The Application of XML Key Management Specification in e-Business

ZhiYong Gan
Department of Compute Science
South China Normal Univerity
GuangZhou China
ganzhiyong@21cn.com

YongPing Liu
Department of Apply Mathematics
South China Univerity of Technology
GuangZhou China
maypliu@scut.edu.cn

Abstract

The technology of PKI can satisfy most of the needs of security in e_Business. But it is restricted to use the PKI technology adequately because applications can not know the PKI System very well without an uniform architecture standard of PKI system. Now, XKMS gives the standard of key management, so applications can use the underlying PKI system transparently by using the standard interfaces. In this paper, we introduce the XKMS Web Services and the client software development kit (brief as CSDK) developed by ourselves. The web services and the CSDK are all in accord with the XKMS specification.. We also have developed an application system about e_Business to see how to call the web services by using the CSDK.

Key Words: XKMS; PKI; Web Services

1. Introduction

Information security has become the very hot topic, especially the rising of electronic commerce make it more important. The Public Key Infrastructure (PKI) can satisfy most of the needs of information security. The using of PKI is restricted because it needs that applications can know the architecture of PKI very well. XKMS[1] can solve the problem well. In this paper, the research state of XKMS is introduced firstly. Secondly, we introduce the XKMS web services[2] and the client software development kit developed by ourselves in detail. At last, we introduce an application of e_Business briefly which is used to demonstrate that the XKMS web service and CSDK can solve the security problem in e_Business or not.

2. Overview of XKMS

2.1 Research Background of XKMS

Information security has become the very hot topic, especially the rising of electronic commerce make it more important. The Public Key Infrastructure (PKI) can satisfy most of the needs of information security. The using of PKI is restricted because it needs that applications can know the architecture of PKI very well, but most of applications, such as database and deal processing software, could know it very poor.

World Wide Web Consortium (W3C) has issued XML Key Management Specification(XKMS) which is proposed by Microsoft, VeriSign and WebMethods. The specification could simplify the integration of PKI and digital certificate by using XML applications. PKI is made to be transparent to any application. So programmer can add the function about XML signature and XML encryption easily in the application of e_Business.

As XML is an extensible language, the information expressed by XML can be known by different systems. It is good at sharing data

between different systems. Furthermore, it will promote the application of PKI widely in internet.

2.2 Present Research Condition of XKMS

W3C had issued XML Key Management Specification(XKMS) which was proposed by Microsoft,VeriSign and WebMethods in November 27,2000. It denotes that the research of XKMS has been a rudiment. W3C issued XKMS version 1.0 in March 30,2001. It includes XML Key Information Service Specification(X_KISS)[1] and XML Key Registration Service Specification(X_KRSS)[1]. X_KISS includes “locate service” and “authentication service”. X_KRSS includes “registration service”, “revocation service”, “key recovery service”. In XKMS it gives not only the format of request message of client and response message of server, but also the description of element and compute formula. W3C issued XKMS version 2.0 in August 1,2002 and “reissue service” was added in X_KRSS. W3C is to modify XKMS continually in order to make it more well. The latest working draft is the one modified in April 21,2003. In the meanwhile, W3C issued other specifications, such as XML Signature Specification (XML_SIG)[3], XML Encryption Specification (XML_ENC)[4] and XML Query Specification (XML_Query)[5]. These specifications can provide valid security service for e_Business application.

With developing of XKMS, many large corporations has developed a lot of products to support XKMS. VeriSign corp. gives a XKMS Java SDK which can be ued to develop XKMS client rapidly. Entrust corp. has accomplished the function of X_KRSS now. The function of X_KISS will be provided in the near time. Microsoft also supports XKMS in ASP.NET. Evicible corp. has embedded XKMS in their product named as Evicible INK which can link to Trust Web Service transparently.

2.3 Inclusion of XKMS

There are two part of contents in XKMS. One is X-KISS (XML Key Information Service Specification), the other is X-KRSS (XML Key Registration Service Specification). The former includes “locate service” and “authentication service”. The later includes “registration service”, “reissue service”, “revocation service” and “key recovery service”. There are request message and response message in every service. Request message is made by client and analyze by server. Response message is on the contrary.

