International Journal of Research in Advent Technology, Vol.2, No.3, March 2014

E-ISSN: 2321-9637

Implementation of X.509 Certificate for Online Applications

Sapna Sejwani[1], Sarvesh Tanwar[2]

Student, Master of Technology II Year, Computer Science and Engineering[1], , Assistant Professor, Computer Science and Engineering[2], Mody University of Science and Technology, Laxmangarh, Rajasthan, India[1, 2]

Email: sapnasejwani@gmail.com[1]

Abstract- With the globalization in the e-commerce, where everything is digital and is done online, may it be online shopping, money transfer, e-banking, e-voting, e- registration, sending email, security is the main priority. Reliance on electronic communications makes information vulnerable to unauthorized users. Hence the users need confidentiality, message integrity, sender non-repudiation and sender and authentication. Public Key Infrastructure provides these services. And it ensure that public keys are public keys are securely, conveniently and efficiently are distributed. There are many types of PKI implementations. A X.509 certificate binds a name to public key value. The role of certificate is to associate a public key with the identity contained in the X.509 certificate. In this paper we are discussing the implementation of certificate X.509, how it is generated and stored in database and retrieved when required.

Index Terms- Digital certificates, Digital signature, Private Key, Network of trust, Public key, Symmetric encryption, Trust model, X.509 certificate.

1. INTRODUCTION

The Internet provides an excellent vehicle for extending the scope of communication and business through electronic means (e-commerce). The growth of computer networks and the opening that their interconnection brings, through Internet, mean that a great amount of information is being travelled through network and crossing numerous intermediate systems. This results in the increase of the number of possible attacks both active and passive. In order to protect networks from these dangers it is necessary to provide security services. They should guarantee the identity of the communicating parties(digital certificates), the prevention that communicating parties won't deny the transmitted message (non repudiation), the protection against unauthorized writing (authentication) and, in some cases, unauthorized reading of transferred data (confidentiality is hindered). These services of authentication, non repudiation, integrity and confidentiality, respectively, can be provided using cryptosystems [1]

1.1 X.509 CERTIFICATE

An X.509 certificate binds a name to a public key of a user. The aim of the certificate is to associate a public key with user identity contained in the X.509 certificate.

Authentication of secure online applications depends on the integrity of the public key of the user in the application's certificate. If an attacker replaces the public key with his public key, it can impersonate the true application called masquerade and gain access to confidential data. To prevent this type of attack, all certificates must be signed by a trust worthy third party called a certification authority (CA). A CA confirms the integrity of the public key of the user in a certificate. A CA signs a user certificate by appending his digital signature in the certificate. A digital signature is a message digest of all the fields of certificate encoded with the CA's private key. The CA's public key is available to the required applications by distributing a certificate for the CA. These Applications verify the authenticity of certificate by decoding the CA's digital signature with the CA's public key and performing hash function on all the fields of certificate and hence comparing it.

Probably the most widely noticeable application of X.509 certificates today is in web browsers (such as Microsoft Internet Explorer and Mozilla Firefox) that support the TLS protocol (Transport Layer Security). TLS is a security protocol which provides authentication and confidentiality for our network traffic.

Other technologies that rely on X.509 certificates technology include:

- E-Commerce protocols, such as SET.

- Various code-signing schemes, such as signed Microsoft Authenticode and Java ARchives.

- Various secure E-Mail standards, such as S/MIME and PEM.

1.2 X.509 CERTIFICATE FIELDS

Figure 1 illustrates the structure of an X.509 v3 certificate. Each organization evaluates its business needs relative to the constructs of the public key certificates that it wishes to issue [6,7].

