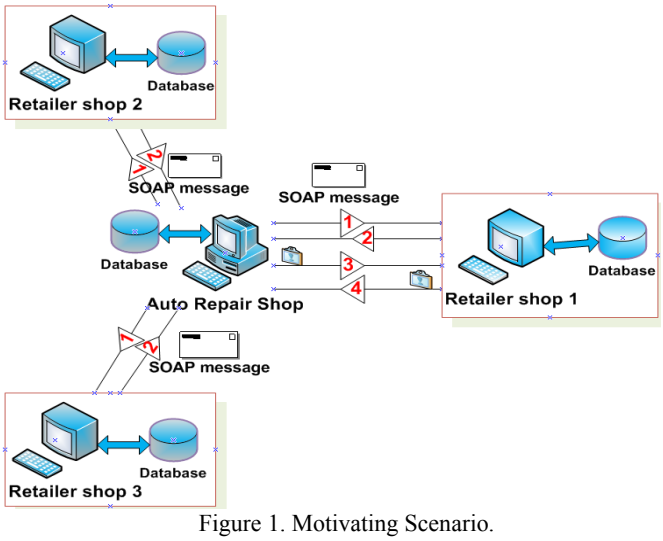

Figure 1. Motivating Scenario.

1. ARS sends message to many RS to buy a tire.
2. When RS receive the SOAP message from ARS, RS checks their inventory database and replies, "The tire is available"
3. After ARS receives response from RS, ARS decides where to buy and then send an order-message which includes their personal information, encrypted credit card information and so on to the RS.
4. Depending on the company's policy, RS will decrypt the received message, then choose the appropriate information for VIP, common customer, encrypt it and send it to ARS.

The problems in this scenario are that someone wants to change the information from common customer into VIP customer during the message transferring. RS want to avoid someone who is not their members. They want to avoid some IP address and so on. A lot of security requirements have to be satisfied. Therefore, it's difficult to be sure that the Web service gets the security level which we want.

4. System design

Figure 2 describe the general design of the proposed system. We will divide explanation into 4 parts as following:

(1) First, we have to have security requirements depending on the company's policy. Then write WS–security policies depending on those security requirements in a policy file. Because the importance of the elements of a SOAP message is different, we can sort them in the decreasing order.

(2) Mapping file includes a set of mappings and WS–security policies. Each mapping associates a SOAP message to a policy.

(3) Bit-Stream technique will read inputs from policy file, mapping file, SOAP request. Depend on the policies in policy file, it will find the match between policy and SOAP request element. For each match, we define its value as 1. Otherwise, its value is 0. The number of matching can be counted to get security level of the SOAP message.

(4) As a result, we have a bit stream which has the different important bits depending on the position of the bit. Left bit is more important and get higher security. Right bit is less important and get lower security.

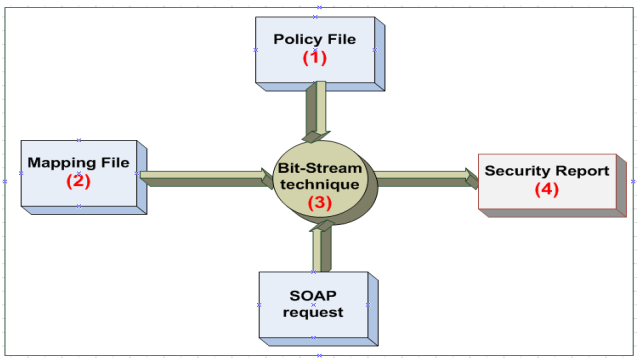

Figure 2. System Design

5. Analysis

Assume that a Web service require some security requirement as following. In SOAP request, the importance of SOAP elements are different and in decreasing order. Body element has to be existed, SOAP header has to have MessageID, To, Action element. SOAP header doesn't have to have From, FaultTo element. These requirements are summarized in the next table:

Table 1. Summary of SOAP security requirements

Importance level	SOAP element	Require
15	Wsp:Body()	1
10	Wsp:Header(wsa:From)	0
10	Wsp:Header(wsa:MessageID)	1
9	Wsp:Header(wsa:FaultTo)	0
5	Wsp:Header(wsa:Action)	1
3	Wsp:Header(wsa:To)	1

Table 2. Comparing results after matching in Bit-Stream

Importance level	SOAP element	Require	Detect	Result
15	Wsp:Body()	1	1	1
10	Wsp:Header(wsa:From)	0	0	1
10	Wsp:Header(wsa:MessageID)	1	0	0
9	Wsp:Header(wsa:FaultTo)	0	0	1
5	Wsp:Header(wsa:Action)	1	0	0
3	Wsp:Header(wsa:To)	1	0	0

The result is 110100. With this bit stream, we can quickly recognize which bit is 0. Therefore, we can know which corresponding SOAP element is not suggested, not match policies. Then, we can fix it more valuable. Left bits are more important and get higher security then right bits. Depend on the company policies, we can make the priority to correct the SOAP request element which is more important or not to reduce vulnerabilities.

6. Conclusion and future work

In this research, we achieve a Bit-Stream technique which is based on rules in order to detect, validate the security level of a Web service more quickly, exactly and convenient. The technique finds the match between policy and SOAP elements to verify they correctly implement the intended security goals.

In some big system, diversity of policies leads to conflicts between policies. The need to resolve priorities between policies for getting more exactly security level is the problem we will solve in the future.

Reference

[1] Nils Gruschka, Meiko Jensen, Torben Dziuk," Event-based Application of WS-SecurityPolicy on SOAP Messages", *SWS'07,* November 2, 2007, Fairfax, Virginia, USA

[2] Michiaki Tatsubori, Takeshi Imamura, Yuhichi Nakamura,"Best practice patterns and tool support for configring secure web services messaging**",** Proceedings of the IEEE International Conference on Web Services (ICWS'04)

[3] Karthikeyan Bhargavan, C´edric Fournet , Andrew D. Gordon, Greg O'Shea, "An Advisor for Web Services Security Policies", *SWS'05,* November 11, 2005, Fairfax, Virginia, USA.

[4] Karthikeyan Bhargavan, C´edric Fournet, Andrew D. Gordon, "Verifying PolicyBased Security for Web Services", *CCS'04,* October 25-29,2004, Washington, DC, USA.