International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 10, October 2022)

Web Services Hybrid RSA-AES Encryption

Sanidayal Gupta[1], Prof. Vishal Paranjape[2]
Department of Computer Science & Engineering Global Nature Care Sangathan's Group of Institutions, India

***Abstract*-- Today's distributed systems make extensive use of XML and Web services. The security of XML-based communication and the Web services themselves is crucial for the overall security of these systems. To encourage interoperability, the safety precautions should preferably be based on tested standards. In this work, we present a tutorial on recent safety standards for XML and Web services. The technologies that are covered include eXtensible Access Control Markup Language (XACML), SAML, WS-Security, WS-Trust, WS-Secure Conversation, XML Signature, XML Encryption, and XML Key Management Specification (XKMS).**

The limits of traditional digital multi-signature systems are examined in this study. By utilising the inherent structure of the XML documents and the security provided by conventional digital signatures, we suggest a multi-signature strategy for RSA-based XML documents. In our system, an XML document is converted into a subdocument using the Xpath, and each participant signer then authenticates the subdocument that belongs to them. Therefore, increase the efficiency and adaptability of signs.

***Keywords*- XML, Multi-signature, RSA, Xpath**

I. INTRODUCTION

A software system created via a network to facilitate interoperable machine-to-machine communication is referred to as a WEB service[1]. In other words, web services come with a platform for system management that is independent of programming language and operating system. Current distributed systems make extensive use of web services, which are also the preferred technology for service-oriented architecture (SOA) implementation. In such systems, loosely linked services may be spread across organisational domains.

The extensive use of the Extensible Markup Language by web services as a markup language is a major factor in demonstrating its viability for the inclusion of diverse structures (XML). The XML-based Web Services Description Language, for instance, is used to describe a Web service interface (WSDL). In addition, XML-based SOAP messages are used to carry out communication. As a result, the security of a Web services-based system also depends on the confidentiality and integrity of the XML-based SOAP messages that are used for communication, in addition to the security of the services themselves.

Over the past few years, various specifications relating to security in Web services and XML have been standardised by the World Wide Web Consortium (W3C) and the Organization for the Advancement of Structured Information Standards (OASIS). An overview of various security requirements is given in this document. It is obvious that using these established standards for developing Web services, as opposed to constructing proprietary solutions, provides benefits for aiding system interoperability and reusability.

The standard format for data exchange on the Internet is now XML. XML is widely used in many industries because to its extensibility, including Web services, EDI (Electronic Data Interchange), e-government, and e-commerce, among others. Due to the growing use of XML, the issue of XML data security is receiving increased attention. The information's integrity, authenticity, and undeniability are all guaranteed by an XML digital signature, which also distinguishes the information's state. When processing an XML document's digital signature, it shows vast technical superiority over a traditional digital signature, making it more unique.

Many papers used in the process of electronic commerce or government require many signatures in addition to the individual signature of the signatory. Examples include contracts, agreements, and request forms. For instance, the management, the treasurer, and the teller must all sign when a corporation enters its accounts. Digital multi-signatures are an extension of the digital signature domain that asks many people to digitally sign the same piece of information [1].

Three steps make up an intuitive process for creating a traditional multi-signature for a document: First, a duplicate of the document is delivered to each participant signatory. Second, using their private key and a well-known digital signature algorithm like RSA's or ElGamal's, each participant signer creates their signature. Finally, a multi-signature for the document is created using the individual signatures. The size of the conventional multi-signature increases in proportion to the number of signers, and the time needed for multi-signature verification is the same as the time needed for multi-signature generation. As a result, it hinders the effectiveness of multi-signatures' generations and verifications.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 10, October 2022)

Wu et al. suggested a delegated multi-signature approach to get around the issue such that participant signers only sign on the subdocuments that they are in charge of[2]. Wu's method is inapplicable to XML documents since it ignores the logical structure of XML and cannot guarantee that the sub-documents deconstructed are meaningful.

Using the logical structure of an XML document, we suggested an XML multi-signature system in this study that is based on the RSA broadcast digital multi-signature. The participant signers only sign on the subdocuments that they are accountable for, in accordance with the rule of visiting node regarding XPath language of XML, building appropriate relationships of each sub-document and the various signers. We created the framework for an XML-based multi-signature system.

