

Survey on XML Encryption

Saurabh Kumar Sah



Department of Computer Science and Engineering

National Institute of Technology Rourkela

Rourkela-769 008, Odisha, India

June 2014

Survey on XML Encryption

Thesis submitted in partial fulfillment of the requirements for the degree of

Master of Technology

in

Computer Science and Engineering

(Specialization: Software Engineering)

by

Saurabh Kumar Sah

(Roll No.- 212CS3377)

under the supervision of

Dr. Sujata Mohanty



Department of Computer Science and Engineering

National Institute of Technology Rourkela

Rourkela, Odisha, 769 008, India

June 2014



Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India.

Certificate

This is to certify that the work in the thesis entitled *Survey on XML Encryption* by *Saurabh Kumar Sah* is a record of an unique research work completed by him under my supervision and direction in halfway satisfaction of the necessities for the honor of the degree of Master of Technology with the specialization of Software Engineering in the department of Computer Science and Engineering, National Institute of Technology Rourkela. Not this thesis or any some piece of it has been submitted for any degree or scholarly honor somewhere else.

Place: NIT Rourkela
Date: June 2, 2014

(**Dr. Sujata Mohanty**)
Assistant Professor, CSE Department
NIT Rourkela, Odisha

Author's Declaration

I hereby declare that all work contained in this report is my own work unless otherwise acknowledged. Also, all of my work has not been submitted for any academic degree. All sources of quoted information has been acknowledged by means of appropriate reference.

Saurabh Kumar Sah

Roll No: 212CS3377

Department of Computer Science & Engineering

Acknowledgment

I am appreciative to various nearby and worldwide associates who have helped towards forming this thesis. At the start, I might want to express my earnest on account of Dr. Sujata Mohanty for her recommendation throughout my proposal work. As my supervisor, she has always swayed me to stay concentrated on attaining my objective. Her perceptions and remarks helped me to build the general heading to the exploration and to advance with examination in profundity. She has helped me enormously and been a wellspring of information.

I am really obligated to Prof. Santanu Ku. Rath, Head-CSE, for his consistent support and backing. He is constantly primed to help with a grin. I am additionally appreciative to all the teachers at the division for their backing.

I might want to thank all my companions and lab-mates for their consolation and comprehension. Their help can never be penned with words.

I must recognize the scholarly assets that I have got from NIT Rourkela. I might want to thank regulatory and specialized staff parts of the Department who have been caring enough to exhort and help in their particular parts.

Last, however not the minimum, I might want to devote this thesis to my family, for their affection, persistence, and comprehension.

Saurabh kumar Sah

Roll-212cs3377

Abstract

Every transaction on the Internet involves some kind of data. Data can be transferred in various modes. Now a days, XML is widely used for transferring and storing the data. There must be some mechanism to protect these data. In most of the literature, two most important techniques i.e. XML Signature and XML Encryption are used for securing these XML data. These two techniques provide signing and encrypting of XML data using cryptographic functionalities and results are also represented in XML format. These two techniques are considered as standard worldwide which is released by W3C. In this thesis we are focusing on XML Encryption.

In this study, W3C standards are used to encrypt sensitive XML data. JavaScript has been used to implement encryption of XML data and "Node.js" as software platform for providing the environment for encrypting. In this study, time elapsed is also measured in case of encryption and decryption. We have used AES and Triple DES algorithm for encryption of XML data. For encryption of symmetric key, RSA is used. Library used is "xml-encryption" for encryption and decryption. Time analysis for encryption and decryption are also shown by graph.

Keywords: XML, XML Encryption, XML Parser, DOM etc.

