Detection of XML Signature Wrapping Attack Using Node Counting

Abhinav Nath Gupta and P. Santhi Thilagam

Abstract In context of web service security, several standards are defined to secure exchanges of SOAP messages in web service environment. Prominent among these security standards is the digital signature. SOAP messages are signed partially or fully before being transmitted. But recent researches has shown that even signed messages are vulnerable to interception and manipulation of content. We refer to these types of attacks as XML signature wrapping attacks. In this paper, an approach is proposed to detect the XML signature wrapping attacks on signed web service requests using node counting. We detect XML signature wrapping attacks by calculating the frequency of each node in web service request. Experiments show that the proposed solution is computationally less expensive and has better performance in securing the exchange of SOAP messages.

Keywords XML digital signature · XML signature wrapping · Web services

1 Introduction

Web Service is generally used to describe web resources that are accessed by the software applications rather than users. Web services are a set of functionalities designed to work in collaboration to complete a task. Web services standardized the business applications and due to the code reusability and interoperability feature provided, the business environment is falling for web services. Due to extensive usage of web services in business scenario it gives enough importance to Web Services to raise the security concerns.

Making Web Services secure means making SOAP messages secure and keeping them secure wherever they go [1]. The group of security standards in WS-Security is used to secure exchanges of SOAP messages in Web Service environment. However, despite all of these security mechanisms, certain attacks on

A.N. Gupta (✉) · P. Santhi Thilagam
Department of Computer Science and Engineering, NITK Surathkal,
Mangalore 575025, India

V. Vijayakumar and V. Neelanarayanan (eds.), *Proceedings of the 3rd International Symposium on Big Data and Cloud Computing Challenges (ISBCC – 16')*,
Smart Innovation, Systems and Technologies 49, DOI 10.1007/978-3-319-30348-2_5

SOAP messages may still occur and lead to significant security faults [2]. Illustrated that the SOAP message, protected by an XML Digital Signature as specified in WS-Security, can be modified without invalidating the signature. These kind of attacks are called XML Signature Wrapping Attacks. They can happen because XML Digital Signature assigned to an object in an XML document does not depend on the location of the object in the document.

Moreover, SOAP extensibility model by default has a very less restriction of the presence of headers and elements inside soap message and hence an unrecognized security header or soap header can be present inside soap message. All of these features along with vulnerabilities of XML Digital Signature gives away a way for performing wrapping attacks on SOAP messages.

Different solutions have been proposed to solve this problem. For example, in Ref. [3] the authors proposed an inline approach that uses the structure information of the SOAP message by adding a new header element called SOAP Account. In Ref. [4] the authors extended the inline approach by considering not only structure but depth and parent child relationships in soap message. However, none of these solutions could properly detect wrapping attacks. Moreover, not much attention has been laid on how to recover from the attack.

Section 2 describes the related work done on the same topic. Section 3 gives the details of system design. Section 4 describes the experimental setup and performance evaluation against other popular approaches to counter xml signature wrapping attack. Section 5 is conclusion followed by references considered for the study.

2 Related Work

To secure the exchange of XML document over the web, to maintain the authenticity and integrity of the exchange, XML digital signature is deployed [2]. XML Digital signature provides mechanism to encrypt an XML document partially or fully thereby securing the content of the XML document. In Fig. 1 [3] the structure of signed XML document is given.

The major vulnerability of xml digital signatures which leads to xml signature wrapping attack is xml processing is done twice when xml digital signature is present: once for the validation, and once for application use. Issue is that, for each case, validation and application, different approach is used to access xml data. XML Signature validation finds the signature element and use the references id inside to locate the signed element. The application parser instead analyze the message thoroughly to find the data application is interested in. Generally the results are same, but in the signature wrapping case, the attacker replaces the original signed element by a fake and relocating the original element inside the soap message from its original place.

SOAP message is depicted in Fig. 1 [3]. The figure shows the function of deleting the user present in the SOAP message body. Authentication and integration

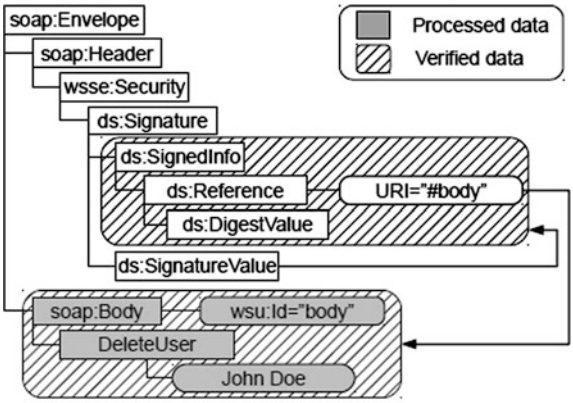

Fig. 1 Example of XML signature applied on the SOAP body

of the SOAP body is done with the help of XML signature. XML signature is present in SOAP header consisting two elements <SignedInfo> which is the identification pointing to the SOAP body and digest value computed over cited element and <SignatureValue>. The authentication for <SignedInfo> is provided by computing the signature value and assigning it to <SignatureValue>. The end-user first searches for the cited element given in <SignedInfo>. Than the digest value over the cited element is computed and comparison is done with the value given in the <DigestValue>. Than the signature is verified using <SignedInfo>. At the end, function defined in the SOAP body is executed. This was first observed by McIntosh and Austel [2], Example of the XML Signature Wrapping attack is shown in Fig. 2 [3]. In this example an attacker moves the original body of the SOAP inside header. Then he creates a body of the SOAP new id and invoke different function. Since the Signature is not altered just relocated and the concerning parts