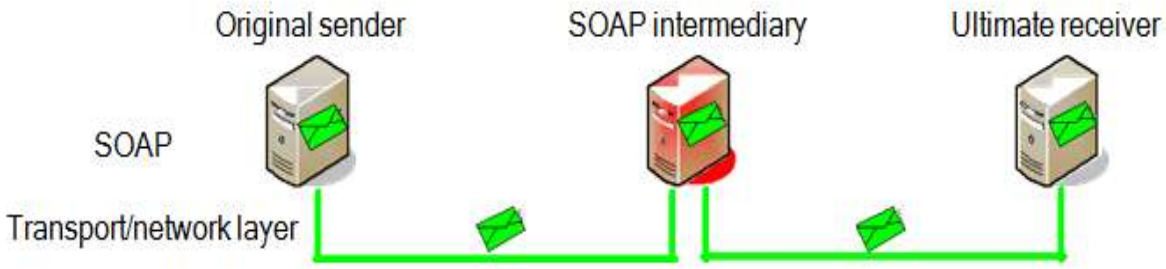


Fig. 1. The transport/network layer security (e.g., SSL/TLS or IPSec) is broken at the intermediary SOAP node. By applying security at the SOAP/XML level, on the other hand, end-to-end security can provided.

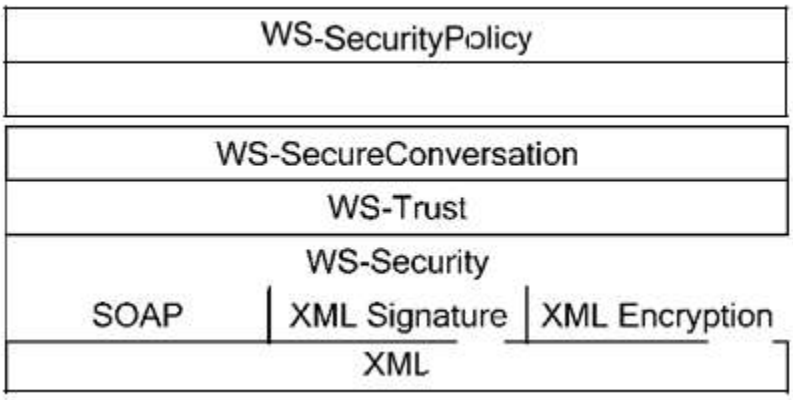


Figure 2

Fig. 2. The conceptual relationship between the XML and Web services security standards. Be aware that Web Services Policy can also be used (independently) for other purposes than security. Also recall that XKMS may provide key management for use with XML Encryption and XML Signature, although this is not shown in the figure.

Encryption. XKMS basically defines simple Web services interfaces for key management, thereby hiding the complex-ity of traditional public key infrastructures (PKIs) from the clients. XML Signature, XML Encryption, and XKMS are all discussed in more detail in Section III.

As noted previously, SOAP is an XML based messaging for-mat. Thus, XML Signature and XML Encryption are obvious candidates for being reused to provide SOAP security as well. As illustrated in Figure 2, this is exactly what WS-Security does. WS-Security specifies how to apply XML Signature and XML Encryption to SOAP messages, effectively providing integrity and confidentiality to SOAP messages (or parts of SOAP messages). As multiple encryptions can be used within the same SOAP message, the different parts of a SOAP message may be encrypted for different receivers (SOAP intermediaries).

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 10, October 2022)

Likewise, a SOAP intermediary may add an additional signature to a SOAP message, thereby providing integrity protection for a newly added header or supporting separation-of-duty through co-signatures.

In addition to providing confidentiality and integrity for SOAP messages, WS-Security also provides a mechanism to avoid replay attacks (i.e., timestamps) and a way to include se-curity tokens in SOAP messages. Security tokens are typically used to provide authentication and authorization.

WS-Security has no notion of a communication session, that is, it is only concerned with securing a single SOAP message or a single SOAP request/response exchange. In cases where multiple message exchanges are expected, WS-SecureConversation may be used to establish and maintain an authenticated context. The authenticated context is represented by a URI in a context token and consists of a shared secretthat can be used for key derivation. WS-SecureConversation relies on WS-Trust to establish the security context.

WS-Trust basically defines a framework for obtaining security tokens (including the context tokens used in WS-SecureConversation) and brokering of trust. WS-Security, WS-Trust, and WS-SecureConversation are all discussed in more detail in Section IV.

