A Security Solution for the Transmission of Confidential Data and Efficient File Authentication Based on DES, AES, DSS and RSA

Rajat Chaudhary, Prem Singh, Ambika Agarwal

Abstract: Data security is an integral part of web based business applications like insurance, banking etc. These applications require a secure infrastructure to meet the security requirements of confidentiality, endpoint authentication, message integrity and no repudiation. Document encryption/decryption and signatures/validation are the data security standards that define XML vocabularies and processing rules to meet these security requirements. In this paper, we present a how a file securely passes from sender (server) to the receiver (client) through a central gateway with web services applications which mean a secure architecture for the exchange of confidential documents. Designing a secured electronic system architecture i.e. connected to the central gateway through which the whole work is established and takes place accordingly to pass only the files over the internet and have security features like digital signatures. Various algorithms, implementations and coding have been developed for encryption/decryption, signatures/validations and web services.

Index terms: Confidentiality, Authentication, Algorithms and Methods.

I. INTRODUCTION

Confidential document is sent by the sender to the receiver via the central gateway i.e. a java application. This application is implemented which offers some security services to the external computers. Security services include encryption, decryption, signatures and validation takes place during the whole process of the designed system. The Signatures and encryption takes place between the sender and gateway whereas the decryption and validation takes place between the receiver side and the gateway. Proper use of symmetric keys and asymmetric keys has been developed for the security services in the system. The system has stored some definite numbers of receivers email-ids to whom the document(s) will be send. If the sender wants to send a confidential file securely, the file must be encrypted first with the proper signatures and finally stored on the central gateway.

After receiving the email confirmation, the receiver accesses the central gateway and validates and decrypts the encrypted XML file. The numbers of encrypted files as stored on the gateway in the database are automatically deleted accordingly to the latest file i.e. created with respect to the date and time. In this way, all the encrypted files are received by the respective receivers and then decrypt it.

Also Web Services Security provides message level security for web services. It is a communications protocol which provides a means for applying security to WS. The protocol contains specifications on how integrity and confidentiality can be enforced on WS messaging. “WS-Security is controlled through WSSE (Web Service Security Encrypt) policy files.”[5] For accessing the secured web service with WS-Security, create a policy file and associate that file with the web service control. This should match the policy file of the target web service. If the target web service requires encrypted incoming data, then a control file targeting that web service should encrypt data before they are sent to the web service. The web service plays an important role while transactions in the whole process of the system. So in this way, there is a secure transmission of confidential documents from sender to the receiver.

Efficient authentication of files is also very necessary to check the authenticity of the right legitimate user. This is about verifying/validating the identity. In an open ecommerce system, a digital certificate is employed to satisfy the authentication requirement. So authentication is done both at the sender side and receiver side with the help of digital signatures and proper validation of their signatures. Integrity of document is also checked because this makes sure that if the content of a message is altered, the receiver can detect it. A digital signature is commonly used to ensure data integrity.

Here non-repudiation plays a very important role because this makes sure that if integrity and authentication can be ensured then the non-repudiation requirement can also be satisfied. So by the help of using various methods such as encryption, decryption, digital signature and validation also by applying various algorithms such as DES, AES, DSA and RSA all the requirements of security i.e. confidentiality, availability, authentication, integrity and non- repudiation is achieved.

II. CONCEPT OF CRYPTOGRAPHY

For building a secure transmission of documents, the cryptographic techniques must be employed. Cryptography is originally about keeping messages secret. Some of the common elements involved in cryptographic processes are public keys, private keys, key pair generator, key factories, key stores, and cryptographic algorithms. Cryptography systems can also be broadly classified as single-key or symmetric-key systems and two key or public-key systems. This concept is generally applied throughout the banks or companies for sending any confidential and private information/data.

Revised Manuscript Received on 30 August 2012

*Correspondence Author(s)
Rajat Chaudhary, Department of Computer Science, DIT (Uttarakhand Technical University)
Prem Singh, Department of Computer Science, DIT (Uttarakhand Technical University)
Ambika Agarwal. Department of Computer Science, DIT (Uttarakhand Technical University)

It is more than a technique for scrambling the information into a form i.e. only readable by authorized people, but is used to address different security requirements.