Contents

Certificate	ii
Acknowledgement	iii
Abstract	v
List of Figures	viii
List of Tables	ix
1 Introduction	1
1.1 Thesis Organization	6
2 Theoretical Background	7
2.1 XML	7
2.2 Basic Cryptography Concepts	8
2.3 XML Encryption [1]	10
2.4 Encryption Granularity	11
2.4.1 Encrypting an XML Element	11
2.4.2 Encrypting XML Element Content (Elements)	12
2.4.3 Encrypting XML Element Content (Character Data)	12
2.4.4 Encrypting Arbitrary Data and XML Documents	12
2.5 Processing Rules	14
2.6 XML Parser or API	14
2.6.1 DOM(Document Object Model) API	14
3 Literature Review	16
3.1 Related Work	16
3.2 Motivation	22

3.3 Objective	22
4 Implementation and Results	23
4.1 Introduction	23
4.2 Implementation	23
4.3 Results	23
5 Conclusion and Future Work	27
Bibliography	28

List of Figures

2.1	Simple XML Example	8
2.2	Structure of an XML Encrypted file	11
2.3	XML Element Encryption	12
2.4	XML Element Content Encryption	12
2.5	Character Data Encryption	13
2.6	Encrypting entire XML document	13
2.7	Transformation to DOM tree	15
4.1	Input XML File	24
4.2	Encryption Result	24
4.3	Encryption Result	25
4.4	Decryption output	25
4.5	Time Analysis of Encryption and Decryption	26

List of Tables

Chapter 1

Introduction

Present techniques in the field of security are not perfect because it does not provide enough high level security and flexibility in securing business data on the web. So there must be some technique which can fulfill our present goal so the system become more secure and flexible. In business transactions there are so much sensitive data used daily. Thats why security of these data is very essential.

For instance, Secure Sockets Layer (SSL) provide secure exchange of important data between a Web server and browser, but the problem with this technique is that, upon reaching the data on the server side it becomes vulnerable because it does not provides security on the server [2]. That is it provides security in the transit only. Therefore we cant use this technique in most business transaction because it lefts date or information unprotected at the server side. If the data or information is encrypted and then transmitted then there is very low chance of exploiting the data or information by attacker or third party on open servers.

So securing the confidential sensitive information is nothing but ensuring their non-repudiation, integrity and authenticity. The largely used method for providing these requirements for proper transmission of data or information is to use digital certificates to apply the digital signing and encryption of the those useful data. The Public Key Infrastructure(PKI) provides the standards and policies which can be applied to sensitive data for signing and encrypting with the help of public key, private key and certificates generated.

In the current scenario XML is largely applied for data or information transfer and storing of data in the Internet. The most important quality which make XML

very powerful for sensitive business transactions is because XML is structured, semantically rich, text-based and of Web-prepared nature. Therefore it gives both chance and dispute for the application of digital encryption and signature to XML information or data. For example, in many conditions where an XML information through various steps between users, and at that time a digital signature points some kind of averment or commitment, each party may wish to sign a particular part for which they are accountable and ponder a attendant level of responsibility. ordinary standards for digital signatures do not provide syntax for getting this provision of high level of flexibility for signature and also do not give privilege for conveying which part a sender want to sign.

There are mainly two new security techniques which are widely used to secure XML data and take edge of the exceptional feature of XML are XML Encryption and XML Signature. These two standards are presently going through the institutionalization operation. XML Signature is the combined effort of the Internet Engineering Task Force (IETF) and World Wide Web Consortium (W3C) and XML Encryption is completely a W3C effort.

Public-Key-Cryptography technique grant users of the computer network, like the Internet, to exchange data or information with surety that it will not be modified and also not improperly accessed. This mechanism can be acquired by changing the information according to an algorithm parameterized by a combination of two numbers, called private and public keys. Each number in the context has these two keys. They keep public key available to all who wants to communicate with him/her and they keep the another key safe private. Apart from this fact that the keys are mathematically connected, if the cryptography system has been made and executed safely, it is mathematically not possible to find the private key from public key and vice versa.

The nature of the connectivity between these two keys in such a way that a cryptographic alteration protected by one key can only be get backed with the another key. This very quality of encryption using public key technique provides security because an information or data encrypted with the public key of a partic-

ular user can be decrypted only by the one who is the owner of the corresponding private key (i.e., the recipient, if they have nicely secured access to the private key). Even though the data is hacked by someone else with the private key they cant read those data because of its encrypted format. It is very important that the privacy can be granted without the use of any secret data with the original message which is the case of symmetric cryptography. As in this case the key is exchanged between sender and receiver which may cause problem when number of users is very high. Thats why public key cryptography has come.

