Securing Web Services Using XML Signature & XML Encryption

Abhijeet G Gaikwad[1], Bhavin P. Dhalwani[2]

Institute of Management & Computer Studies

University of Mumbai, Kalina, India

Abstract: This paper is aimed to evaluate the importance of XML Signature and XMLEncryption for WS-Security. In today's e-business scenario, organizations are investing ahuge amount of their resources in Web Services. Web Service Transactions are donemainly through plain-text XML formats like SOAP and WSDL, hence hacking them is nota tedious task. XML Signature and XML Encryption ensure security to XML documentsas well as retain the structure of the documents, thereby making it easy to implementthem. These two methods are evaluated on the parameters of authentication, access control, availability,integration, confidentiality and non-repudiation.

Keywords: Web Services, Security, XML Encryption, XML Signature

I. Introduction

Web Services are client and server applications that communicate over the World Wide Web's (WWW) Hypertext Transfer Protocol (HTTP). In other words it is a software system designed to support interoperable machine-to-machine interaction over a network. In 1990's web service came into the picture and currently all online transactions rely on web services. By using web services, your application can publish its function or message to the rest of the world. Web services use XML to code and to decode data, and SOAP to transport it.

Due to its wide usage, it is important to keep web service secure. For a successful online transaction, it is necessary that all applications and communications must be secure and reliable.

In this paper, we would give an overview of web service security and then its architecture. Then XML signature and its types with syntax along with an overview of XML encryption and its syntax

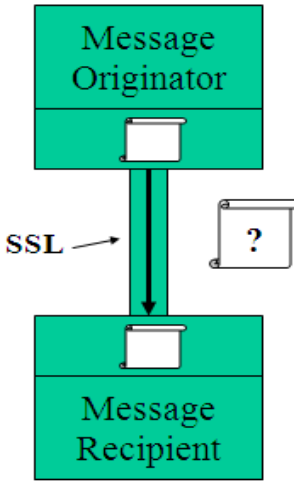


Fig. Web Service Security Basic Picture

ISSN: 2278 – 1323 All Rights Reserved © 2015 IJARCET

II. Web Service Security

Similar to other web applications web service also relies on HTTP protocol and therefore face same problem (i.e. threats and vulnerabilities)

To overcome such defects there are various network security principles are available. These includes,

- **Confidentiality:** It specifies that only the sender and the receiver should be able to access the content of the message. Confidentiality gets compromised if an unauthorized person is able to access the message.

  E.g. user A sends a message to user B. Another user C gets access to this message, which is not desired and hence confidentiality is lost in this case.

- **Authentication:** it helps to establish proof of identities. It ensures the origin of an electronic message or document is correctly identified.

  E.g. user C sends an electronic document over the internet to user B. however user C posed as user A while sending this document to user B.

- **Integrity:** When the contents of the message are changed after the sender sends it, but before it reaches the receiver, we say that integrity of message is lost.

  E.g. here user C tampers the original message send by user A which is actually destined to user B and sends the changed message to user B. user B has no way of knowing that the contents of the message were changed after user A had sent it.

- **Non-repudiation:** There are situations where a user sends a message and later on refuses that he had sent that message.

  E.g. user A send a funds transfer request to bank B over the internet. After the bank performs the funds transfer as per A's instructions, A could claim that he never sent that request.

- **Access Control:** It determines who should be able to access what. An access control maintains a Access Control List (ACL) which specifies the activity given to the user.

  E.g. we should be able to specify that user A can view the records in a database, but cannot update them. However user B might be allowed to make the update as well.

- **Availability:** It states that resources should be available to authorized parties at all times.

  E.g. due to an unauthorized user C, an authorized user A may not be able to contact with the authorized user B.

A. Web Service Security Architecture:

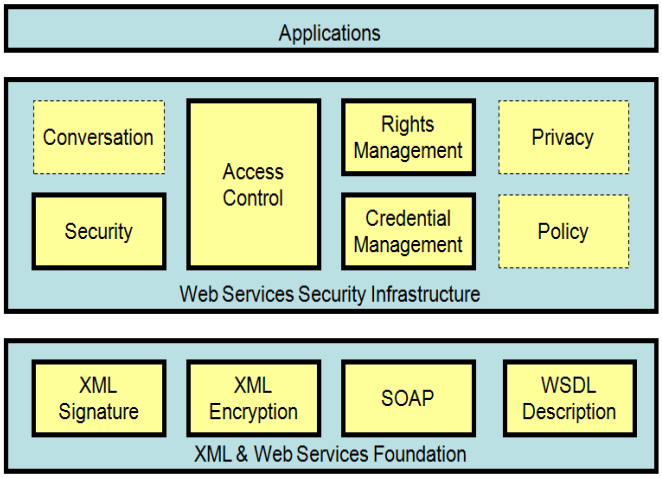


Fig. 1 Architecture of Web Service Security

The WS Security Architecture consists of different layers each as follows,

- **Conversation:** It follows Confidentiality policy. It Describe how to manage and authenticate message exchanges between parties.