There are four message level security requirements named as follows:

1. **Confidentiality:** This makes sure that a message is kept confidential or secret such that only the intended recipient can read it. To provide data confidentiality, encryption is used.
2. **Integrity:** This makes sure that if the content of a message is altered, the receiver can detect it. A digital signature is commonly used to ensure data integrity.
3. **Authentication:** This is about verifying/validating the identity. In an open ecommerce system, a digital certificate is employed to satisfy the authentication requirement.
4. **Non-repudiation:** This makes sure that if integrity and authentication can be ensured then the non-repudiation requirement can also be satisfied. [4]

This paper must satisfy these above four basic and most important requirements through which the goal is completed. After satisfying these goals, the whole system should be secured. If anyone of these goals is unsatisfied then the system will not be secured. This goal can be reached by sending any secure confidential document to various banks, large corporations, IT companies and small businesses with universally accepted standards. An example of such a secure transmission of document - "For example, Receiver wishes to order and pay for a book from sender using the mutually trusted payment system Zip Pay (Prepaid card as issued by Meta-Bank). Sender creates an order form including the book title, price, and his/her account info. Now, the sender wants to sign all of his/her information, but will subsequently encrypt his/her account info for zip Pay only. Sender send this to receiver who affirms the book title and price, signs the form and presents the twice signed order with his/her own payment information to Zip Pay. To validate both signatures, Zip Pay will have to know that the cipher data version of the encrypted information is necessary for validating receiver's signature, but also the plain data form is necessary for validating sender's signature too. Since encryption operations applied to part of the signed content after a signature operation cause a signature not to be verifiable, it is necessary to decrypt the portions encrypted after signing before the signature is verified. The decryption transform provides mechanisms, decrypting first only the signed then encrypted portions.

In this way, a signer can insert this transform in a transform sequence i.e. canonical XML if there is a possibility that someone will encrypt portions of the signature. Here, the sender is the intranet/client side, receiver is the server side, and the Zip Pay is the gateway. In this way, this system is very useful in banking, companies, agencies, business purposes etc." [6][7]

III. VARIOUS SOLUTIONS FOR IMPLEMENTING THE SECURE TRANSMISSION

For fulfilling the system secure, there are some of the following solutions mentioned. These solutions are worked in both practical and theoretical way for implementing the java application i.e. the web service.

First Solution: Proper use of the WS-Security services in the system makes the system flow securely. These security services are: encryption, signatures, decryption, and validations. These services must be used in the proper format which means that, which service must have to come from initial to final position. The proper use of algorithms must be used for implementing the system and makes sure that these algorithms must support these security services in an appropriate manner. Otherwise, the error occurs and the system will not work properly. For developing these security services, Sun have developed some predefined API's for solving the problem of the system with the right way of using certain algorithms.

Second Solution: is to install the software that has to be used for developing some simple applications for creating web service. The client and the gateway should be implemented. The implementation of the gateway is the most critical and important part of the system which makes the whole process complete. Without this java application, the secure documents never think of passing across the system and lack the security purposes. For making the system very much secured, the gateway must be configured with the help of security services. The client is a simple application which helps for communicating with the gateway by any user. All the four methods i.e. encrypt, decrypt, sign and validate must receive a secure document according to the standards digital signatures and encryption. Proper uses of keys are essential for developing the security services. The java application should have the symmetric key for doing the encryption and decryption process whereas the public keys and private keys are used for signing and validating process. The third party of the system i.e. (GW) gateway makes the whole process quite complicated in implementing it but if it's implemented successfully then the whole process of the system is completely secured.

Final Solution: is to implement the server (receiver side) and the gateway so that both can easily communicate with each other. Coding of the predefined API's with the help of java programming which supports the java applications carefully. Make sure of the proper tools that must support these applications and the programming language in which the system is going to be implemented. These applications are quite complicated so that only visual tools fulfill the requirement of the system.

The outline design of the system comprises of the four methods i.e. encryption, decryption, signatures, and validations as shown below in the diagram.