We know that confidentiality is the first quality of cryptography which comes generally into the brain, the relation between private and public key also gives the function to system such that no secret key cryptography is required. It provides authentication, non repudiation and integrity very well. It is very much analogous to the paper world signatures. Since it involves digital data thats why called Digital signature.

To make an advanced signature for any information, the data to be marked is changed by a calculation that uses as enter the private key of that specific sender. Since a change made by the sender's private key must be turned around back if the opposite convert takes as a parameter the sender's open key, a receiver of the interpreted information could make certain of the wellspring of the information(the distinguished of the sender). On the off chance that the data could be confirmed utilizing the sender's open key, then it must be marked utilizing the related private key (to which just the sender ought to have entrance).

For the mark confirmation to be capable, the verifier must have surety that people in general key truly fit in with the sender (overall an actor could confirm to be the sender, dispensing her open enter set up of the real one). A testament, supplied by a Certification-Authority, is an affirmation of the pass-ability of the affiliation between the confirmation's subject and his/her open key such that distinctive clients could verify that open key does undoubtedly identify with the subject who attests it as her own.

Broadly because of the execution conduct of open key calculations, the com-

plete data or information is predominantly not itself deciphered straight with the private key. Rather a little unique part of the data or information, called a condensation or hash quality is changed. Since the hashing calculation is really touchy to any alteration in the source data, the hash of the first allows a recipient to confirm that the data was not changed (by matching the hash or process that was sent to them with the hash or overview they found after count from the data they got). Besides, by deciphering the hash or overview with their private key, the sender likewise allows the recipient to validate that it was really the sender that done the changing (on the grounds that the beneficiary could utilize the sender's open key to switch back the conversion). The hash or overview of the data or information, interpreted with the sender's private key, in this manner demonstrations as a computerized signature for that specific data or information and could be transmitted freely alongside the information to the beneficiary. The collector affirms the signature by getting to the hash of the message and providing for it as info to a verifying calculation together with the mark that included the message and the sender's open key. In the event that the yield is fruitful, the beneficiary might make certain of both the terms that is, uprightness and legitimacy of the data or delicate information.

XML marks are only computerized marks utilized for applying within XML information transactions. This system gives a pattern to recognizing the consequence of an advanced mark performed identified with self-spellbinding (really as often as possible XML) data. Similar to non-XML information advanced marks (e.g. PKCS), XML marks gives information respectability, validation and bear for non revocation to the data or information that they will sign. Regardless, not like non-XML advanced mark guidelines, XML digital signature has been utilized for both record for and make utilization of XML and the Internet.

A basic nature of XML Signature is the quality to sign simply particular portions of the XML archive instead of the whole record. This is suitable if a solitary XML report may have a long data in which the assorted parts are composed at different times by diverse clients, every one marking simply those segments suitable

to itself. This flexibility will in like manner be fundamental in conditions where it is key to assurance the trustworthiness of particular parcels of a XML record, while leaving open the probability for diverse parts of the record to change. Case in point, a marked XML structure passed on to a customer for satisfaction. On the off chance that the mark is for the full XML archive, any change by the client to the default structure qualities might not accept the genuine mark.

A XML mark might be utilized to sign more than one sort of asset. For example, a solitary XML mark may blanket twofold encoded data (a JPG), character-encoded data (HTML), XML-encoded data, and a specific region of a XML record.

Approval of mark obliges that the information or data that was marked be open. The XML mark demonstrate that what information has been marked and it may be demonstrated by a reference. This reference might be referenced by an Uniform asset identifier(uri) inside the XML mark; have its XML mark settled inside itself (the mark is the youngster component); be altered inside the XML signature (the mark is the guardian component); stays inside the same asset as the XML signature (the mark is the kin).