2.3.1 X_KISS

A protocol to support the delegation by an application to a service of the processing of Key Information associated with an XML signature, XML encryption, or other public key. Its functions include the location of required public keys and describing the binding of such keys to identification information

X-KISS allows a client to delegate part or all of the tasks required to process XML Signature <ds:KeyInfo> elements to a Trust service. A key objective of the protocol design is to minimize the complexity of applications using XML Signature. By becoming a client of the trust service, the application is relieved of the complexity and syntax of the underlying PKI used to establish trust relationships, which may be based upon a different specification such as X.509/PKIX, SPKI or PGP.X-KISS.

By design, the XML Signature Specification does not mandate use of a particular trust policy. The signer of a document is not required to include any key information but may include a <ds:KeyInfo> element that specifies the key itself, a key name, X.509 certificate, a PGP Key Identifier etc. Alternatively, a link may be provided to a location where the full <ds:KeyInfo>

information may be found.

2.3.2 X_KRSS

A protocol to support the registration of a key pair by a key pair holder, with the intent that the key pair subsequently be usable in conjunction with the XML Key Information Service Specification or higher level trust assertion service such as XML Trust Assertion Service Specification [XTASS].

The XML Key Registration Service Specification permits management of information that is bound to a public key pair. The XKRSS service specification supports the following operations:

- Register

Information is bound to a public key pair through a key binding

- Reissue

A previously registered key binding is reissued.

- Revoke

A previously registered key binding is revoked.

- Recover

The private key associated with a key binding is recovered.

The Register operation does not in itself place any requirement on the Registration Service to communicate that information to any other party. In most applications, however, a Registration Service will provide key information to other trust services such as those described in the XKMS specification or a separate underlying PKI such as PKIX.

3. Main Working

In this section, we will introduce our main working, including the XKMS web services, CSDK and the applicaion of e_Business. All of them are developed by our lab.

3.1 Former Working Flow of out PKI System

Our lab has developed a PKI system. Figure3-1 shows the working flow.

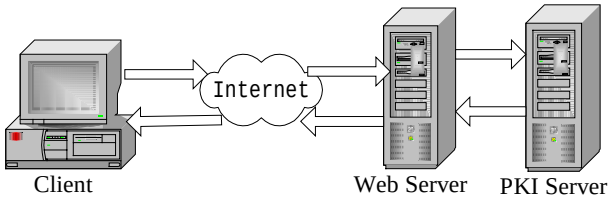

Figure 3-1 former working flow

Without a unified standard of PKI system, so user can visit it only by explorer. The function of PKI system can not be embedded in an application.

3.2 What To Do

In order to make user to use PKI system conveniently, we accomplished the functions of X_KISS and X_KRSS and deployed them as web services. In the meanwhile, we also developed the CSDK which is provided to user. We call the PKI System developed by ourselves as underlying PKI system. The CSDK gives the interfaces to call the web services. The details of underlying PKI system have been hidden and it is fully transparent to applications. Applications can only use the interface functions provided by the CSDK to operate the underlying PKI system by sending XKMS request message and receiving XKMS response message. Applications is not concerned with the architecture of the underlying PKI system .

We deploy the web services into XKMS server. XKMS server receives the request message from client and transmits it to the underlying PKI system. After processing the request message, the underlying PKI system sends the result to XKMS server. At last, XKMS server send the result to client.

In the meanwhile, we developd an application named as Fund Project Appraisal System By Experts Through Internet. We hope that we can solve the security problem of the application by calling the XKMS web services with the CSDK. It is the fact that the application can do it without knowing any detail of the underlying PKI system.

3.3 How To Do

We have developed server software module and client module module. The server software module accomplishs the functions of X_KISS and X_KRSS. All the functions of XKMS are deployed as web services in the server named as XKMS server which is supported by Apache tomcat[8] server. CSDK gives interface functions to user.

SOAP[6] protocol is used to correspond between in XKMS server and client. Client uses RPC (Remote Procedure Call) to call the XKMS web services deployed in the XKMS server. Sender packs up XKMS message to SOAP message and sends it to receiver. Receiver gets the valid information form the SOAP message received from sender. In order to make the underlying PKI system not to modify more, XKMS server corresponds to the underlying PKI system with HTTP protocol which can be known by it. So XKMS server should know the architecture and details of underlying PKI system. On the one hand, XKMS server would convert the request message received from client to the one which can be known by the underlying PKI system. On the other hand, it would get the valid information from the processing message returned from the underlying PKI system. After that, it would convert them to XKMS response message and send it to client. The correspondance between them is fully transparent to an user. The user should only know the functions of the XKMS web services in XKMS server. It enlarges the area to use a traditional PKI system . Figure 3-2 shows the working flow.