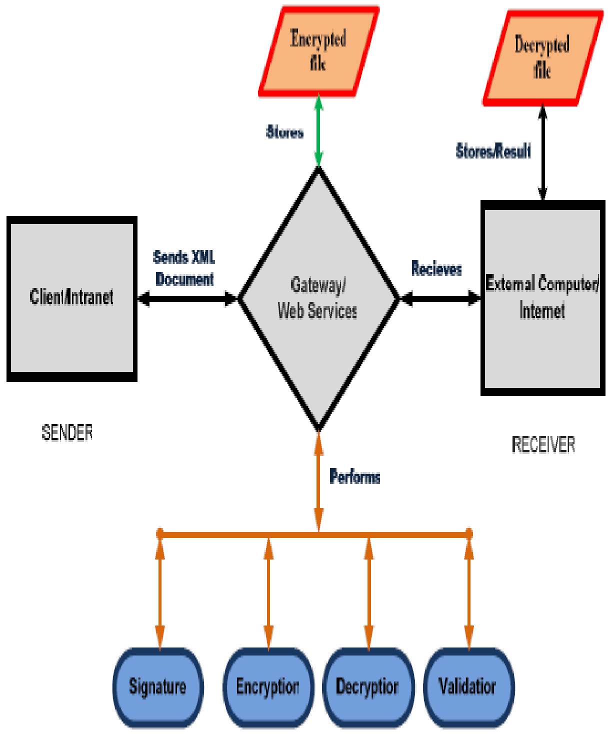

Figure: Outline Design of a Secure System

IV. CRYPTOGRAPHIC FUNCTIONS:

The following notation is used for describing the various cryptographic functions:

Output = Function algorithm [Input | Key type, owner] -- ***equation (1).***

In this notation, "Function", "Input" and "Output" are mandatory. Other parameters are used to describe a function in more specific terms. Each "Function" has an "Input" and an "Output". The "Function" can be implemented using a particular "algorithm". Most functions are controlled by a "key". A key may have a "type" and also specify its "owner".

"The possible parameters are given as follows:

Function: {E, D, H, S……..}

Algorithm: {RSA, DES, AES, DSA…..}

Type: {public, private, session….}

Where,

"E", "D", "H" and "S" represent encryption, decryption, hash and digital Signatures, respectively. [3]

V. VARIOUS APPROACHES FOR APPLYING THE METHODS

5.1. Approaches for Data Encryption:

"Three approaches in which the encryption must perform are mentioned below:

1. **Encrypt the file using symmetric encryption only:** Only one session key is used and this should be the same key that encrypts the file which is used to decrypt it. This means that there is only one key used for encryption and decryption i.e. the symmetric key. The key is not stored with the encrypted file and so the key needs to be loaded during the process and protected when stored. The whole system is designed by using this approach.
2. **Encrypt the file using a combination of asymmetric and symmetric encryption:** The dual approach requires a symmetric session key to encrypt the data and an asymmetric key to protect the session key. Both the encrypted session key and the encrypted data are stored together in the document. The public asymmetric key is used to encrypt the session key while the private asymmetric key is used to decrypt the key.
3. **Encrypt the file using a X.509 Certificate:** This approach uses a X.509 certificate as the symmetrical key. X.509 certificates are provided by a third party vendor such as VeriSign. [1]

For example: "A general example for encrypting the whole document by using the encryption methods:

The sample XML files to be encrypted:

<purchaseOrder>
<Order>
<Item>book</Item>
<Id>123-958-74598</Id>
<Quantity>12</Quantity>
</Order>
<Payment>
<CardId>123654-8988889-9996874</CardId>
<CardName>visa</CardName>
<ValidDate>12-10-2004</ValidDate>
</Payment>
</purchaseOrder>

The above XML file in encrypted mode:

<?xml version='1.0' encoding="UTF-8"?>
<EncryptedData xmlns='http://www.w3.org/2001/04/xmlenc#' Type='http://www.isi.edu/in-notes/iana/assignments/media-types/text/xml'>
<CipherData>
<CipherValue>A23B45C56</CipherValue>
</CipherData>
</EncryptedData>

This above XML encrypted file shows the cipher text which contains the <CipherData> and the <CipherValue> tags. The actual encrypted data appears as contents of the <CipherValue> tag as shown in the above example. The complete CipherData element appears within an EncryptedData element. The EncryptedData element contains the XML namespace used for encryption. The other attribute, xmlns specifies the XML Encryption namespace that we used to encrypt the XML data." [19]

5.2. Approaches for Data Decryption:

Data Decryption is defined as a W3C standard for decrypting the encrypted file. "Decryption is the process of converting encrypted data back into its original form."[9] This means that to recover the plaintext from the ciphertext. This is also called as a reverse process of encryption. Normally decryption is performed when there is an encryption or vice versa. Only with these two methods, the security fulfills its requirements and through this a secured system can be developed. In this case, the ciphertext is passed through a decryption process as controlled by a decryption key.

The following decryption function is,

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

→ D [ciphertext | decrypt_key] = D [E [plaintext| encrypt_key] | decrypt_key]
D [ciphertext | decrypt_key] = plaintext -- ***equation (3).***
Where,
"D" is the decryption.
"Decrypt_key" is the decryption key. [3]

5.3. Approaches for Data Signatures:

Document signatures are done using digital signatures i.e. designed for use in file transactions also called Dsig and XML-Sig. This method is used to provide data integrity. The file will be signed by computing the message digest of the file and then only the encryption takes place for encrypting the message digest with the sender's private key. The message and the digital signature are then sent to the respective receiver(s) as shown in the following figure.