The XML Encryption standard characterizes how to encode XML reports. Encryption comes commonly in two sorts:

Symmetric or secret key encryption:- In this sort of encryption the sender and the collector imparts the single mystery key which is utilized to both scramble and decode information or data.

Asymmetric or public key encryption:- Here the recipient issues his open key which permits any sender to encode an information or data. Just the recipient can decode the message utilizing his/her private key. The primary characteristic of symmetric key encryption is that it is quick. Its primary instrument is that both participant(sender and beneficiary) need to concur first on a mystery key which may be troublesome in practice (telephone, paper...): symmetric key encryption does not deal with the key dissemination issue.

The most vital profit of utilizing open key calculations is that it tackles the key appropriation issue which happen in mystery key cryptography.

1.1 Thesis Organization

The remaining part of the thesis is organized as follows.

In **Chapter-2**, Theoretical background of XML Security, particularly XML Encryption is given and the related points are explained in detail.

Chapter-3 contains review of various literatures of related papers, their drawbacks, motivation and objective of our idea. In **Chapter-4** Implementation in **”Javascript”** for XML Encryption using **”Node.js”** as software platform and the library **”xml-encryption”** is used. For Encryption AES and 3DES is used. For encryption of symmetric key RSA is used. Results are also shown for our implementation.

Chapter 2

Theoretical Background

2.1 XML

The acronym XML remains for Extensible Markup Language. It is a W3c proposal since 1998 [3]. W3c is a group of data engineering masters. This association distributes specialized details and proposals to guarantee a long haul development of the World Wide Web. XML was intended to transport and store information. Initially, it ought to be comprehensible to some degree for debugging and other managerial work. A XML report comprises of markup and character information. Markup is characterized as all labels, references, assertions, segments, and remarks. Character information is all content that is not markup. There are a few manages how markup components could be utilized, which brings about strict, tree-organized records. A basic XML case is demonstrated in Figure 2.1. The center idea is the XML component that comprises of a begin label, an end-label and substance. Labels must be overall framed, i.e. for each one begin label must exist an end-tag with the same name in the report. Generally, a parser might toss a special case. All labels must be settled rightly. That methods all end-labels must be shut in the inverse request than the begin labels. Each XML report has a header and a body. The header details that the content record is a XML archive. Figure 2.1 is a basic XML archive and its header is the first line `¡?xml version=1.0?¿`. Close to the rendition quality, it can additionally hold qualities like encoding or standalone to signalize the XML parser how to decipher the body's substance effectively.

```
1 <?xml version='1.0'?>
2 <PaymentInfo xmlns='http://example.org/paymentv2'>
3   <Name>John Smith</Name>
4   <CreditCard Limit='5,000' Currency='USD'>
5     <Number>4019 2445 0277 5567</Number>
6     <Issuer>Example Bank</Issuer>
7     <Expiration>04/02</Expiration>
8   </CreditCard>
9 </PaymentInfo>
```

Figure 2.1: Simple XML Example

2.2 Basic Cryptography Concepts

Cryptography is a critical some piece of keeping private information from being hacked. If an assaulter were to exploit into your workstation or capture your messages in any case they won't have the capacity to peruse the information in the event that it is ensured by cryptography or encoded. notwithstanding disguising the importance of information, cryptography performs other discriminating security prerequisites for information including confirmation, disavowal, classifieds, and honesty. Cryptography could be utilized to confirm that the sender of a message is the real sender and not a faker. Encryption additionally accommodates disavowal, which is like verification, and is utilized to demonstrate that somebody really communicated something specific or accomplished an activity. , for example it can used to demonstrate a criminal accomplished a particular budgetary transaction. Cryptography guarantees secrecy on the grounds that just a spectator with the right translating calculation or key can be read the encoded information or message. Ultimately, Cryptography can ensure the respectability of data by guaranteeing that information messages have never been adjusted. Cryptography originates from the Greek words signifying "concealed composition". Cryptography changes over decipherable information or text into encoded information called cipher text. By definition cryptography is the exploration of concealing data so that unapproved clients can't read it. Mystery composing is an aged practice that

goes once more to antiquated Egypt yet it is till basic to securing information today. indeed, encryption is totally important when sending delicate information over unprotected environment like the Internet. The three basic sorts of calculations utilized for encryption are:

- Hashing
- Symmetric, likewise also known as private or mystery key
- Asymmetric, additionally also knows as open key.