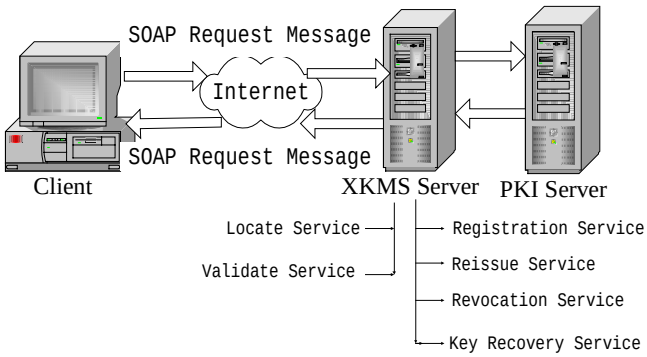

Figure 3-2 working flow

3.3.1 Accomplish X_KISS

X_KISS includes “locate service” and “authentication service”.

(1) Locate Service

“Locate service” includes “locate public key” and “locate certificate”. When an user received a signature document and the corresponding certificate, he would want to know that the signature is valid or not. As client couldn’t analyze the X.509 certificate, he need to send the request message of “locate public key” to XKMS server.. The certificate contains the public key should be included in the request message. Since XKMS server is able to analyze a X.509 certificate, it would analyze the certificate by itself after receiving the request message of “locate public key” from client. There is no any request to be sent to the underlying PKI system. XKMS server sends response message which includes the public key got from the certificate.

When client sends “locate certificate” request message, we add some function to XKMS because there are more than one certificate in the locate result. We divide “locate certificate” into “locate certificate list” and “locate specified certificate”. When client sends “locate certificate” request message, we will

locate certificate list firstly. XKMS server sends a locate certificate request message to the underlying PKI system and the certificate list according with locate condition will be returned to XKMS server. There are certificate serial number, DN, issue date, invalid date, key length, revocation date, certificate status, Email algorithm and certificate type in the certificate list. As HTTP protocol is used in the corresponding between XKMS server and the underlying PKI system, XKMS server should process the response message from the underlying PKI system and find the data demanded by the certificate list. After finished it, XKMS server would send a response message containing these information with SOAP protocol to client.

The user selects one certificate from the certificate list and sends a "locate specified certificate" request message containing the serial number of the certificate to XKMS server. After receiving it, XKMS server sends a locate certificate request message containing the certificate serial number received form client to the underlying PKI system. The underlying PKI system will locate the certificate with the serial number and locate the CA certificate . After that , two certificates will be sent to XKMS server. As the same reason, XKMS server also needs to process the return message and get the two certificates. A response message contains the two certificate will be sent to client from XKMS server.

The method to process the HTTP message received from the underlying PKI system in XKMS server is to look the HTTP message as a string and analyze its structure. The method in other service is the same when XKMS server communicates with the underlying PKI system.

(2) Authentication Service

When client receives some signature Email and corresponding certificate, client need to authenticate the certificate is valid or not, is revocatory or not, is the sender's one or not etc. Since client has not the authentication function, it needs to send "authentication certificate" request message to XKMS server. The request message contains the certificate which would be authenticated and its certificate chain.after receiving the request message, XKMS server gets the serial number of the certificate and locates the certificate using the serial number from the underlying PKI system. The underlying PKI system returns a certificate list which contains the certificate. In this certificate list there are a lot of information about the certificate. XKMS server finds the certificate status and the public key of the certificate after processing the return message. The certificate status and the public key are all sent to client.

3.3.2 Accomplish X_KRSS

X- XRSS includes "registration service", "reissue service", "revocation service" and "key recovery service."

(1) Registration Service

The Register request is used to assert a binding of information to a public key pair.. Generation of the public key pair MAY be performed by either the client or the Registration service. Registration service can registry not only a personal certificate but also a server certificate.