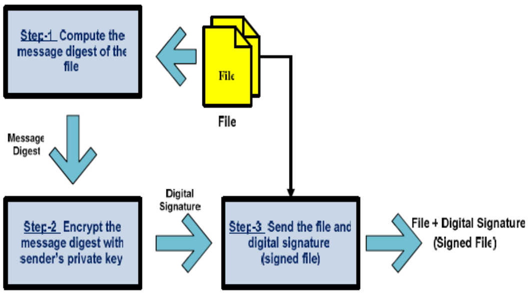
Step-1 Compute the message digest of the file
File
File
Message Digest
Step-2 Encrypt the message digest with sender's private key
Digital Signature
Step-3 Send the file and digital signature (signed file)
File + Digital Signature (Signed File)

Figure: Steps in digital signature generation.

This part meets the requirement of signing portions of documents via Transforms: processing the document before signing for example: XPath etc. It can be used to specify a signature over a list of resources i.e. one or more parts of a document or different document". [3][20]

5.4. Approaches for Data Validation:

"After Data Signatures, the validation is must required. The electronic signatures are also easily verified by using a reverse encryption process i.e. decryption. It is also defined as the validating of the decrypted file which verifies the authentication and data integrity by the receiver. The validation of the file always performs in true or false statement. In the system, the validation always performs on the gateway and works to remove the signed information from the file which results into the original file. For verifying the signature, the same hash function is needed as used by the sender, and the signer's public signature key.

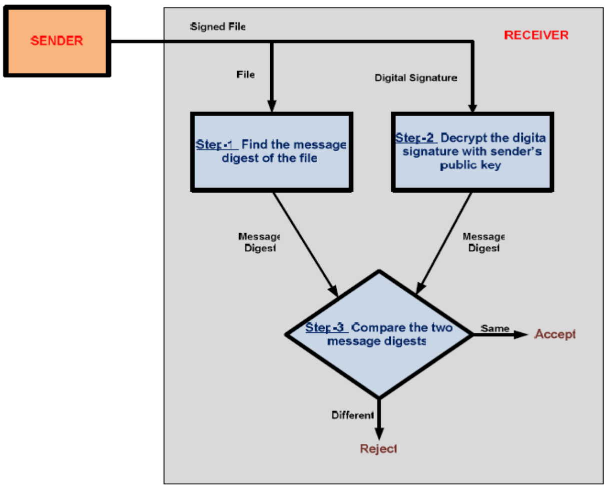
SENDER
Signed File
RECEIVER
File
Digital Signature
Step-1 Find the message digest of the file
Step-2 Decrypt the digita signature with sender's public key
Message Digest
Message Digest
Step-3 Compare the two message digests
Same
Accept
Different
Reject

Figure: Steps for verification of digital signature

From the above figure, the message digest and the digital signature are sent to the receiver for verification. For verifying the digital signatures, the recipient performs the following steps takes place:

Step-1: Firstly, find the message digest of the file.

Step-2: Decrypt the digital signatures with the sender's public key to get the message digest by using the digest algorithm i.e. specified in SignatureMethod element. This step is used for verifying the signature of the SignedInfo element.

Step-3: Finally, recalculate the digest of the reference contained within the SignedInfo element and compare the above two message digest (as mentioned in the Step-1 and Step-2) to the digest values. If the content of the file has not been changed then the both message digests are equal. [3][10]

VI. IMPLEMENTING VARIOUS ALGORITHMS:

"An encryption algorithm performs mathematical operations to conduct substitutions and transformations to the plaintext". [11] Algorithms have a variety of uses in ensuring the integrity of communications. Security is necessary when communicating over any entrusted medium. Many different algorithms are used for encryption, but certain elements are common to all of them. "Algorithms can be divided into classes depending on the technique and approach employed.

_ Symmetric Algorithm (e.g. DES, AES)
_ Asymmetric Algorithms (e.g. RSA)
_ Hash Algorithms (e.g. Message Digest)
_ Key Management [13]

The algorithm *AES (Advanced Encryption Standard)* and *DES (Data Encryption Standard)* are used for encryption whereas the algorithm *RSA (Rivest, Shamir and Adleman)* and *DSS (Digital Signature Standard)* are used for signatures. By using these algorithms for making the proper method calling function for encrypting, signing, decrypting and validating the document respectively.