With a range of Web services standards, interoperability becomes very difficult unless the communicating parties know what standards to use and how these standards are to be used. Web Services Policy provides the means by which service providers and clients can specify their interoperability requirements and capabilities. WS-SecurityPolicy can be viewed as an extension to Web Services Policy, defining how Web Services Policy can be used to specify requirements and capabilities regarding the use of WS-Security, WS-Trust, and WS-SecureConversation. For instance, a service provider may specify using WS-Policy/WS-SecurityPolicy that it requires certain message parts to be encrypted. WS-Policy and WS-SecurityPolicy are also further discussed in Section IV.

The last two standards covered in this paper are the Security Assertion Markup Language (SAML) and the eXtensible Access Control Markup Language (XACML). SAML may be used to communicate authentication, attribute, and authorization information in a trusted way. SAML is based on XML and although its original motivation was single sign-on for Web browsing, it is also well suited for use in Web services. XACML on the other hand is used to define access control policies in XML, and may be used to define access control policies for any type of resource.

Because SAML and XACML are not targeted exclusively at Web services, SAML and XACML were not included in Figure 2. However, this does not imply that there is no interaction between these standards. A XACML implementation may for instance rely on the security tokens of WS-Security for authentication. As to security tokens, there is also a SAML based security token in WS-Security. SAML and XACML are both further discussed in Section V.

II. An Overview Of XML And WEB Services Security

XML based SOAP messages form the basis for exchanging information between entities in Web services systems. The information contained within these SOAP messages may be subject to both confidentiality and integrity requirements. Although mechanisms at lower layers may provide end-to-end security, these lower layer mechanisms are often insufficient. This is due to the fact that a SOAP message may be subject to processing and even modification (e.g., removal/insertion of a SOAP header) at intermediary nodes. The result being that the end-to-end security provided by lower layer mechanisms (e.g., SSL/TLS) is broken, as illustrated in Figure 1. Relying on lower layers for end-to-end security may also cause problems if a message is to pass through various networks utilizing different transport protocols. Furthermore, security at the XML level has the advantage of enabling confidentiality and source integrity to be maintained also during storage at the receiving node(s).

XML Signature and XML Encryption are used to provide integrity and confidentiality respectively. Although these two standards are based on digital signatures and encryption, none of them define any new cryptographic algorithms. Instead, XML Signature and XML Encryption define how to apply well established digital signature/encryption algorithms to XML. This includes:

A standardized way to represent signatures, encrypted data, and information about the associated key(s) in XML, independent of whether the signed/encrypted re-source is an XML resource or not.

The possibility to sign and/or encrypt selected parts of an XML document.

The means to transform two logically equivalent XML documents, but with syntactic differences, into the same physical representation. This is referred to as canonical-ization.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 10, October 2022)

In order to be able to verify the signature of an XML resource that has had its representation changed, but still has the same logical meaning, it is essential that canonicalization is performed as part of the XML signature creation and verification processes.

As both XML Signature and XML Encryption rely on the use of cryptographic keys, key management is a prerequisite for their effective use on a larger scale. Therefore, the XML Key Management Specification (XKMS) was created to be suitable for use in combination with XML Signature and XML

