Volume 2, Issue 5, May 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Secure Payment Information Using XML Technology

Ajeet Singh Member IEEE
Dept. of Computer Science
Debre Berhan University
Debre Berhan, Ethiopia
ajeetsinghiet@gmail.com

Karan Singh Member IEEE
School of ICT
Gautama Buddha University
Gr. Noida, India

Shahazad
Dept. of Computer Science
Samara University
Samara, Ethiopia

Azath M.
Dept. of Computer Science
Debre Berhan University
Debre Berhan, Ethiopia

Sathish Kumar Konga
Dept. of Computer Science
Debre Berhan University
Debre Berhan, Ethiopia

Abstract- These days we face a big problem with the payment Information when we send payment Information through Internet to the payee. If the payment information is transmitted over the communication links in plain text form, there is a possibility of eavesdropping. It is possible that somebody listening to the network traffic can get access to sensitive data information, such as card numbers, cvc number, card type and all other information related to the card holder person.

In this paper we will attempt to explain how to secure payment information using XML Technology. The XML Technologies are the XML Signature and Encryption standards are being used extensively as building-block technologies for payment Information.

Keywords: XML Signature and XML Encryption, Dual signatures, triple-DES, RSA.

1. INTRODUCTION

XML was originally developed by the XML Working Group in 1996. XML became a full W3C recommendation in February 1998. XML documents contain either parsed or unparsed character data. Some of the data is composed of characters and some is composed of markup, which describes the layout and structure of information. XML is a markup language much like HTML. XML is a meta-language that describes the content of the document. XML was designed to describe data. XML is not a replacement for HTML [19].

XML tags are not predefined. You must define your own tags. XML was designed to carry data, not to display data. XML is designed to be self-descriptive. Extensible Markup Language (XML) is a set of rules for encoding documents in machine-readable form. It is defined in the XML 1.0 Specification produced by the W3C, and several other related specifications, all gratis open standards [19].

The extensible markup language (XML) is a markup language promoted by the World Wide Web consortium (W3C). XML overcomes the limitations of hypertext markup language (HTML) and represents an important opportunity to solve the problem of protecting information distributed on the Web, with the definition of access restrictions directly on the structure and content of the document [5].

XML Encryption provides end-to-end security for applications that require secure exchange of structured data. XML itself is the most popular technology for structuring data, and therefore XML-based encryption is the natural way to handle complex requirements for security in data interchange applications [5, 6]. XML Signature and XML Encryption standards are being used widely as building-block technologies [1, 7].

The XML Encryption specification must include a discussion of potential vulnerabilities and recommended practices when using the defined processing model in a larger application context. While it is impossible to predict all the ways an XML Encryption standard may be used, the discussion should alert users to ways in which potentially subtle weaknesses might be introduced. The XML Signature is a method of associating a key with referenced data (octets);

it does not normatively specify how keys are associated with persons or institutions, nor the meaning of the data being referenced and signed. [7]

We have known that SSL was launched in 1994 by Netscape. . Secure Socket Layer is the world's most widely used protocol for securing communication on the Internet.

Primary goal of SSL providing secure communications between web browsers and web servers. SSL is increasing in importance for Internet security. SSL works between the application and transport layer. SSL

relies on an underlying reliable protocol to assure that bytes are not lost or inserted. SSL is designed to make use of TCP to provide a reliable end-to-end secure service. Netscape originated SSL. The advantage of using SSL is that it makes use of the reliability and flow control mechanisms of TCP. SSL has two main objectives: To ensure confidentiality, by encrypting the data that moves between the communicating parties (client and the server). And also provide authentication of the session partners, through RSA algorithm [18].

Presently, Transport Layer Security (TLS) is used to secure communication over the Internet. TLS is an end-to-end security protocol because the famous SSL [18, 12].

TLS provide the link of communication between customer and merchant. TLS provide security and data integrity at the transport layer between two web applications. In a Customer and Merchant, TLS ensure that no third party may temper or eavesdrop of the information.

SSL/TLS provides confidentiality using symmetric encryption and message integrity using a message authentication code.

SSL/TLS includes protocol mechanisms to enable two TCP users to deter-mine the security mechanisms and services they will use.

2. RELATED WORK

An essential requirement of new Internet-wide security standards is that they apply to content created using extensible markup language (XML) [1, 2].

XML is also at the basis of interoperability protocols used to integrate applications across the Internet, such as Web services protocols: the Web service technology relies on different XML-based languages such as Simple Object Access Protocol (SOAP), Web Service Definition Language (WSDL), and Universal Description Discovery and Integration (UDDI) [4].