The information used to registry a personal certificate are as follows.

Name: CN
Department: OU
Unit: O
City: L
Provice: SP
Country: C
Email：E

All the information are binding to the

certificate. Moreover, key length, authentication code and revocation code are all necessary.

Certificate request encoding (refer to the standard of PKCS#10[7]) is the only required to registry a server certificate. Registration service can get the necessary information from the encoding. Authentication code and revocation code are also necessary

Before registration , client needs to get an authentication code with offline mode from XKMS server. The authentication code is used to authenticate his request. Client selects a revocation code to authenticate itself should be necessary to revoke the registration at a later date If the key pair is generated by client, client should make a signature of a string with the private key to demonstrate that it owns the private key. If the key pair is generated by server, server should encrypt the private key and send it to client after finishing the registration. In this case, the private key can only be an encryption key . To be a signature key is denied.

After finishing the registration, client will receive a message to show the registration is successful or not. If success, client will save the private key generated by client or received from server to a temporary file. After the RA and CA all pass the registration, the private key stored in the temporary will be stored into a keystore file. In the keystore file the serial number of corresponding certificate will be the alias name of the private key.

XKMS server will authenticate the request is been authorized or not after received a registration request. If the authorization is passed, XKMS server still need to distinguish which type of certificate is registried. The certificate request encoding will be sent to the underlying PKI system directly when it is to registry a server certificate. If it is to registry a personal certificate XKMS server should distinguish how to generate a key pair.. If the key pair is generated by client XKMS server should authenticate that the user own the private key or not. If the authentication is also passed, XKMS server will get public key, DN and other information from the request message. If the key pair is generated by server, XKMS server will get DN and other information form the request message. XKMS server sends to the underlying PKI system a request message which contains public key , DN and other information. The underlying PKI system will registry a certificate with these information and return a response message to XKMS server to denote the registration is successful or not. XKMS server convert this message to a XKMS message and send it to client.

(2) Reissue Service

The registration service may permit clients to reissue previously issued assertions. The reissue request is made in the same manner as the initial registration of a key. The principal reason a client would make a reissue request is to cause the registration service to generate new credentials in the underlying PKI system, e.g. X.509 Certificates. The key and DN will not vary in the new certificate.

(3) Revocation Service

The revocation service may permit clients to revoke previously issued assertions. Clients should give the revocation code contained in the registration request message to authenticate its identity.

The working flow is as follows.

Client send a revocation request message contains the certificate to be revoked and the revocation code to XKMS server. XKMS server gets the revocation code and the serial number of the certificate form the request message and send them to the underlying PKI system. XKMS server will return to client a response message contains the information about revocation is successful or not after receiving the return

message from the underlying PKI system.

(4) Key Recovery Service

The key recovery service may permit clients to request to recover the backup private key in the XKMS server. The backup private key may be the one generated by server. It also may be the one generate by client. If the key is generated by client, it should be encrypted with the public key of server certificate and sent to XKMS server. Only encryption key can be backup and recovered. XKMS server will revoke the private key whenever key recovery is performed. Client also needs to get an authentication code with offline mode from XKMS server. The authentication code is used to authenticate his request is authorized by XKMS server or not.

The working flow is as follows.

Client computes the digest of the authentication code and makes the signature of the digest with signature private key. The digest, signature and signature public key will be sent to XKMS server After receiving the request message, XKMS server will validate the signature. If the validation is right, XKMS server will send a request message to the underlying PKI system to request key recovery. XKMS server encrypts the private key returned by the underlying PKI system with the user's signature public key. And sends it to client. Client will decrypt the encrypted private key with its signature private key after receiving the response message from XKMS server. At last, the private key will be stored into the keystore file.