6.1. AES:

"AES is also known as Rijndael in the cryptography world. This is a symmetric block cipher with variable block and key-length. Symmetric block cipher is used for encrypt (encipher) and decrypt (decipher) information. This algorithm is quite complicated as compared to others because it is very advanced than others. AES is still in competition which as of early 2000. The AES was published by NIST (National Institute of Standards and Technology) in 2001. [14][15] "AES is much faster than triple-DES but may be faster than DES also. The Rijndael proposal for AES defined a cipher in which the block length and key length can be independently specified to be 128, 192 or 256 bits to encrypt and decrypt data in blocks of 128 bits. The AES specification uses the same three key size alternatives but limits the block length to 128 bits. The algorithm AES is basically used for encryption. This algorithm is basically used for protecting the electronic data. The algorithm may be used with three different key lengths as indicated previously.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

Therefore, these different flavors referred to as: “AES-128”, “AES-192”, and “AES-256”.

The specification of this includes the following as mentioned:

I. Definitions of terms, acronyms and algorithm parameters, symbols and functions.

II. Notation and conventions used in the algorithm specification, including the ordering and numbering of bits, bytes and words.

III. Mathematical properties that is useful in understanding the algorithm.

IV. Algorithm specification, covering the key expansion, encryption and decryption routines.

V. Implementation issues, such as key length support, keying restrictions and additional Block/key/round sizes.” [14]

The following are the structure of Document Encryption in which the document to be represented as after encrypting the document by using the algorithm. These are the following information which the encryption consists of:

<enc:EncryptedData Id? Type? MimeType?>
<enc:EncryptionMethod Algorithm/>?
<dsig:KeyInfo>?
<enc:CipherData>
<enc:CipherValue>?
<enc:CipherReference URI?>?
</enc:CipherData>
<enc:EncryptionProperties>?
</enc:EncryptedData>
Xmlns:enc=http://www.w3.org/2001/04/xmlenc# [21]

“The basic structure of the encryption elements are described as follows in details:

_ The EncryptedData element is broken down into a number of child elements with the type of attributes. This element is contained the whole information i.e. in the http://www.w3.org/2001/04/xmlenc#.

_ The EncryptionMethod element is used to specify the symmetric method used when encrypting the data. It does this by using an Algorithm attribute containing a W3 URL that describes the method used.

Forexample:

“http://www.w3.org/2001/04/xmlenc#aes156-cbc" indicates the data was encrypted using **AES (Rijndael)** with a 156k key size.

_ The KeyInfo element is used to store the information about the symmetric keys. This element can also store information about more than one key.

_ The CipherData and CipherValue elements that are found as part of the EncryptedKey and EncryptedData elements contain the cipher data. The actual cipher data is stored in the CipherValue element. The EncryptedKey element stores the encrypted key, while in the encrypted data is stored in the CipherValue for the EncryptedData element.

_ The EncryptedKey element and its child elements contain information about one key stored in a KeyInfo element.

_ The EncryptionMethod element of the KeyInfo contains the asymmetric encryption method which is used to encrypt the session key. It will do by using the Algorithm attribute i.e. set to a W3 URL.

6.2. DES:

“DES” is also known as *DEA (Data Encryption Algorithm)* and basically used for the computer security. DES is the most popular private key encryption method based on research work by IBM. It is used for encrypting and decrypting binary coded information. Encrypting data converts it to an unintelligible form called cipher. Decrypting cipher converts the data back to its original form called plaintext. Given the potential vulnerability of DES to a brute-force attack, there has been considerable interest in finding an alternative. DES was adopted by the government in the United States as an encryption standard in 1977. It was also published in the Federal Register on March 17, 1975. This algorithm is more than 20 years old and due to its too small key-space is now essentially obsolete. The two cryptographic algorithms: Data Encryption Standard (DES) and the Triple Data Encryption Algorithm (TDEA) which may be used by Federal Organizations to protect sensitive data. Protection of data during transmission or while in storage may be necessary to maintain the confidentiality and integrity of the information represented by the data. The algorithms uniquely define the mathematical steps required to transform data into a cryptographic cipher and also to transform the cipher back to the original form. The Data Encryption Standard is being made available for providing total security program consisting of physical security procedures, good information management practices and computer system/network access controls.” [16] “Basically, DES encrypts 64-bit data blocks through many stages of transposition and substitution using a 56-bit encryption key. The following notation explains the DES encryption and decryption respectively as shown below:

→ ciphertext = EDES [plaintext | key]

→ DDES [ciphertext | key] = DDES [EDES [plaintext | key] | key] = plaintext

Where,

“E” stands for encryption and “D” stands for decryption.
“Key” is the encryption/decryption key.