In this scenario, it is necessary to provide integrity, confidentiality and other security benefits to XML documents or portions of them, in a way that does not prevent further processing by standard XML tools [10].

In a this research set out to ascertain the suitability and applicability of using XSLT to achieve XML document security, it is not the only way in which XML document security can be achieved. A simple up till now effective way, adhering to the basis of element wise encryption and utilizing an XML processor when can be constructed in almost any programming language [11, 17].

The understandable benefit of such an approach is a reduction in the amount of complexity required to construct and implement. Problems with this approach are that such a package is required at each point in the system (where the security of the XML document is dealt with) and a new mechanism for determining selection for encryption within a source XML document is required.

For example of an implementation of this type is that of W3C's, XML Encryption, which incorporates security concerns into an XML processor, controlled by attributes in an XML document [12]. A realization of this processor is the XML Security Suite from IBM [9].

3. PROBLEM STATEMENT

XML Encryption is not proposed to swap SL/TLS. XML provides a mechanism for security requirements that are not covered by SSL/TLS. The following are two important areas not addressed by SSL:

- Encrypting part of the data being exchanged.
- Secure sessions between more than two parties.

By way of XML Encryption, each party can keep up secure or insecure states with any of the communicating parties. Together non secure and secure data can be exchanged in the identical document.

4. PURPOSED WORK

There are two procedures to XML Encryption and XML signature.

4.1 XML ENCRYPTION

XML encryption can be used to encrypt random data. The main advantage of the XML encryption is that it supports the encryption of specific portions of an XML document rather than the complete document. The most motivating part about XML encryption is that we tin can encrypt an entire document or its selected portions of a document. This is very hard to achieve in the non-XML planet. We can encrypt one or all of the following portions of an XML Document:

- The entire XML document.
- An element and all its sub elements of an XML document.
- The content portion of an XNL document.
- A reference to a resource outside of an XML document.

The steps involved in XML encryption are quite simple and are as follows:

1. choose the XML to be encrypted (one of the items listed earlier, i.e. All or part of an XML document)
2. The data is Covert to be encrypted in a canonical form (optional part).
3. Encrypt the result using public key encryption.
4. We send the encrypted XML document to the recipient.
5. All information needed to decrypt a document is contained within the document.
6. The Session could be secured on the document level and shared between multiple parties of transaction.
7. Sensitive data is easily interchanged between applications.

A Code, Show a simple XML Document, containing the details of a credit card of a user.

```
<?xml version="1.0" encoding="utf-8"?>
<Payment Info xmlns='http://mybank.org'>
  <C-Name> Deepak Singh </C-Name>
< Credit card Limit='100000, Currency
Type='EUR'>
   <Number> 1617 1718 0181 4444
</Number>
     <ExpiryDate>
       <date month="09"year="2009" />
          </ExpiryDate>
           <Issuer>  Master </Issuer>
           <cvc>123</cvc>
<cardAddress>
```



```
<address>
<firstName>Deepak</firstName>
<lastName>Singh</lastName>
<address1>H. No-03</address1>
<address2>Saboli bagh </address2>
<postalCode>110093</postalCode>
<CountryCode>IN</countryCode>
<TelephoneNumber>
+91-11-203-29530
</Telephone Number>
</address>
</cardAddress>
< /Credit card>
</Payment Info>
```

We will be not explain the versions details of this type XML document and likely simply remark that it contains the *credit card* details, such as the user's name, credit card limit, currency type, credit card number, issuer authority name and expiry date details. Let us assume that we want to encrypt this. When we perform XML encryption, a standard tag called *EncryptedData* comes to picture. As we have mentioned before. We can choose to encrypt selected portion of the XML document or we can encrypt it as a whole. For design purposes, we will see what happens when we only the actual credit card details were encrypt (such as its number, issuer name, cvc, expiry details). We can use the *CipherData* tag to encrypted text.

```
<? xml version="1.0" encoding="utf-8"?>
<Payment Info xmlns='http://mybank.org'>
<C-Name> Deepak Singh </C-Name>
< Credit card Limit='100000, Currency Type='EUR'>
<EncryptedData Tpye = http://www.w3.org/2001/04/
xmlenc# Content' xmlns= http://www.w3.org/2001
/04/xmlenc#'>
<CipherData>
< Cipher value> D7T60UB6 < /Cipher value>
</CipherData>
</EncryptedData>
< /Credit card>
</Payment Info>
```

As we can see, the credit card details are now encrypted therefore cannot be read/ changes. The truth that we have encrypted the details of the XML document is signified by using the *xmlenc# Content* value. If we had encrypted the full *Credit card* element, this would have been change to *xmlenc# Element*.