Version – This identifies which version of the X.509 (version 1/2/3) standard applies to this certificate, which affects what information can be specified in it.
• **Serial number** - The entity (CA/ RA) that created the certificate is responsible for assigning it a serial number, unique identifier, to distinguish it from other certificates it issues.
• **Issuer name** - This is the X.500 name of the entity that signed the certificate. This is normally a CA in few cases is RA. Using this certificate implies trusting the entity that signed this certificate. In some cases, such as root CA certificates, the issuer signs its own certificate as he is the top authority [8].
• **Signature algorithm identifier** - This identifies the algorithm used by the CA to sign the certificate (e.g., RSA, DSA, etc.)
• **Validity period** - Each certificate is valid only for a limited amount of time. This period is described by a start date and time and an end date and time.
• **Subject name** - The name of the entity whose public key the certificate identifies. This name uses the X.500 standard, so it is intended to be unique across the Internet. This is the Distinguished Name (DN) of the entity, for example, CN(Common Name)=SapnaSejwani,OU(OrganisationUnit)=Engineering,O(Organisation)=MUST,L(Location)=Sikar,S(State)=Rajasthan,C(Country)=India [2].
• **Subject public key information** - This is the public key of the entity being named, together with an algorithm identifier which specifies which public key crypto system this key belongs to and any associated key parameters.
• **Issuer digital signature** - This is the digital signature of the issuer

FIELDS OF X.509 CERTIFICATE VERSION 3
1. Version (Of certificate format)
2. Certificate Serial No.
3. Signature Algorithm Identifier (For CA's Signature)
4. Issuer X.509 Name (Certification Authority)
5. Validity Period (Start and expiration date/time)
6. Subject X.509 Name
7. Subject Public Key Information a. Algorithm Identifier b. Public Key Value
8. Issuer Unique Identifier
9. Subject Unique Identifier
10. Extensions
11. Issuer's Digital Signature (Certificate Authority)

Figure 1: Structure of X.509 Version 3 Certificate

2. DESIGN AND IMPLEMENTATION

2.1 Technologies:

- Front end : Java Development Tool Kit (version 1.3), Swing, Socket programming
- Back End : SQL server management studio

2.2 Application use:

- Multi-level login includes the following levels: new-visit to the application, additional visits to check and update of the personnel Information.

- Digital signature provides enhanced security.

- Create digital certificates (X.509 version 3 format) of users of different unit in an organization.
- Maintain digital certificates of all the members according to time stamp (valid from, valid to).
- Generate key pair using following public key generation algorithm:
  i. RSA
  ii. Digital Signature Algorithm
  iii. Diffie Hellman
- Encryption of digital certificate using RSA algorithm ensures confidentiality.
- Uses MD5 with RSA and SHA1 with DSA for signature generation.

2.3 Three tier architecture model

This application is based on three tier architecture.

Client: User who is requesting for the certificate.

Server: CA who fires has access to database. And is ready for listening request from user for granting the certificate.

Database: Where all the certificates are stored. (Figure 2).

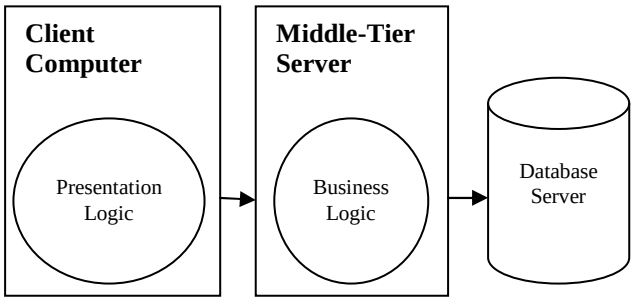

Figure. 2: A three tier architecture model

2.4 Code

(a) Server socket ready to accept the request from client.

```
try{
    while(true){

    Socket client=serversocket.accept();
    Connection1 con=new Connection1(client);
            }
    }
```

(b) JDBC-ODBC connectivity with "certificate" as database name, "sa" is user name and "sapna1234" as password.

```
try{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver";
con=DriverManager.getConnection("jdbc:odbc:c
ertificate","sa","sapna1234");
    }
```