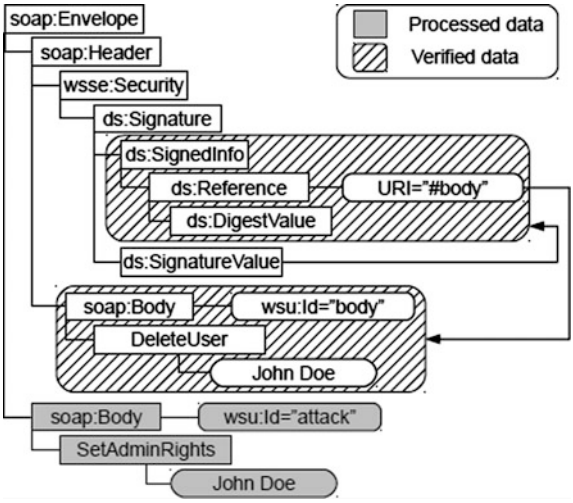

Fig. 2 Example of XML signature wrapping attack on the SOAP body

are also unchanged, the security logic will be able to do the verification of integrity and authenticity. This new SOAP body is taken as the input by the business logics.

XML Signature Wrapping attack is a newly discovered attack and as explained above, this attack exploits the loophole in processing of XML signature, which is designed for the purpose of authentication and integrity of the request message, to inject malicious data inside the request. Only a handful of papers are available to provide insight of this attack and even less counter measures. McIntosh and Austel [2] show ways to defend against wrapping attacks by laying certain message exchange policies for both sender and receiver. These policies have to be hardcoded into the application. But due to this the advantages of service oriented architectures is lost as the web services no longer remain independent.

Another popular countermeasure is XML Schema validation [3]. However, performing schema validation in the Web Services framework is not suitable, since it could adversely affect the performance of the service. Furthermore xml schema validation doesn't provide good security against the xml signature wrapping attack.

Second category of proposed countermeasures is called the inline approach and was presented in [3]. This approach fixes the relative location of the signed element so that any movement or alteration is detected. This approach works by adding a header element called SOAP account for each signed element containing its number of child element, parent elements, depth from SOAP header and Envelope.

But this idea has some disadvantages, first among those is its not being a standard of xml. Secondly, attacker could alter the message content while still being validated successfully. That means that inline approach cannot prevent against signature wrapping attacks in general.

Other approaches like schema hardening [3], attaching XPath expressions and ontology based approaches in [3]. But All these approaches have to be hardcoded into the application itself, and it means loss of flexibility and independence of service oriented architecture. All the approaches defined above are proactive approaches and demand a good understanding of the system from both the client side and service provider side.

3 System Design

The System is divided into 3 main modules, interception module, detection module, and logging module. First module is basically the interception module which intercepts the incoming digitally signed SOAP requests and forward it to the detection module.

Detection module then applies the algorithm given below to detect the presence of the XML signature wrapping attacks by analysing the request thoroughly. If detected, the request is denied and log is generated via logging module else the request is forwarded to intended recipient and another log is generated of successful forwarding of the request.

The Algorithm to detect the XML signature wrapping is

Input: InputStream of SOAP Envelope

1. See if <Signature> is child of <Header> element using//Signature//Reference [@URI] xpath expression, if present move to step 2 else there is no attack.
2. If found, extract the URI attached with the Reference attribute using string api in java, and check if it begins with #, if it does move to step 3 else match the URI against the URI Signatures.
3. If the URI begins with #, check the SOAP Header for optional header elements using//*[contains(@mustUnderstand,′0′)]/*/*xpath expression. It also extract all the children of the optional header element and save it in a NodeList data structure from java xpath api.
4. Extract all the children of SOAP Body element including itself using/Envelope/Body/*and save them in another NodeList.
5. Now compare NodeList from above two steps by counting the frequency of child nodes, if a single element is common in both list there is XML Signature Wrapping attack, else no attack.

Output: Boolean value notifying the presence of attack vector.

This concludes the algorithm to detect the XML signature wrapping attack.

4 Implementation and Performance Evaluation

We have implemented 3 services as per the specifications provided by a popular benchmark for web services named TPC-APP, and deployed them on Apache Tomcat using Axis2 service engine.

Test Environment machine composed of Hardware: 100 Mbps Ethernet card, 4 GB memory, Intel core i5 cpu with clock speed of 2.40 GHz, and Software: Windows 7 Home Basic Edition, Java2 Standard Edition jdk 1.7.0. All the implementation including web services and WS-IDS is in java.