In DES, the same key is used for encryption and decryption. For making the DES more secure, one can use more stages of encryption i.e. a triple-DES method. DES is reasonably easy to implement in either hardware or software. Altogether, DES has fared better than many later symmetric ciphers in terms of resistance to new ideas of cryptanalytic attacks. So, it is built into so many pieces of software. DES is much more complicated than classical ciphers. In typical implementation, instead of using three stages of encryption, it uses two stages of encryption and one stage of decryption. This fundament is explained in the following function:

→ EDES [DDES [EDES [plain_text | key1] | key2] | key1] = cipher_text

Hence, it is also called *DESede where “ede” stands for “encryption, decryption and encryption”*. So, for this case, there are only two keys are used i.e. (key1 and key2). If the two keys are the same then effectively it becomes the single-stage DES. Hence, it allows backward compatibility with the conventional DES.” [3][10]

“The applicability of this standard may be used by Federal departments and agencies when the following conditions apply:

I. An authorized official or manager responsible for data security or the security of any computer system decides that cryptographic protection is required.

II. The data is not classified according to the National Security Act of 1947, as amended, or the Atomic Energy Act of 1954, as amended.” [16]

6.3. DSA:

DSA is also known as *DSS (Digital Signature Standard)*. DSA uses an algorithm that is designed to provide only the digital signature function. Unlike RSA, it cannot be used for encryption or key exchange. Nevertheless, this is a public key algorithm, the secret key operates on the message hash generated by *SHA-1 (Secure Hash Algorithms)* to verify a signature as shown in the figure and on recomputed, it uses the public key to decrypt the signature and then compare the results. This is basically used for computer security and cryptography. The National Institute of Standard and Technology (NIST) have published Federal Information Processing Standard FIPS 186, known as DSS. “The DSA digital signature is a pair of large numbers represented in a computer as strings of binary digits. The DSA provides the capability to generate and verify the signatures. Signature generation makes use of a private key to generate a digital signature whereas signature verification makes use of a public key which corresponds to, but is not the same as, the private key.

Each user possesses a private and public key pair i.e. called as symmetric key. Public keys are assumed to be known to the public in general. Private keys are never shared. Anyone can verify the signature of a user by employing that user's public key. Signature generation can be performed only by the possessor of the user's private key.

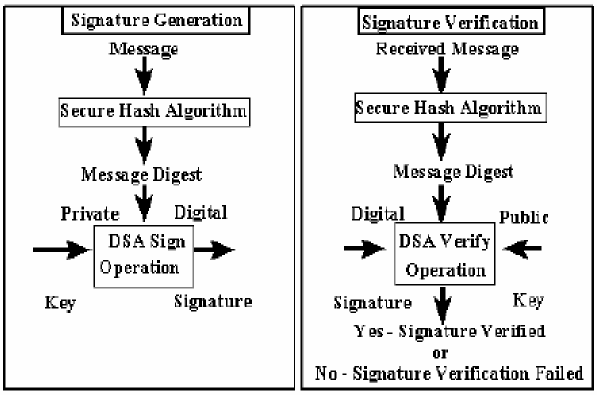

Figure: Signature Generation & Signature Verification with DSA/DSS.

In the above figure, the hash function is used for signature generation and verification. The message digest is then input to the DSA to generate the digital signature. The digital signature is sent to the intended verifier along with the signed data (often called the message). The verifier of the message and signature verifies the signature by using the sender's public key. The same hash function must also be used in the verification process. The hash function is specified in a separate standard, the Secure Hash Standard (SHS), FIPS 180. Similar procedures may be used to generate and verify signatures for stored as well as transmitted data.” [17]

Generate the key pair with the help of DSA (DSS) algorithm and initialize it to 512 bits.

```
KeyPairGenerator kpg = KeyPairGenerator.getInstance
("DSA");
kpg.initialize (512);
KeyPair kp = kpg.generateKeyPair ();
```

6.4. RSA:

“Unlike private key encryption, which is usually based on transposition and substitution techniques, public key encryption is usually based on number theory. This algorithm used for public key cryptography. Its security is based on the difficulty of factoring large integers. RSA is a block cipher, it encrypts message in blocks. It is often used for signature and key exchange purposes because encryption and decryption are slow as it consumes time. One of the first responses to the challenge was developed in 1977 by Ron Rivest, Adi Shamir and Len Adleman at MIT and first published in 1978 [RIVE78]. It was the first algorithm known to be suitable for signing as well as encryption. As ecommerce system relies heavily on the RSA encryption algorithm and is believed to be secure given sufficiently long keys and the use of up-to-date implementations.” [18]