4.2 XML SIGNATURE

An XML signature is a digital signature obtained by applying a digital signature operation to arbitrary data. However, while the existing technologies allow us to sign only a whole XML document, XML signature provides a means to sign a portion of a document. This functionality is very important in a distributed multi party environment, where the necessity to sign only a portion of a document arises whenever changes and additions to the document are required. For instance, only order information can be seen by merchant and payment informant can be seen by payment gateway. This important feature is supported by XML signature. The extensible nature of XML also allows support for multiple signatures inside the same document. It is also important to highlight that the possibility of signing online a portion of a document and inserting the signature inside the document avoids the development of ad hoc methods to manage persistent signatures, and provides a flexible mechanism to sign and preserve part of the document.

The XML digital signature specification defines a number of XML elements, which describe the characteristics of an XML signature.

- Insure that a message has not been altered or tampered with. (Integrity).
- Protection against attacks that modify a message but maintain integrity. (Message authentication).
- Provide a means for message audit so that messages may not be repudiated. (Signer authenticity).

The steps in performing XML digital signature are as follows.

1. Create a SignedInfo element with SignatureMethod, nonicalizationMethod and Reverences.
2. Canonicalize the XML document.
3. Calculate the signature value, depending on the algorithms specified in the SignedInfo element.
4. Create the digital signature (i.e. signature element), which also includes the SignedInfo, KeyInfo and Signature Value element.

A simplistic example of a XML digital signature is show in bellow. We are also explaining important aspects of the signature.

```
<Signature>
<SignedInfo>
<SignatureMethod Algorithm="xmldsig#-
sha1"/>
</SignedInfo>
<Signature Value>
SSKSHBVSSMLATLKGHDWAGH
</Signature Value>
</Signature>
```

Let us discuss the contents of the digital signature in brief.

- *<Signature>……<Signature>*-This lock identifies the start and end of the XML digital signature.
- *<SignedInfo>……<SignedInfo>*- In This block specifies the algorithm used: first of all calculating the message digest (Using SHA-1) and then for preparing the XML digital signature (using RSA).
- *<SignatureValue>…….<Signature Value>*- In This block contains the actual XML digital signature for security purpose.

4.3 XML KEY MANAGEMENT SPECIFICATIONS (XKMS)

XML signature and XML encryption specifications provide mechanisms to sign and encrypt XML documents in critical e-services scenario and they involve the use of cryptographic keys. The need to integrate public key infrastructure (PKI) [14, 15] and digital certificates with XML-based applications arises and a W3C working group has been developing an open specifications named XML key management specifications (XKMS) [13]. Used together with XML

Signature and XML Encryption by XKMS specified a protocol for distributing and registering public keys,. The main goal of XKMS is to allow the development of XML-based trust services managing PKI-based cryptographic keys. XKMS is also aimed at reducing the complexity of PKI technology by simplifying the addition of security mechanisms in applications and relying on a trusted third party for all the activities related to PKI tasks.

5. CONCLUSION

In this paper we introduced the most important XML security technologies. We described two important initiatives, namely XML signature and XML encryption. We are facing the problem of protecting payment information distributed on the Internet. We then briefly reviewed the XML key management specifications, which provides facilities for the management of public keys used together with XML signature and XML encryption.

So that, by using XML technology, payment information is more secure as comparison to SSL.