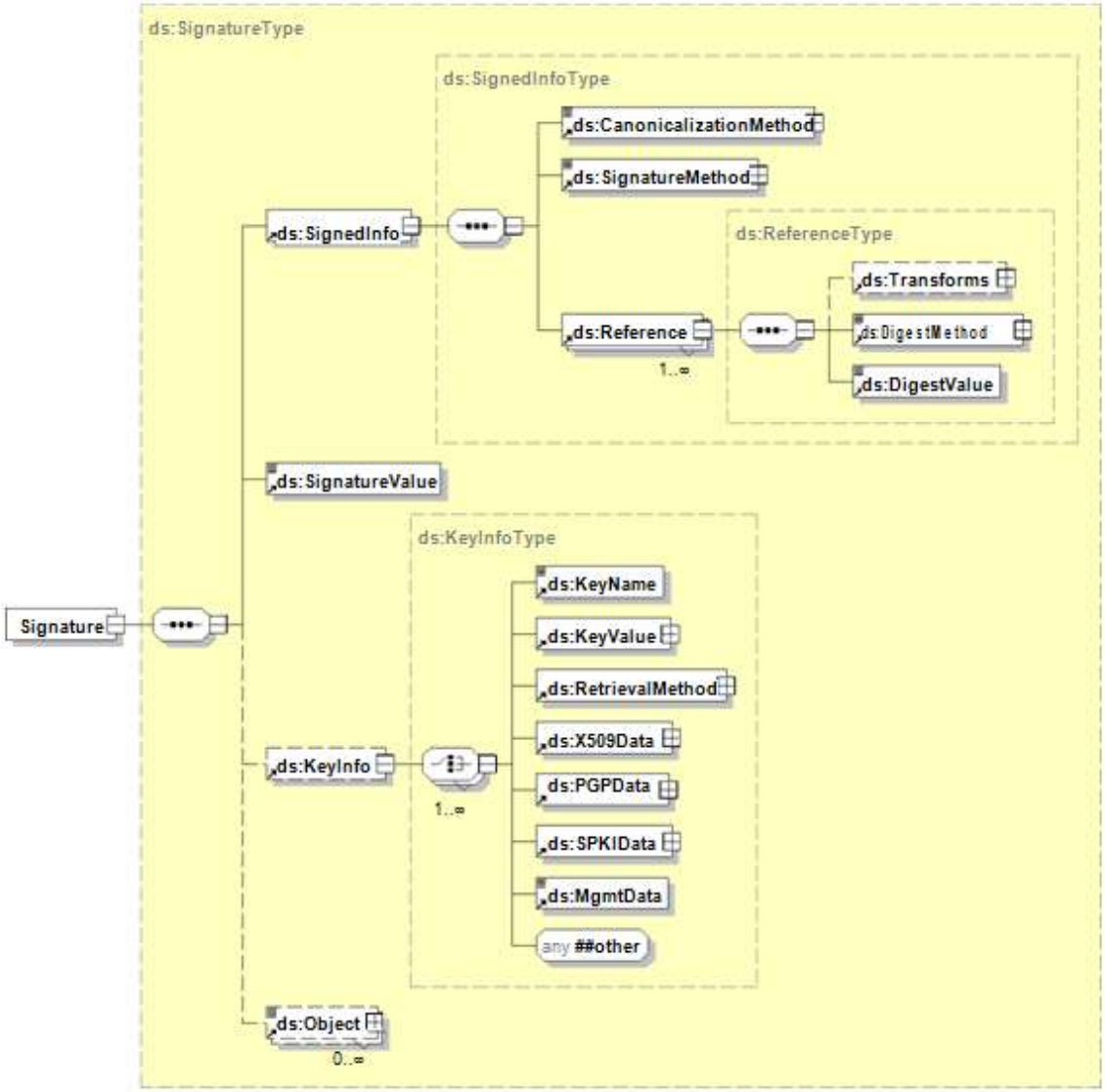


Fig. 3. The Signature element. (XML attributes are not shown.)

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 10, October 2022)

III. AN XML MULTI-SIGNATURE SCHEME

A frame of multi-signature scheme

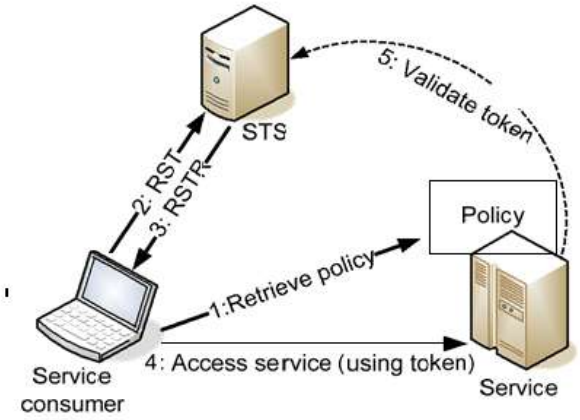
Generally speaking [1] , the digital multi-signature scheme's participant has the news sender (issuer), the news signer (signer), the signature verifier (verifier) and the signature gatherer (collector).