The following are the structure of Document Encryption in which the document to be represented as after encrypting the document by using the algorithm. These are the following information which the encryption consists of:

```
<enc:EncryptedData Id? Type? MimeType?>
<enc:EncryptionMethod Algorithm/>?
<dsig:KeyInfo>?
<enc:CipherData>
<enc:CipherValue>?
<enc:CipherReference URI?>?
</enc:CipherData>
<enc:EncryptionProperties>?
</enc:EncryptedData>
Xmlns:enc=http://www.w3.org/2001/04/xmlenc#
```

Forexample:

“http://www.w3.org/2001/04/xmlenc#rsa-1_5” describes that RSA asymmetric encryption was used to encrypt the session key.

_ The KeyName element is an identifier used to find the key.

_ The CipherReference consists of the URI references with the CipherData.

http://www.w3.org/2001/04/xmlenc# namespace is the root of the encrypted data.”

Use of DSA and RSA algorithm for validating the signature. These two algorithms are used if the signature is true otherwise the statement is false.

```
if (algName.equalsIgnoreCase("DSA") &&
algURI.equalsIgnoreCase(SignatureMethod.DSA_SHA1
)) {
return true;
} else if (algName.equalsIgnoreCase("RSA") &&
algURI.equalsIgnoreCase(SignatureMethod.RSA_SHA1
)) {
return true;
} else {
return false;
}          [22]
```

VII. CONCLUSION:

As Secure transmission becomes a vital component of the emerging electronic business infrastructure, so one must need trustable, secure messages to form the basis of business transactions. So, the design of the system is complete and developed as per requirements. The Security standards define languages and processing rules for meeting common security requirements. This is also using legacy cryptographic and security technologies with some flexible, extensible and practical solutions. These security requirements are Signature/Validations and Encryption/Decryption that can be implemented in a large number of applications and libraries.

The digital signature operations of Sign and Validate are computationally expensive, and more than 30 percent of the CPU time can be spent in doing the actual cryptographic operations. Hence, for this one can use the hardware cryptographic accelerators to meet the demanding performance requirements of cryptographic operations. “Without these all four methods, the cryptographic operation is incomplete and unsecured. Due to these standards, the web service is also implemented successfully with the help of several API’s i.e. used. Security standards will be essential to moving business online as technologies are adopted for Web Services, Digital Rights Management and other emerging applications.

Security have met the authentication, authorization, confidentiality, integrity, signature and privacy requirements for a making a secured system. The system goals are achieved completely up-to-date. The whole system is used to present a brief introduction to the set of standards and how they work together.