REFERENCES

1. Apache XML Project http://xml.apache.org/.
2. N.Bradley (2002).The XMLCompanion. AddisonWesley, 3rd Ed.
3. E. Newcomer (2002). Understanding Web Services: XML, WSDL, SOAP, and UDDI. Addison Wesley.
4. P. Samarati, S. DeCapitanidi Vimercati (2001). Access control: Policies, models, and mechanisms. In Focardi R, Gorrieri R, editors, Foundations of Security Analysis and Design, LNCS 2171. SpringerVerlag.
5. XML Encryption Syntax and Processing, W3C-Recommendation-(2002). http://www.w3.org/TR/xmlenc-core/.
6. XML-Signature Syntax and Processing, W3C Recommendation (2002). http://www.w3.org/TR/xmldsigcore/.
7. W3C, "Extensible Markup Language (XML) 1.0 (Second Edition); W3C Recommendation 6-October-2000", http://www.w3.org/TR/2000/
8. REC-xml-20001006, October 2000.
9. W3C, "XSL Transformations (XSLT) Version 1.1; W3C Working Draft 24 August-2001", http://www.w3.org/TR/2001/WD-xslt-20010824/, August 2001.
10. Tidwell D., "The XML Security Suite: Increasing the security of e-business", http://www 4.ibm.com/ software/developer/library/xmlsecuritysuite/index.html, April 2000.
11. Nehren D., "XML through the Wall", http://www.xmlmag.com/upload/free/features/xml/ 2001/05may01/dn0102/dn0102.asp, May 2001.
12. Maruyama H. and Imamura T., "Element-Wise-XML-Encryption", http://lists.w3.org/Archives/Public/xml-encryption/2000Apr/att-0005/01-xmlenc.html, April 2000.
13. W3C, "XML Encryption Syntax and Processing; WG Working Draft 26 June 2001",http://www.w3.org TR/2001/WD-xmlenc-core 20010626/, June 2001.
14. W.Fordetal(2001).XML-Key management Specification (XKMS), W3CNote.http://www.w3.org/TR/xkms/
15. A. Arsenault, S. Turner (2002). Internet X.509 Public Key Infrastructure: Roadmap. Internet Draft, Internet Engineering Task Force.
16. A. Essiari, S. Mudumbai, M.R. Thompson (2003). Certificate-Based Authorization Policy in a PKI Environment. ACM Transactions on Information and System Security, 6(4):566–588.
17. Bakale Dournaee. XML Security by Mc Graw-Hill companies-2002.
18. R. G. Bartlett and M. W. Cook, "XML Security Using XSLT", 36th Hawaii International Conference on System Sciences– 2003.
19. William Stallings "Cryptography and Network Security", Fourth Edition.
20. Ajeet Singh "*Data Exchange and Mapping using XML"*, **LAP Lambert academic-Publication,Germany, www.lap.publishing .com,** ISBN-978-3-8484-8318-1 in 2012.

AUTHOR BIOGRAPHIES

AJEET SINGH: Born in 1984 in a town (Shikarpur) in District Bulandshahr of U.P. state in North India, Ajeet

Singh completed his schooling from a district town in U.P. Then, he obtained his **B.Sc** in **Computer Science** from Dr. Bhim Rao Ambedkar University, Agra, India in 2004. He obtained **Master of Computer Application** from Institute of Engineering &Technology (Govt. Engg. College), Lucknow, India in 2007. He then did his **Master of Technology** in Computer Science Engineering, specializing in Information Security, from Motilal Nehru National Institute of Technology, Allahabad, India (Deemed to be University of Central Govt. of India) in 2009. He started teaching soon at Skyline Institute of Engineering & Technology, Greater Noida. He was also a Counselor at the Indra Gandhi National Open University, MNNIT Center, Allahabad, India.

At present, he is a Faculty in the Dept. of Computer Science, School of Computing in **Debre Berhan University, Ethiopia**. Ajeet Singh has published eleven articles in reputed international conferences/journals and also one Book "*Data Exchange and Mapping using XML"*, LAP Lambert academic-Publication, Germany**.** Besides this, he has also contributed articles to different national & international conferences. He has the distinction of publishing an article in IEEE, a prestigious international engineering journal/conference. He is a member of the IEEE since 2008. He is guiding Undergraduate (B.Tech) and Postgraduate (MCA, M.Tech) students in the area of Computer Network Wireless Networks, Computer Security. His area of research: Computer

Network Wireless Networks, Computer Security, Mobile Computing, and Cryptography.

Dr. KARAN SINGH:
[B.Tech. (CSE), from KINT Sultanpur, India. M.Tech.(CSE), Ph.D.(CSE) from MNNIT, Allahabad, India]
Presently, I am working as Assistant Professor in School of Information and Communication Technology, Gautama Buddha University, Greater Noida, UP, India. To contribute strong teaching as well as laboratory skills and experience in Computer Network and Information Security with research capacity

SHAHZAD:
Presently working in Dept. of Computer Science, Samara University, ETHIOPIA. He did his Master in Computer Science and also M.Tech in Computer Science Engineering from MNNIT, Allahabad, India.

AZATH M.: Presently working as a Lecturer in Dept. of Computer Science, Debre Berhan University, Debre, Ethiopia. Gold Medalist in Master of Software Engineering from Anna University, Tamilnadu, India. He have Published two articles in Springer Journal.

SATISH KUMAR KONGA: Presently working at Dept. of Computer Science, Debre Berhan University, ETHIOPIA. He is Pursuing PhD from Dravidian University, Andhra Pradesh, INDIA. Masters in Computer Science from Kakatiya University, AP, India. Since 4 years he has been guiding many Master students (India), published few articles in reputed journals.