(c) X.509 certificate creation

```
X500Name x500Name = new X500Name(CN,
OU, O, L, S, C);
pkcs10_RSA.encodeAndSign(new
X500Signer(signature_RSA,
x500Name));ByteArrayOutputStream bs = new
ByteArrayOutputStream();
```

(d) Signing the certificate with private key of the issuer and encrypting it with issuer's public key.

```
System.out.println("\nSigning
withMD5withRSA");
sigAlg = "MD5WithRSA";PKCS10 pkcs10_RSA
= new PKCS10(publicKey_RSA);Signature
signatureRSA = Signature.getInstance(sigAlg);
signatureRSA.initSign(privateKey_RSA);
resultstr_sign_RSA=signatureRSA.sign();
```

2.5 Snapshots

Step 1: When a new user is there, it fills the sign up consisting of various fields required for certificate generation (Common name (CN), Organization unit (OU), Organization name (O), Country (C), State (S), Location (L), contact details etc.) form it is stored in database. Figure 3.

Step 2: When CA accepts the certificate creation. The CA then generates the key pair for the user and generates his/her digital certificate and signs his/her certificate with CA's private key [4].

Step 3: The user's certificate is then decrypted by his issuer's (in his/her case its CA) public key. Also, his private key which was encrypted by his/her issuer's private key is retrieved and is decrypted by issuer's public key.

Step 4: The user can now view his digital certificate which has his public key in it. Also it has all the details regarding validity period of certificate, issuer details, user's personal details. Figure 4,5 and 6 are the snapshots of the certificate in general view, detailed view and extensions of Certificate.

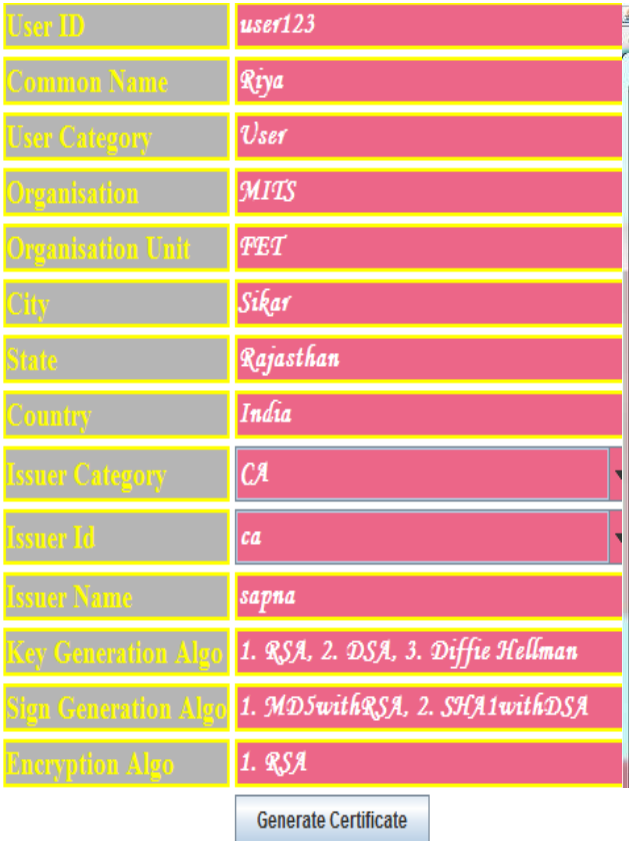

Figure 3: User's certificate filling form consisting of CN, OU, O, L, S, C fields