Colored Petri Net (CPN) tools version 4.0 are used for the simulation of XML signature wrapping attack in Lab Environment. Using the CPN tools we have simulated 100 web services request messages which contains 20 malicious request of namespace injection, 20 each of both id based and xpath based signature wrapping attacks and rest of the request are all valid. Several attacking cases including Replay Attack, Redirection Attack and Multiple Header Attacks is simulated inside the header and body of service requests along with attacks specific to XML signature wrapping. We have also considered the specific case of XML signature wrapping attacks and also sub types of the attack itself like namespace injection or id based wrapping. The system starts by intercepting the incoming SOAP requests and analysing them according to three approaches, i.e., WS Security, SOAP Account and Node Counting to detect the XML signature wrapping attack. The results are presented in Table 1.

Table 1 Performance of node counting against other detection approaches

XML signature wrapping attack detection technique	False positive	True positive
WS security	8	52
Node counting	0	60
SOAP account	16	44

The results presented in Table 1 makes it is easy to recognize that our approach has bettered the other approaches. WS-Security approaches show the lowest attacks detection because as we mentioned in Sect. 2, XML Digital Signature has limitations to protect SOAP message from wrapping-attacks.

The next, SOAP Account approaches show the second best result. This is because with this approach, analysis of SOAP account, a header for each referenced element contains the number of child nodes of that element, would decrease the performance of the approach. Results may vary on different configurations and more rigorous testing, but while comparing Node Counting approach with other approaches it is noticed that Node Counting is only approach independent of any prior interaction to web service client or web service provider of any kind. In case of SOAP Account approach a separate header element for each of referenced element has to be added inside SOAP Header which has to be analysed at the receiving end in order to detect any XML Signature attack, a computationally expensive task.

We lose the independency and flexibility of web services as it is not a standard yet. Same is the case with WS-Security approach in which an xpath expression is included to locate the referenced element, again expensive in terms of computation cost. Also it has to be noted that even though approach bettered other approaches, it cannot detect 100 % of attacks. It is because slight variations in the behaviour of the attack may happen or new attack may appear.

Aside from detection rate of the XML signature wrapping attack, there is one more factor to consider and that is time taken to analyse the service request to detect the presence of XML signature wrapping attack. Node counting approach performs better than WS Security and SOAP Account as it takes less time to analyse the service request to detect the presence of attack as shown in Table 2.

The reason for this time difference is node counting approach directly analysis the received service request without referencing or dereferencing other extensions to the original request like the case in SOAP Account and WS Security Approach.

Table 2 Timed analysis of XML signature wrapping attack detection

XML signature wrapping attack detection technique	Time taken in analysis (ms)
WS security	1200
Node counting	450
SOAP account	700

5 Conclusions

In this paper, we have studied about web services and attacks on web services, specifically about the XML signature wrapping attack. We have studied various vulnerabilities of xpath and XML digital signatures and also considered various scenarios in which those vulnerabilities are exploited to mount signature wrapping attack on web services. We have also studied popular proposed mechanism to counter the XML signature wrapping attacks and there shortcomings in catering the XML signature wrapping attack.

In this paper, we proposed a mechanism based on node counting to combat with XML Signature Wrapping Attacks. Experiments showed that the proposed solutions have better performance in securing the exchange of SOAP message comparing to other methods.

Thus, we believe that our approach can protect SOAP message from wrapping-attacks and therefore, bring a reasonable protection to entire Web Service environment. Our current method may cause a reduction of the effectiveness when slight variations in the behaviors of the attack happen or when new attacks appear.

References

1. Bhargavan, K., Fournet, C., Gordon, A.D.: Verifying policy based security for web services. In: Proceedings of the 11th ACM conference on Computer and communications security, pp. 268–277 (2004)
2. McIntosh, M., Austel, P.: XML signature element wrapping attacks and countermeasures. In: Proceedings of the 2005 Workshop on Secure Web Services, pp. 20–27 (2005)
3. Jensen, M., Meyer, C., Somorovsky J., Schwenk, J.: On the effectiveness of XML schema validation for countering XML signature wrapping attacks. In: 1st International Workshop on Securing Services on the Cloud (IWSSC), pp. 7–13 (2007)
4. Benameur, A., Kadir, F.A., Fenet, S.: XML rewriting attacks: existing solutions and their limitations. IADIS Appl. Comput. **812**(1), 4181–4190 (2008)
5. Bhargavan, K., Fournet, C., Gordon, A.D., O'Shea, G.: An advisor for web services security policies. In: Proceedings of the 2005 Workshop on Secure Web Services, pp. 1–9 (2005)
6. Bartel, M., Boyer, J., Fox, B., LaMacchia, B., Simon, E.: XML-signature syntax and processing. W3C Recomm. **12** (2002)
7. Gajek, S., Liao, L., Schwenk, J.: Breaking and fixing the inline approach. In: Proceedings of the 2007 ACM Workshop on Secure Web Services, pp. 37–43 (2007)
8. Nasridinov, A., Byun, J.-Y., Park, Y.-H.: UNWRAP: An approach on wrapping attack tolerant SOAP messages. In: Second International Conference on Cloud and Green Computing (CGC), pp. 794–798 (2012)