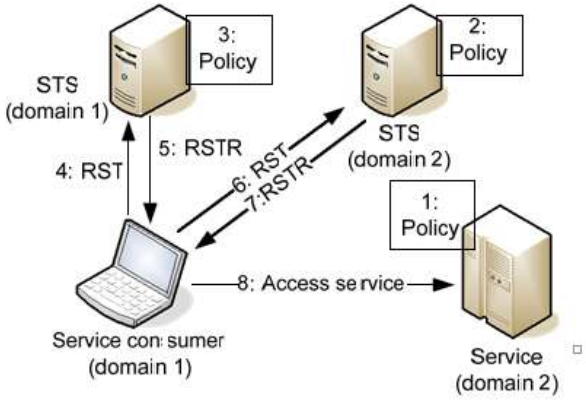
We designed a frame of the multi-signature scheme. There are four modules involved in this scheme: a single signature module (SS), a system authorization module(SA), the documents dispatcher module(DD), and signature collection module(SC). SA provides services such as the initialization of system parameters and issuing the certificate to each signer. DD is responsible for XML documents decomposition and subdocument delegation.

These subdocuments assign for every signer by a delegation algorithm. SS produces each signer's signature. SC collects and verifies personal signatures generated by each signer andconstructs a multi-signature for the group. Each module's relation is shown in Figure 1[3]. Let $G=\{U_1,U_2,U_3\ldots,U_n\}$ bethe registered group, and U_i is one signer of G . Let $M=\{M_1,M_2,M_3\ldots M_n\}$ be the document to be signed, and M_i be the sub-document of M delegated to U_i by DD.

The whole process for generating a multi- signature in our scheme

The procedure for generating a multi-signature consists of four stages: SA sends certificate to each signer stage, documents division stage, the multi-signature generation stage, and the multi-signature verification stage. Each stage is described as follows.





The policy of the service (which is expressed using Web Services Policy/WS-SecurityPolicy) specifies a security token required to access the service. After retrieving the policy (e.g., from the WSDL file of the service), the service consumer uses the information in the policy to obtain the correct security token from the security token service (STS). The communication with the STS consists of a request security token (RST) element/message and a request security token response (RSTR) element/message. When receiving the security token from the service consumer, the service may have the token validated by the STS or validate the token itself.

The policy of the service specifies that a security token from the STS in domain 2 is required. The policy of the STS in domain 2 requires a security token from the STS in domain 1 (i.e., one of its policy alternatives accepts such a security token). The STS in domain 1, for which the service consumer has a valid username/password, requires a username token for authentication. After retrieving the policies in the given order (1, 2, and 3), the service consumer obtains a security token, from the STS in domain 1 (4-5), which is then used to obtain a security token from the STS in domain 2 (6-7). This last security token is then used to access the service (8).

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 10, October 2022)

1) SA sends certificate to every signer First, SA generates the private/public key for itself based on RSA algorithm. Each stage of generating key is described as follows.

Step 1. SA chooses two large prime number p and q ;

*Step 2. SA calculates n= p*q , n is a modulus of RSA;*

Step 3. SA calculates Φ(n),Φ(n)= (P-1)(q-1);

Step 4. SA chooses e randomly, and e should satisfy 1< e <Φ(n) and gcd(e,Φ(n))=1 . Thus （e, n） is the public key of SA ;

Step 5. SA calculates d by ed=1 mod Φ(n). （d, n） is the private key of SA ;
Let H() is a one-way hash function such as SHA, MD5. In order to distribute certificate to every Ui , SA selects a one-way hash function and calculates the following formula by using it's private key .

hi=H(IDi)
si= hi -d(mod n)
Finally, SA sends (IDi,si)as certification to every Ui∈G
via a secure channel. Then every signer calculates hi = si -e(mod n) to verify (IDi,si) [4] .

2) Documents division

Xpath, the XML Path Language, is used to locate information inside an XML document. DD utilizes Xpath to locate a subdocument of an XML document, and decomposes the document M into a set of subdocuments M={M1,M2,…,Mn}. Mi is delegated to Ui .

Suppose that the following XML document is needed to sign.

<?xml version="1.0" encoding="gb2312"?><order>
<bookinfo>
<title> data structure </title><author>YAN Wei-min</author>

<publisher>Tsinghua University publishing house </publisher>

<ISBN>0-764-58007-8</ISBN><price>40.00 </price>
</bookinfo>
<creditcard>
<name>XingShan</name>
<number>123456789</number>
<expiry>7/20/2008</expiry>
</creditcard>
</order>

We can use Xpath transformation show as <Transform
Algorithm=http://www.w3.org/TR/1999/REC-xpath-199911 16>
<XPath> descendant-or-self::creditcard </XPath></Transform>

The result of the Xpath transformation is shown as <creditcard>

<name>XingShan</name>
<number>123456789</number>
<expiry>7/20/2008</expiry>
</creditcard>

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 10, October 2022)

3) The process for generating and verifying a multi-signature

In our scheme, let M be the XML document to be cooperative signed by the signers in G. Let T ={t_1 , t_2,…, t_m} be a set of rules that is Xpath expressions. Let T_i and M_i be an element of T and an element of M delegated to user U_i respectively. Through a simple Xpath processor C with the rule T_i, one can easily obtain a subdocument $M_i=Ct_i(M)$.