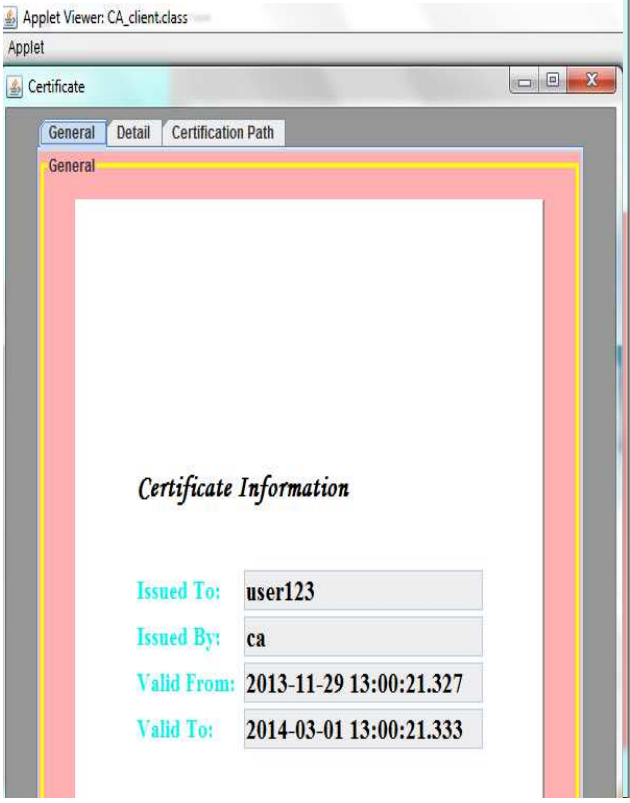

Figure 4: User's certificate: general view

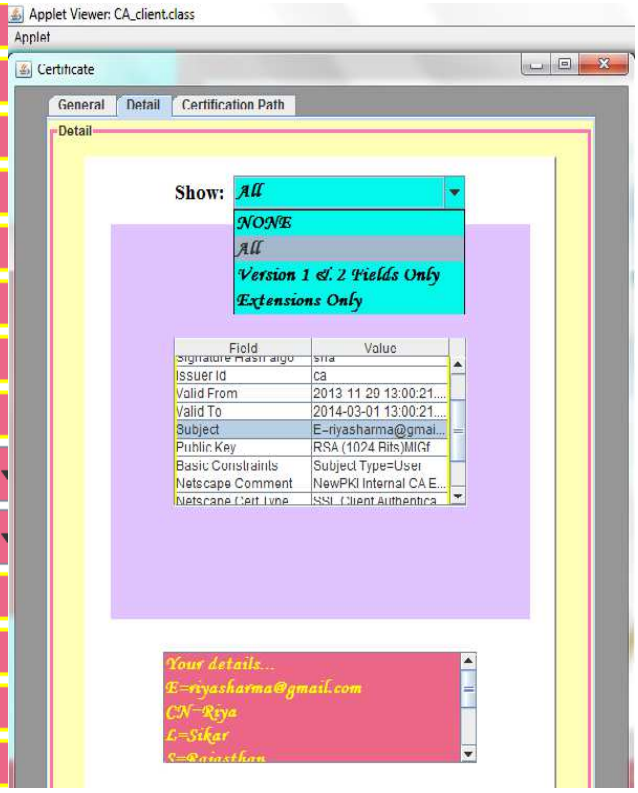

Figure 5: User's certificate: Detail view

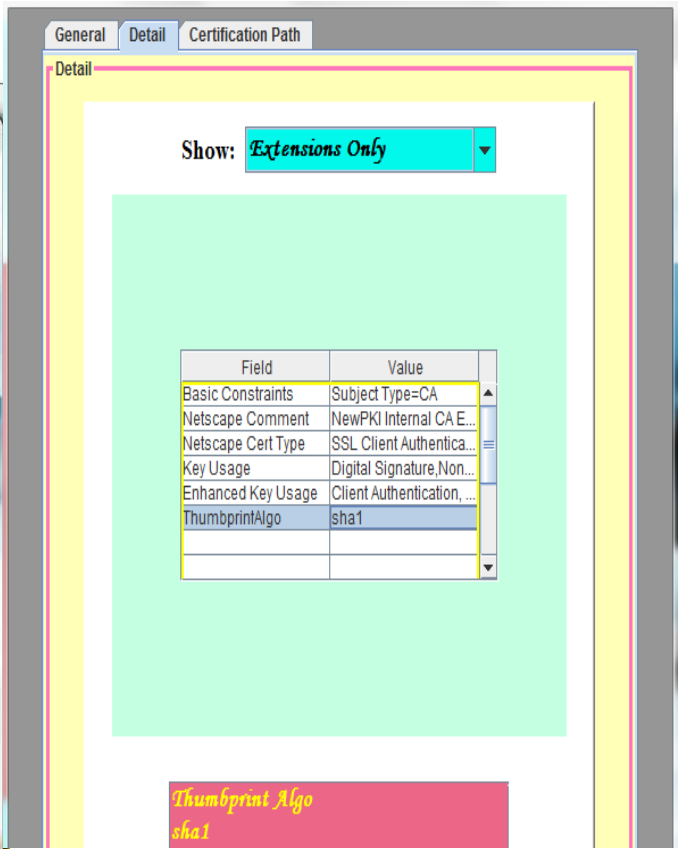

Figure 6: User's certificate: Extension Fields

serial_no	signature_algo	signature_hash...	valid_from	valid_to	public_key
9	sha1RSA ...	sha ...	2013-10-28 11:...	2014-01-28 11:...	RSA (1024 Bits) ...
NULL	NULL	NULL	NULL	NULL	NULL

Figure 7: Certificate storage in database

KRISHNA-PC\SQLE... - dbo.certiwww

	owner_public_key	owner_private...	owner_certificate	issuer_public...
▶	MIGfMA0GCSqG...	<Binary data>	<Binary data>	MIGfMA0GCSqG...
*	NULL	NULL	NULL	NULL

Figure 8: Private key and certificate in encrypted form (Binary) in database

CONCLUSION

This application is depicting the implementation of X.509 version 3 (with extensions). This application can be merged with any online transaction to provide **strong authentication, privacy, confidentiality, data integrity, tamper detection and non repudiation.** PKI uses public/private keys and includes the infrastructure to manage and maintain the keys, resulting in an electronic environment that is private, confidential, and legally binding. The industries are moving to PKI and certificates for safe internet transactions. X.509 is a mechanism for supporting large scale deployment of security, based