A hashing figuring is utilized to make an irreversible code of a bit of data. This hashed code is known as a hash or chart and is exceptional to the data and may be utilized as a signature for the information. A hash is utilized for correspondence purposes to check information has not been changed; in this manner it guarantees the respectability of a message. A symmetric cryptographic reckoning could be unscrambled, rather than being irreversible like hashing. There are a few sorts of symmetric tallies. Irrefutably the most well known are:

- Data Encryption Standard (DES)
- Advanced Encryption Standard (AES) etc.

DES was one of the first extensively utilized figurings at any rate it has been part and is no more perceived secure. AES has not been part and is utilized by the US government while IDEA is supported by European countries. RC stays for "Ron's Code" and is a social occasion of calculations made by Ron Rivest in 1987. Blowfish is a solid open-source symmetric reckoning made in 1993. Uneven cryptographic processing shift from symmetric estimations in that it obliges two "keys" to encode and unscramble information rather than the symmetric number's single key.

Hilter kilter or open key encryption utilizes two numerically related keys: an open key known by everybody to encode messages and a private key, known basically by the beneficiary of the message to decipher the data. Disproportionate

cryptography is generally utilized and underlies Transport Layer Security (TLS) and PGP (Pretty Good Privacy) get-together. Some regular uneven estimations are RSA and Diffie-Hellman.

Digital Signature and Encryption provides the following security functionalities:

Authentication: This mechanism provides the identity of a sender. The intended receiver upon getting the message authenticate it that it came from a source from which it is required. **Privacy/confidentiality:** It provides the function that the data or information is not read or altered by anybody in between sender and receiver.

Integrity: Confirming the receiver that the received message has not been adjusted at all from the first.

Non-repudiation:This determines the truthfulness of the sender of sending data to the receiver.

2.3 XML Encryption [1]

XML Encryption is the process of Encrypting mainly XML data. It provides a syntax for encrypted XML data to represent in XML format. It is a W3C recommendation since 2002. It also provide the flexibility of encrypting specific portion of XML document viz. XML element, element content, character data or the complete XML document. However it can be used to encrypt any type of data but generally XML data is encrypted. The encrypted portion is replaced by an EncryptedData element with associating child elements. Today XML encryption is must where XML is frequently used for information transmission and storage purposes. Below figure shows the structure of an XML encrypted document. Figure 2.2 shows structure of an Encrypted XML document. In the figure, EncryptedData is the root element. Here "?" denotes zero ore more occurrences, "+" indicates one or more occurrences, "—" denotes a choice and "*" denotes zero or more occurrences. There is EncryptioMethod element which contains the algorithm used for encrypting which is identified by a URI. KeyInfo element contains the key used for

```
<EncryptedData Id? Type? MimeType? Encoding?>
  <EncryptionMethod/?>
  <ds:KeyInfo>
    <EncryptedKey/?>
    <AgreementMethod/?>
    <ds:KeyName/?>
    <ds:RetrievalMethod/?>
    <ds:*/?>
  </ds:KeyInfo/?>
  <CipherData>
    <CipherValue | <CipherReference URI?>
  </CipherData>
  <EncryptionProperties/?>
</EncryptedData>
```

Figure 2.2: Structure of an XML Encrypted file

encrypting which further may be encrypted. It contains AgreementMethod, Key-Name and RetrievalMethod elements. Inside the EncryptedData element there is cypherData element which contains the CipherValue or CipherReference element. CipherValue element contains the base64 encoded value of encrypted data and CipherReference provides a URI where CipherValue is stored. Apart from these elements there may be optional elements like EncryptionProperties etc.