- **Security:** Describes how to attach signatures and encryption headers to SOAP messages

- **Privacy :** Describes a model for how Web Services and requesters state privacy preferences and organizational privacy practice statements

- **Policy:** Describes the capabilities and constraints of the security policies on intermediaries and endpoints

ISSN: 2278 – 1323 All Rights Reserved © 2015 IJARCET

There are two main standards for XML Security

- ✓ **XML Signature**
- ✓ **XML Encryption**

III. XML Signature

XML signatures are digital signatures designed for use in XML transactions. An XML signature can sign more than one type of resource. For example, a single XML signature might cover character-encoded data (HTML), binary-encoded data (a JPG), XML-encoded data, and a specific section of an XML file.

A. How to Create an XML Signature:

- Determine which resources are to be signed.
- Calculate the digest of each resource.
- Collect the Reference elements
- Signing
- Add key information
- Enclose in a Signature element

The basic structure is as follows:

```
<Signature ID?>
        <SignedInfo>
        <CanonicalizationMethod/>
        <SignatureMethod/>
     (<Reference URI? >
            (<Transforms>)?
<DigestMethod>
                  <DigestValue>
        </Reference>)+
        </SignedInfo>
        <SignatureValue/>
        (<KeyInfo/>)?
        (<Object ID?/>)*
</Signature>
```

B. XML Signature Example:

```
<Signature Id="MyFirstSignature"
xmlns=http://www.w3.org/2000/09/xmldsig#>
<SignedInfo>
<CanonicalizationMethod Algorithm="…"/>
<SignatureMethod Algorithm="…"/>
<Reference URI="…">
<Transforms>
<Transform Algorithm="…"/>
</Transforms>
<DigestMethod
Algorithm="…"/><DigestValue>j6lwx3rvEPO0vKtMup4Nb
eVu8nk=
</DigestValue>
</Reference>
</SignedInfo>
<SignatureValue>MC0CFFrVLtRlk=...</SignatureValue>
<KeyInfo>
<KeyValue>
<DSAKeyValue></DSAKeyValue>
</KeyValue>
</KeyInfo>
</Signature>
```

C. Explanation of the Example

The above example shows the working of XML Signature. **"MyFirstSignature "**is the signature id of the document. The **"xmlns"**specifies the URI name on which we apply XML Signature algorithm. The **"CanonicalizationMethod"** is used to create canonical XML.

"SignatureMethod" is used to hash and sign the contents. The **"Reference URI"** Contains the digest method and the digest value. It can occur multiple times. URI attribute points to the digested resource. The **"SignatureValue"** uses base64 encoded value of the signature.

"KeyInfo" is optional which includes the key that can be used to validate the signature.

D. Verifying an XML Signature:

- Verify the signature of the <SignedInfo> element. To do so, recalculate the digest of the <SignedInfo> element (using the digest algorithm specified in the <SignatureMethod> element) and use the public verification key to verify that the value of the <SignatureValue> element is correct for the digest of the <SignedInfo> element.

- If this step passes, recalculate the digests of the references contained within the <SignedInfo> element and compare them to the digest values expressed in each

ISSN: 2278 – 1323 All Rights Reserved © 2015 IJARCET

<Reference> element's corresponding <DigestValue>element.

IV. XML Encryption

XML Encryption is an encryption technology that is optimized for XML data. Its practical benefits include partial encryption, which encrypts specific tags contained in XML data multiple encryption, which encrypts data multiple times, and a complex encryption, such as the designation of recipients who were permitted to decrypt respective portions of data. The use of XML Encryption also helps solve security problems, including XML data eavesdropping.

XML Encryption was established by the W3C as formal version of W3C recommendations in Dec.2002. The W3C also established related specifications that solve problems raised when XML Encryption and XML Signature are used in combination.

A. Encryption Syntax Structure:

```
<EncryptedData Id? Type? MimeType? Encoding?>
    <EncryptionMethod/>?
    <ds:KeyInfo>
        <EncryptedKey>?
        <AgreementMethod>?
        <ds:KeyName>?
        <ds:RetrievalMethod>?
        <ds:*>?
    </ds:KeyInfo>?
    <CipherData>
        <CipherValue>?
        <CipherReference URI?>?
    </CipherData>
    <EncryptionProperties>?
</EncryptedData>
```