on asymmetric cryptography. PKI (Public Key Infrastructure) is needed to support X.509. PKI is currently the only technology that provides the required level of data integrity and protection to support electronic commerce through X.509 certificate [3].

ACKNOWLEDGEMENT

We thank to **Prof. Prema K.V** (HOD, FET, CSE Dept, Mody University) , **Mr. Anurag Saxena** for their suggestions.

REFERENCES

[1] Al-Janabi, Sufyan Faraj, and Amer Kais Obaid. "Development of Certificate Authority services for web applications." Future Communication Networks (ICFCN), 2012 International Conference on. IEEE, 2012.

[2] Prohic, Natasa. "Public key infrastructures–pgp vs. x.509." INFOTECH Seminar Advanced Communication Services (ACS). 2005.

[3] Jancic, A., and Matthew J. Warren. "PKI-Advantages and Obstacles." AISM. 2004

.[4] Ten Have, Steven, Marcel vander. Elst, and Wouter ten Have. Key management models. Financial Times Prentice Hall, 2003.

[5] Hunt, Ray. "PKI and digital certification infrastructure." Networks, 2001. Proceedings. Ninth IEEE International Conference on. IEEE, 2001.

[6] Nykänen, Toni. "Attribute certificates in x. 509." Techn. Ber., Helsinki University of Technology (2000).

[7] Gerck, Ed. "Overview of Certification Systems: X. 509, PKIX, CA, PGP & SKIP." The Bell 1.3 (2000): 8.

[8] Curry, Ian. “Version 3 X.509 Cerificate.” Entrust Technologies(1996)