Digital Signature based secure XML emails

Srushti Arekar¹, Prajakta Jagtap², Priyanka Jagtap³, Sharanya Shivakumar⁴
Savitribai Phule Pune University, Bharati Vidyapeeth College of Engineering for Women, Dhankawadi, Pune-411043, India.

Abstract
XML has become one of the important standards for data exchange on web and uniform data model for data integration. For XML data, digital signature is applied to it, which is nothing but XML Digital Signature. XML Digital Signature can be said as a standard for signing the XML data or document and also to represent XML Dig-Sig in an XML format. The use of XML format in the email system ensures security and privacy in its transaction [3]. A higher level of security can be provided to the Digital Signature of our XML data by using XML Encryption [3]. It is a flexible methodology for representing encrypted data in XML format. XML encryption supports the signing of the whole document or the partial document [3].

Keywords: XML encryption, XML security, XML signature, Web services, XML decryption, XML key management.

1. INTRODUCTION
Security has always been an important aspect in the business world as it ensures the integrity of content and transactions to maintain its confidentiality [4]. Development of electronic email system has played an important role in technical and sociological developments. PGP and S/MIME mechanisms provide functionalities like non-repudiation, data integrity, data confidentiality, which provides security to message body only where the headers remain unauthenticated [7]. The solution for this can be, using XML format which provides the specific structure for storing data. Also, if changes occur in format of data, it does not affect the semantics of document. The XML specification uses its own rules and syntaxes to define tags. These tags are further used to structure the information in a format.

In recent years numbers of ways have been developed to save email in XML format. In this approach save the header of the email is saved in one tag that is from is saved in <From> and the rest of the body is saved in the tag body i.e. <body>. The email is constructed in a specified XML format which reproduce the email structure strictly, along with the structure of the signature and the encryption. XML format though is more efficient in searching, processing, and archiving, but these approaches do not provide any more security properties than the normal mail.

2. GOAL OF XML EMAILS
We design a framework in XML format which is used to exchange email, called XML email that will take the advantage of both XML and email. The framework also uses XML digital signature for authenticity, which will provide end to end security and transported via the existing system.

3. DIGITAL SIGNATURE
Digital Signature is a mathematical technique used to check authenticity and the integrity of a message, software or any digital document.

Example: Bob has been given two keys called public and private key. Bob’s public key is available to everyone but he keeps his private key to himself. When Susan wants to send a message to Bob, she will encrypt the message using Bob’s public key. Bob uses his private key to decrypt the message. Any of Bob’s co-workers might have access to Susan’s message but without Bob’s private key the data is worthless.

Problems:
“Only with trusted applications”
One of the main differences between digital signature and written signature is that the user does not know what he signs. The user application presents the hash code to be signed by digital signing algorithm. An attacker who gains control of user’s PC can possibly replace the user applications with foreign substitutes. To protect this authentication system must be setup between users’ application.

4. XML SIGNATURE
XML Signature provides facility that to verify the message was originated from the intended sender. It checks message integrity by defining whether a message was altered during transmission. It requires canonicalization before hashing and signing.

Example: Alice wants to send a message to Bob. Alice’s message will get converted into XML and then it gets encrypted with Bob’s public key. At receiver’s side Bob will decrypt message using Bob’s private key. If the decrypted message is in readable format, Bob will be assured that the message’s repudiation is maintained as the mail has been received from Alice. If Eve tampers the message Bob will realize that the message is tampered as the hash value calculated later won’t match. Hence this will provide message integrity.

Problems:
• For legal or business reasons we need to be able to verify who sent a particular message. Otherwise, we may not be sure of its origin and the sender may deny having sent it (repudiation). We assume the sender has signed the message to prove she is its author.
• Messages may be altered during transmission, so we need to verify that the data is in its original form when it reaches its destination.
• Producing a signed message should not require a large computational power or take a long time.
• We need to express a digital signature in a standardised XML format, so interoperability can be ensured between applications.

5. XML DIGITAL SIGNATURE
XML Digital Signature supports
• XML Signature.
• Creation of XML Digital Signature.
• Verification of XML Digital Signature.
• Signing via Digital Certificates or password.
• Multiple transformation option.
• Signing of XML, XML schema, XBRL, WSDL and other files.