Fig. 3 XML Encryption Syntax

B. Example of XML Encryption:

- **XML File**

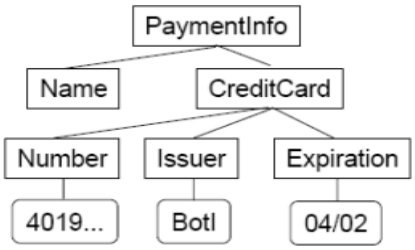
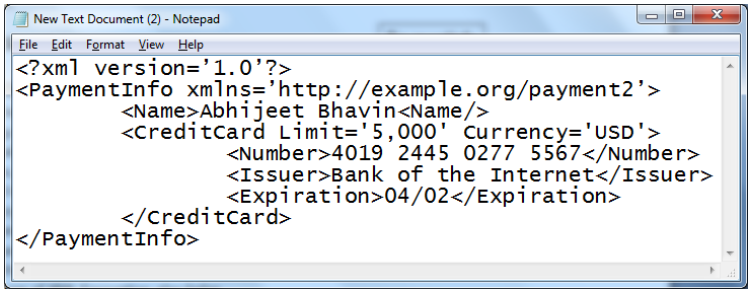


Fig. 4 XML File Example



- **Encryption**

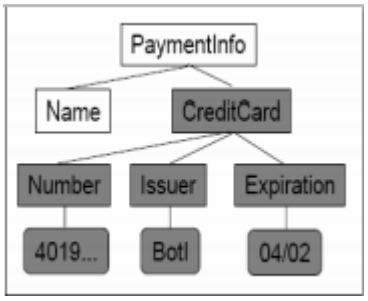
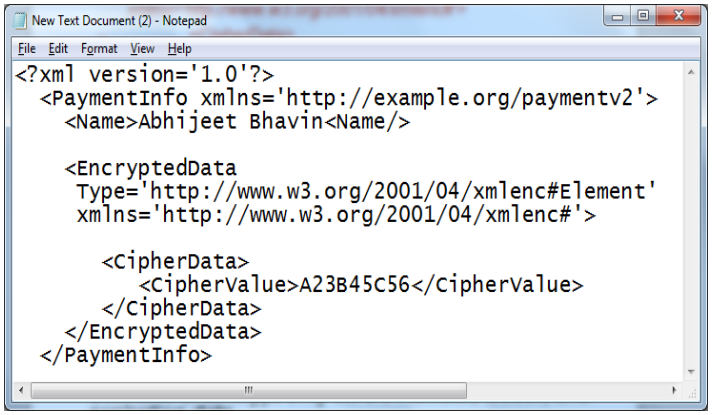


Fig. 5 Encryption of XML File



ISSN: 2278 – 1323 All Rights Reserved © 2015 IJARCET

C.Explanation of the Example

In the above example we encrypt CreditCard details of the PaymentInfo using XML Encryption standards. PaymentInfo is the XML file which has Name and CreditCardas attributes. Each attribute has a value.

"EncryptedData" contains data that is being encrypted. **"CipherData"** contains encrypted data that is being decrypted at the other end.

V. Conclusion

Without Security and Trust, "Web Services are dead On Arrival."

As XML becomes a vital component of the emerging electronic business infrastructure, we need trustable, secure XML messages to form the basis of business transactions. One key to enabling secure transactions is the concept of a digital signature, ensuring the integrity and authenticity of origin for business documents. XML Signature is an evolving standard for digital signatures that both addresses the special issues and requirements that XML presents for signing operations and uses XML syntax for capturing the result, simplifying its integration into XML applications.

Acknowledgement

The Research Paper is creative work of many minds. A proper synchronization between individual is must for any project or paper to be completed successfully.

Through words are not enough to express our gratitude to all those who contributed in making this Research Paper. We thank our teachers from bottom of our heart for providing their valuable suggestions as an when we approached them and kept us informing time to time about what we were supposed to do.

We thank our guide for the exclusive guidance on the topic we have work with. There were times when things seem to be vast to be done and we were running short of energy and time but, made us endures such times with the words of encouragement. We would also thank our principal and other staff members of Information Technology Department for their constant guidance and support.

We are grateful to our parents and friends for their constant inspiration and encouragement.

Finally a few thankful words to all those who have stood along with us throughout the process and we believe that it is their support and presence that make us to work in this paper.

References

A. Links:

[1] http://en.wikipedia.org/wiki/XML

[2] http://users.informatik.haw-hamburg.de/~schmidt/it/presentations/XMLEncrypt-limiao-06.pdf

[3] http://www.xml.com/pub/a/2001/08/08/xmldsig.html

B. Books:

[1] Cryptography and Network Security by AtulKahate, pp. 7 - 10

Abhijeet G. Gaikwadis currently studying in Mumbai University at IMCOST, Thane since 2012, currently pursuing MCA with excellent academics. He is having interest in Studying New Technologies.

Bhavin P. Dhalwani is currently studying in Mumbai University at IMCOST, Thane since 2012, currently pursuing MCA with excellent academics. He is having interest in Studying New Technologies.

ISSN: 2278 – 1323 All Rights Reserved © 2015 IJARCET