3.3.3 Explanation ff Correlative Computation

(1) Computation of Authentication Code

Computation：

Number "0X1" is as the seed to generate a random key. Algorithm HMAC-SHA1 is used to compute the digest of authentication code. The formula is as follows.

digest = HMAC-SHA1 (authentication code, 0X1)…….(1)

For example, if the authentication code is "024837". The formula is as follows.

digest = HMAC-SHA1 ("024837", 0X1)……(2)

Then we will make the signature of the digest with the private key K_{pri} generated by client. The formula is as follows.

signature = K_{pri} (digest)……(3)

Authentication:

XKMS server decrypts the signature with the public key K_{pub} received from client to get the original digest'. The formula is as follows.

digest' = K_{pub} (signature)= K_{pub} (K_{pri} (digest))……(4)

Compare digest' with digest firstly. It demonstrates that the signature is right if they are equation. Or else the signature is wrong. Secondly, XKMS server computes the digest" with authentication code "024837" at the same method. The formula is as follows.

digest" = HMAC-SHA1 ("024837"，0X1)…...(5)

Compare digest" with digest. If they are equation, the request is passed. Or else it is not passed.

(2) Computation of Revocation Code

The compute method of revocation code resembles as the compute method of authentication code. But the digest should be computed twice. when XKMS server authenticates the request it is the only step to validate the signature.

Computation:

Firstly, Number "0X1" is as the seed to generate a random key. Algorithm HMAC-SHA1 is used to compute the digest of authentication code. The formula is as follows.

digetst1 = HMAC-SHA1 (revocation code, 0X1)…….(6)

Secondly, number "0X1" is as the seed to

generate a random key. Algorithm HMAC-SHA1 is used to compute the digest of digetst1. The formula is as follows.

digest = HMAC-SHA1 (digetst1, 0X2)

= HMAC-SHA1 (HMAC-SHA1 (revocation code，0X1), 0X2) ….……...…(7)

The digest is the result.

For example, if the revocation code is “I Will Reveal My Key”. The formula is as follows.

digest = HMAC-SHA1 ((HMAC-SHA1 (“I Will Reveal My Key”，0X1)，0X2)……..(8)

Then we will make the signature of the digest with the private key K_{pri} generated by client. The formula is as follows.

signature = K_{pri} (digest)……(9)

Authentication:

XKMS server decrypts the signature with the public key K_{pub} received from client to get the original digest’. The formula is as follows.

digest’ = K_{pub} (signature) = K_{pub} (K_{pri} (digest))……..(10)

Compare digest’ with digest firstly. It demonstrates that the signature is right if they are equation. It shows that the client owns the private key. Or else the signature is wrong and the client doesn’t own the private key.

3.3.4 develop an application

Furthermore, we have developed an e_Business system named as Fund Project Appraisal System By Experts Through Internet. The system includes five main steps. The first step is researcher applys a project. The second step is to authorize by college and technology department. The three step is the subject director selects some experts to appraise this project. The four step is the experts give their appraisal opinion. The last step is the subject director confirms the project is passed or not. All of these operations are doing through internet. It demands that the system can not only encrypt the data transmitted in the internet but also validate the identity of all kinds of users.

We have embedded our CSDK into the system. The system calls the XKMS web services by CSDK to satisfy the need of security. The fact is the CSDK can solve the security problem easily in e_Business.

4. Further Working

We have accomplished the six kinds of service functions of XKMS. But we should make them more sophisticated in order to adapt new needing because XKMS is developing constantly.

The main working we have done is only adapted for the PKI system developed by ourselves. Our next plan is to make the web services and CSDK adapting for more different PKI system to enhance their commonality. So we will provide a group of standard interfaces for underlying PKI system for the use of sending data to it or receiving data from it. After accomplishing these working, the format of input data or output data of all kinds of PKI system with different architecture is unified and a standard user interface of PKI system is provided for users through our XKMS web services.

5. Conclusion

With the using of PKI in e_Business becoming more and more widely, XKMS would be more important. XKMS could not only develop following the development of PKI, but also promote the development and using of PKI technology.

6. Reference

[1] W3C Recommendation “XML Key Management Specification (XKMS) 2.0”, 2002

[2] W3C Notes “Web Services Description

Language (WSDL) 1.1”, 2002
[3] W3C Recommendation “XML-Signature Syntax and Processing” 2002
[4] W3C Candidate Recommendation “XML Encryption Syntax and Processing”, 2002
[5] W3C Working Draft “XML Query Use cases”, 2002
[6] W3C Note”Simple Object Access Protocol (SOAP) 1.2” 2002
[7] [PKCS#10] Certification Request Syntax Standard (v1.7), 2000
[8] The Jakarta Site---Apache tomcat http://jakarta.apache.org/tomcat/