6. PROPOSED SOLUTIONS
6.1 MATHEMATICAL MODEL: SET THEORY
Let U is a universal set where U= {Us, S, R, Es, Ec, PuK, PrK, Db, Sr}
Where………..
Us is a set of user who will be using XML cloud service for secure transmission and reception of emails.
Us={u1, u2, u3, u4…un} ⊆ U
S is a set of sender who will send XML mail to the intended recipient
S={s1,s2,s3…sn}
R is the set of receivers who will receive the mail from the sender.
R={r1,r2,r3…rn}
Es is the singleton set of email server
Ec is the set of email client such that
Ec={ec1, ec2,…..ecn}
PuK is the public key of the recipient.
PrK is the private key of the recipient.
Db is the singleton set of the database that we use to store the public, private keys and content of our email. It is the copy of the server database.
Sr is singleton the server component of the system. The server is responsible for registering, authenticating and providing associations to the end user

6.2 FUNCTIONALITIES
Db' = RegisterUser(uid , password , fullname, address, country, contact, email);
password = SHA1(input_password);
sr1 = {AuthenticateUser(uid, password, Db'), AssignPublicKey(uid, password, fullname, address, country, contact, email), AssignPrivateKey(uid, password, fullname, address, country, contact, email) , FindRecipient(uid), ReturnSortedList(Sorted_list), UpdateDatabase(uid, password, fullname, address, country, contact, email)}
S1 = {Login(uid,password),Register(uid, password, fullname , address , country , contact , email), AddRecipients(uid_of_receiver),
FecthUser(uid_of_receiver),
AddRecipient(uid,PublicKey),Encrypt(privateKey)}
R1={ Login(uid,password), Register(uid, password , fullname ,address,country,contact, email),Decrypt(privateKey)}
Consider Us1, Us2 be the user of the system. Let PuK1, PrK1 are the keys associated with user Us1. Similarly, keys PuK2, PrK2 are associated with user Us2.
Consider a scenario where Us1 sending message M to user Us2.
Following are the prerequisites for sending message:
1. User Us1, Us2 must be authorized/registered user.
2. Each of the users knows public key of each other.
3. We are assuming that we are not sending any attachment.

7. STEPS
7.1 Following are the steps required for sending mail:
a) At the sender side, let H(x) be the hashing function that is used to calculate hash of the given message M. Let M' be the hash of given message.
M'= M * H(x) where x be any arbitrary integer value.
b) Find out modulus i.e.N1 such that N>0
Where N1=P1*Q1
P1, Q1 ε set of prime numbers.
Calculate cipher text of given message using
C=M*e(mod N1)
Where C=cipher text
e=encryption exponent
c) Convert the message M into standard XML format. Consider it as Cx.
Where Cx is the message in XML format.

7.2 Following are the steps for retrieving mail
a) Convert received mail Cx into C cipher text message.
b) As N1 is publically known we can re-calculate original message M using following function.
M=C^d (mod N1) Where d = decryption exponent
N1= P*Q where P, Q are set of prime numbers C=cipher text.
c) Recalculate hash of message M. Consider it as M2' using hash function H(x).
M2'=M*H(x)
d) Compare M2' with M’ if both matches then data is not tampered.

Solution with XML Digital Signature
The signature embedded within XML content that was signed is called “Enveloped Signature”. Following format represent Enveloped Signature:
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<EMAIL>
<item>
<data>
<mail_from></mail_from>
<algo_index></algo_index>
<mailTo></mailTo>
<stat></stat>
</Signature>
</SignedInfo>
8. CHARACTERISTICS

- This XML format along with XML digital signature is used to sign only the specific portion of document rather than the entire documents.
- Joint effort between World Wide Web Consortium (W3C) and Internet Engineering Task (IET).
- Can sign more than one type of resources:
  1) Character encoded data (HTML)
  2) Binary encoded data (JPG)
  3) XML encoded data

The use of XML Digital Signature involves two parts

1) XML DigSig creation
The signature is placed as an element of signed document. It consists of following parts:
   a) CanonicalizationMethod: Defines algorithm used to canonicalize the signedInfo element.
   b) SignatureMethod: Defines the signature algorithm used to generate signature.
   c) Reference: Identifies the data to be digested and also indicates the root. Any valid URI can be used as a reference and more than one reference can be used.
   d) Transforms: Defines the processing transformation used before digesting.
   e) DigestMethod: Defines digest method used.
   f) DigestValue: This will contain actual digest value of the reference.
   g) SignatureValue: ActualBase64 signature of signedInfo.
   h) KeyInfo: It is optional information about key needed to validate signature
   i) SignedInfo: This will contain information about signature.