Step l. Each $U_i \in G$ chooses a number r_i randomly, and calculates $R_i = r_i^e$（mod n）, then sends R_i to SC.

Step 2.SC calculates $R= R_1 *…* R_n$（mod n）, and broadcasts R.

Step 3.DD sends {H(M),M_i, T_i}and{H(T), H(M)}to U_i and SC, respectively.

Step 4.Each signer U_i verifies M_i which has been received by calculating $Ct_i(M)$. If $M_i=Ct_i(M)$, it indicates that M_i is the sub -document which U_i need to sign. All $U_i \in G$ cooperatively check the integrity of M by verifying $H(M)=H(M_1||M_2|| … ||M_n)$ where "||"is the concatenation symbol.

Step 5.Each U_i calculates the following formula.
$m_i=H$（R, M_i）
$D_i=r_i s_i^m$（mod n）

Each U_i sends his own signature result D_i to SC(the signature collection module).

Step 6.SC calculates $D = D_1D_2…D_n$ and $m= m_1 m_2… m_n$, and publishes（D, R, m）as the multi-signature result of M for G.

Signature collection module SC acts as the verifier and SC needs to carry on the following operation.

Step 1.SC calculates $h_1 = H（ID_1）$,…, $h_n = H（ID_n）$.

Step 2.SC calculates both $T^*=D^e（h_1h_2…h_n）^m$ $m^*=H（M, T^*）$

Step 3.SC checks the result of multi-signature by verifying m = m*. If m = m* is tenable, explained the multi-signature correctly.

The security analysis of this scheme

Because of the public ID_i of each signer, anybody can calculate H（ID_i）, but can not calculate $s_i= h_i^{-d}$(mod n) . Because two big prime numbers p,q are unknown, and Φ(n) is also unknown. So nobody can get d from e. The difficulty for solving d is equal to decomposing a great integer.

Suppose that the attacker wants to sign M(the document needs to sign) as a signer U_i, from above signature process, the attacker can choose a number ri',and calculate Ri'= $(r_i')^e$mod n）and $m_i'=H（R_i',M_i）$ to counterfeit R_i and m_i. But he can not counterfeit D_i, because he cann't get s_i, thedifficulty for solving the si is equal to decomposing a great integer[1,4] .

IV. CONCLUSION

The conventional digital signature technology can not sufficiently consider the characteristics of XML such as structure and description.

Moreover, it can not meet the new XML requirements of more fine-grained encryption and signature, and multiple signatures. This XML documents multi-signature scheme proposed in this paper has fully considered the advantages of XML documents structure, which is based on the RSA broadcasting multi-signature. This scheme uses the Xpath transform rule of the XML correlation technique to compartmentalize the documents, and the signer needs to sign for their own sub-document. This scheme improves the efficiency of the signature and correspondence, and it has certain extension and flexibility.

REFERENCES

[1] hao ze-mao.theory of digital signature.beijing of china:science press,2007.

[2] tzong-chen wu,shih-chan huang,d.j.guan. delegated multi-signature scheme with document decomposition.journal of systems and software,2001,55(3):321-328.

[3] meng jiang,cao li-ming.design and realization on multisignature scheme of xml-based electronic medical record.computer engineering,2006,32(19):264:266

[4] zhang jian-hong,wei yong-zhuang,wang yu-min.digital multisignature scheme based on rsa. journal of china institute of communications,2003,24(8):150-154

[5] zhu sheng-lin,xiao de-qin, lin pi-yuan.research and implementation on multisignature of electronic official documents based on xml. journal of south china agricultural university,2005,26(1):115-118

[6] hu yingsong, liu zhipeng. a multisignature scheme with xml document decomposition, net security technologies and application,2006,(7):87-89

[7] hakim khali, ahcene farah,dsa and ecdsa-based multi-signature schemes.ijcsns international journal of computer science and network secutity,2007,7(7) :11-18