REFERENCES

1. Haas, H., Seminar, “XML Security: Signature, Encryption, and Key Management”, December 2005. http://www.w3.org/2004/Talks/0520-hhxmlsec/
2. Organization for the Advancement of Structured Information Standards (OASIS), Web Services Security: SOAP Message Security 1.0. (2004).
3. Chi Chester: Wiley, “E-commerce: fundamentals and applications”, [Henry Chanetal.], pp 203-217, (2001).
4. “Cryptography and Network Security” by- William Stallings.
5. Wikipedia, the Free Encyclopedia, “WS-Security”, no. Wikimedia Foundation, Inc. (2006). http://en.wikipedia.org/wiki/Web_Services_Security
6. W3C, “Decryption Transform for XML Signature”, W3C Candidate Recommendation: March 04, 2002. http://www.w3.org/TR/2002/CR-xmlenc-decrypt-20020304
7. ZipPay MasterCard® Gift Card is issued by MetaBank pursuant to license by MasterCard International Incorporated, Columbus Data services, (2007). http://www.zippaycard.com/terms.htm
8. The Apache XML Project, “Welcome to XML Security”, Apache XML Security, published on September, 2007. http://santuario.apache.org/
9. University of West Indies at Cave Hill, “Decryption/Encryption”, published on 2003. http://scitec.uwichill.edu.bb/cmp/online/ CS22K/security/decrypt_encrypt.htm
10. Santa Barbara, Calif., “Advances in cryptology - CRYPTO '89: proceedings”, CRYPTO (Conference-1989), published by Springer, XII, 634 p. (1990).
11. “Private key (Symmetric) Encryption”, MyCrypto.net, (2008).
12. Garrett, Paul B., “Making, breaking codes: an introduction to cryptography”, published by Prentice Hall, XIX, 524 p., (2001).
13. “Encryption Algorithms”, CES Setting the Standard, (2003). http://www.cescomm.co.nz/about/algorithms.html
14. FIPS PUB 197, Federal Information Processing Standards Publication, “Advance Encryption Standard (AES)”, U.S. Department of Commerce/National Institute of Standards and Technology, dated on November 2001. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
15. Wikipedia, the Free Encyclopedia, “Advanced Encryption Standard (AES)”, no. Wikimedia Foundation, Inc. (2008). http://en.wikipedia.org/wiki/Advanced_Encryption_Standard
16. FIPS PUB 46-3, Federal Information Processing Standards Publication, “Data Encryption Standard (DES)”, U.S. Department of Commerce/National Institute of Standards and Technology, dated on October 1999. http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf
17. FIPS PUB 186, Federal Information Processing Standards Publication 186, announcing the Standard for, “Digital Signature Standard (DSS)”, dated on May 1994. http://www.itl.nist.gov/fipspubs/fip186.htm
18. Wikipedia, the Free Encyclopedia, “RSA”, no. Wikimedia Foundation, Inc. (2008). http://en.wikipedia.org/wiki/RSA
19. Bilal Siddiqui, CEO, WAP Monster, “Exploring XML Encryption Part-1”, IBM, Demonstrating the secure exchange of structured data, published on March 2002. http://www.ibm.com/developerworks/xml/library/x-encrypt/#code1
20. W3C, “XML Signature Features”, W3C Recommendation, (2002). http://www.w3.org/2004/Talks/0520-hh-xmlsec/ slide6-0.html
21. Robert Richards, “XML Security”, published on September 2006. http://66.102.9.104/search?q=cache:KqMFAXAetVcJ:cdatazone.org/talks/php_works_2006/xml_security.ppt+XML+Security&hl=en&ct=clnk&cd=5&client=firefoxa
22. Sean Mullan, “Programming with the Java XML Digital Signature API”, Sun Microsystems, Sun Developer Network (SDN), March 2007.http://java.sun.com/developer/technicalArticles/xml/dig_signature_api/

AUTHOR PROFILE

Rajat Chaudhary, I am Rajat Chaudhary, age 24 years born on April 09, 1988 in Meerut. My hometown is Meerut and I have done B.Tech. From Vidya College of Engineering, Meerut affiliated from U.P.T.U. in 2010 batch. After that I have taken admission in Dehradun Institute of Technology, Dehradun affiliated from Uttrakhand Technical University for doing my M.Tech. in Information Security Management in 2010-2012. It is last semester over there and it is going to complete till August 2012. Just now I am teaching in Millennium Institute of Technology which is at Dehradun-Roorkee Highway, NH-72 and is affiliated from GBTU/MMTU.

My areas of interest are Computer Network, Cryptography and Network Security, Distributed System and Mobile Computing. I have done thesis on “Secure Xml Document Encryption” from where I have idea on this research paper. This research paper is very interesting because the concept of cryptography and security of information plays a major role. In this paper I have done how to use algorithm for using Symmetric and Asymmetric keys. My friends Prem and Ambika have a great contribution in this paper they guided me how to use Algorithms and methods for confidentiality and authentication of documents and I am very grateful for them.

Ambika Agarwal, 23 years of age belongs to distt. Pauri (Uttarakhand). I have done my schooling from St. Mary’s convent school and St. Bede’s college, Himachal Pradesh. I completed my bachelor’s degree in Information Technology from RIT, Roorkee (UTU) in the year 2010. Then I went over to pursue my master’s in Information Security & Management from DIT, Dehradun also affiliated to UTU. Currently I am also working as a lecturer in Uttaranchal Institute of Technology, Dehradun (UTU) for the past one year. My areas of interest are Cryptography, Networking and Operating System.

Prem Singh, born on July 05, 1987. I have done my B.Sc. and M.Sc. (I.T.) from Soban Singh Jeena Campus, Almora which is affiliated from Kumaun University, Nainital. After that I have taken admission in Dehradun Institute of Technology, Dehradun affiliated from Uttrakhand Technical University for doing my M.Tech. in Information Security and Management in 2010-2012 batch. It is my last semester over there and it is going to complete till August 2012. The idea of this research paper comes on my friend and my batch-mate Rajat Chaudhary’s mind. He discuss with me to work with him on this topic. After that we (Rajat Chaudhary, me and Ambika Agarwal) work together on this topic and the result of this comes in front of you as a research paper.