2) XML DlgSig verification
   Step 1: Canonicalize the SignedInfo element.
   Step 2: Compute the digest of signedInfo element using the method described within it.
   Step 3: Compare the above value with value got from applying the signer’s public key to the value in the signatureValue element.
   Step 4: Compute digest of referenced items and compare those digest found within each Reference tag.

9. FLOW DIAGRAM OF THE SYSTEM

Figure 1: Flow Diagram

The flow of our system is as follows:
Firstly the system calculates hash of the message content using SHA-1. Encrypt message content and its hash using receiver’s public key which is generated by RSA algorithm. Convert encrypted data along with hash into XML format. Send the encrypted XML data to server. Server will store it into ‘email’ database.
At the receiver side, user will decrypt using user’s private key. User will get original data with hash. User will again calculate hash using SHA. Compare calculated hash with decrypted hash if verification is successful then user can be concluded that data is not at all transferred.
10. SEQUENTIAL DIAGRAM OF THE SYSTEM

The figure shows the sequential flow of our system which describes the steps of the working of our project.

10.1 Actors: User, Access Application, Server, Store in the database.

10.2 Precondition: User must be registered in the system.

10.3 Description: When user registers the system will allocate public and private key to user. User will login using user’s user id and password. After writing email user will fetch the list of recipient from server. Server will provide recipient’s sorted list along with their public key. System will encrypt mail using receiver’s public key and total encrypted mail along with its hash transmitted to server. Mail content along with its header will be stored at the server database. Recipient can fetch it anytime from server and decrypt it using its private key and verifies it using the reverse procedure.

11. SYSTEM SNAPSHOTS

Figure 2: Sequential Diagram

Figure 3: Dig-Sig Email system

Figure 4: Message in XML format

Figure 5: Successful email retrieval

Figure 6: Email retrieval after attack by hacker in the network
12. CONCLUSION
In this paper, the system proposed by us takes the advantages of the XML, web services and XML digital signature. We have made sure about a secure end to end authentication with the help of XML encryption. In MIME, message transmission protocol we have to embed outer headers into inner headers to secure it. This will increase unnecessary overhead to transmit the data. In our system, we are converting whole document including inner as well as outer header into XML format so it’s not necessary embed inner header with outer header. Various organizations provides predefined XML schema. This schema is in well-defined format and easy to understand therefore information retrieval is very easy. The main property of authenticity, integrity and non-repudiation of digital signature is embedded in XML hence it provides more security. Foreseeing many issues of spamming and hacking of current system, our system fights back all these problems to provide privacy to receiver and sender as well as security to email system.

ACKNOWLEDGMENT
We were guided by Prof. Mrs. Anjali Kadam, assistant professor of Computer Department in Bharati Vidyapeeth College of Engineering for Women. Our sincere thanks for her support and guidance. Her knowledge and support in the field of XML and web services was of great help in making our project paper.

REFERENCES
[6]. Digital Signature with Hashing and XML Signature Patterns.
[7]. Albert Levi, Can Berk Guder,”Understanding the limitations of S/MIME digital signatures for E-mails: A GUI based approach”, Faculty of Engineering and Natural Sciences, Sabanci University, Orhanli, Tuzla, 34956 Istanbul, Turkey.
[8]. Alok Gupta, Y. Alex Tung, James R. Marsden , “Digital signature: use and modification to achieve success in next generational e-business processes”, Department of Information and Decision Sciences, Carlson School of Management, University of Minnesota, Minneapolis, MN 55455, USA.

AUTHORS
SRUSHTI AREKAR has been studying in Bharati Vidyapeeth College of Engineering, Dhankawadi, Pune- 43, in Computer Science branch, 2015 Batch.
PRAJAKTA JAGTAP has been studying in Bharati Vidyapeeth College of Engineering, Dhankawadi, Pune- 43, in Computer Science branch, 2015 Batch.
PRIYANKA JAGTAP has been studying in Bharati Vidyapeeth College of Engineering, Dhankawadi, Pune- 43, in Computer Science branch, 2015 Batch.
SHARANYA SHIVAKUMAR has been studying in Bharati Vidyapeeth College of Engineering, Dhankawadi, Pune- 43, in Computer Science branch, 2015 Batch.