2.4 Encryption Granularity

Encryption granularity provides the mechanism for Encrypting desirable parts of an XML document according to demand of application. There are five encryption granularities..

2.4.1 Encrypting an XML Element

Suppose we want to encrypt the Creditcard element of the XML document shown in figure 2.1. After encrypting it this element get replaced by EncryptedData element. Encrypted document look like figure 2.3 after the CreditCard element is encrypted.

```
<?xml version="1.0"?>

<PaymentInfo xmlns="http://example.org/paymentv2">
  <Name>John Smith</Name>
  <EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element"
    xmlns="http://www.w3.org/2001/04/xmlenc#">
    <CipherData>
      <CipherValue>A23B45C56</CipherValue>
    </CipherData>
  </EncryptedData>
</PaymentInfo>
```

Figure 2.3: XML Element Encryption

```
<?xml version="1.0"?>

<PaymentInfo xmlns="http://example.org/paymentv2">
  <Name>John Smith</Name>
  <CreditCard Limit="5,000" Currency="USD">
    <EncryptedData xmlns="http://www.w3.org/2001/04/xmlenc#"
      Type="http://www.w3.org/2001/04/xmlenc#Content">
      <CipherData>
        <CipherValue>A23B45C56</CipherValue>
      </CipherData>
    </EncryptedData>
  </CreditCard>
</PaymentInfo>
```

Figure 2.4: XML Element Content Encryption

2.4.2 Encrypting XML Element Content (Elements)

Suppose we want to encrypt the elements within the Creditcard element then it comes under this type. Figure 2.4 shows how it looks after encryption.

2.4.3 Encrypting XML Element Content (Character Data)

Suppose we want to encrypt XML element content which is character data. So only that character data will be replaced by EncryptedData element. Figure 2.5 shows how it looks after encrypting account number.

2.4.4 Encrypting Arbitrary Data and XML Documents

Figure 2.6 shows how an arbitrary data or XML file can be encrypted. Within

```
<?xml version="1.0"?>
<PaymentInfo xmlns="http://example.org/paymentv2">
  <Name>John Smith</Name>
  <CreditCard Limit="5,000" Currency="USD">
    <Number>
      <EncryptedData xmlns="http://www.w3.org/2001/04/xmlenc#"
                    Type="http://www.w3.org/2001/04/xmlenc#Content">
        <CipherData>
          <CipherValue>A23B45C56</CipherValue>
        </CipherData>
      </EncryptedData>
    </Number>
    <Issuer>Example Bank</Issuer>
    <Expiration>04/02</Expiration>
  </CreditCard>
</PaymentInfo>
```

Figure 2.5: Character Data Encryption

```
<?xml version="1.0"?>
<EncryptedData xmlns="http://www.w3.org/2001/04/xmlenc#"
              MimeTypes="text/xml">
  <CipherData>
    <CipherValue>A23B45C56</CipherValue>
  </CipherData>
</EncryptedData>
```

Figure 2.6: Encrypting entire XML document

the EncryptedData element, the Key resides which may be encrypted. If key is encrypted then it is contained within the EncryptedKey element. Encrypting EncryptedData is called Super Encryption.

2.5 Processing Rules

There are two implementations viz, Encryptor and Decryptor. The role of Encryptor is to Encrypt the data and the role of decryptor is to decrypt that data. There are different types of algorithms used in the process of encryption and decryption like, Block encryption algorithm, key agreement algorithm, message digest, encoding, canonicalization etc.

2.6 XML Parser or API

The role of XML parser is to check the XML document syntactically and semantically and producing the compatible format to the intended application or interface.

2.6.1 DOM(Document Object Model) API

The concept of this API is platform and language independent interface which provides how to represent and interact with documents like XML, HTML etc. This API provides dynamic access to the document and it can dynamically update the style and content of the XML document. A DOM API converts the XML document into a DOM tree first where each element and data in the document is represented by a node. The DOM tree is saved into the memory. Figure 2.7 shows the transformation.

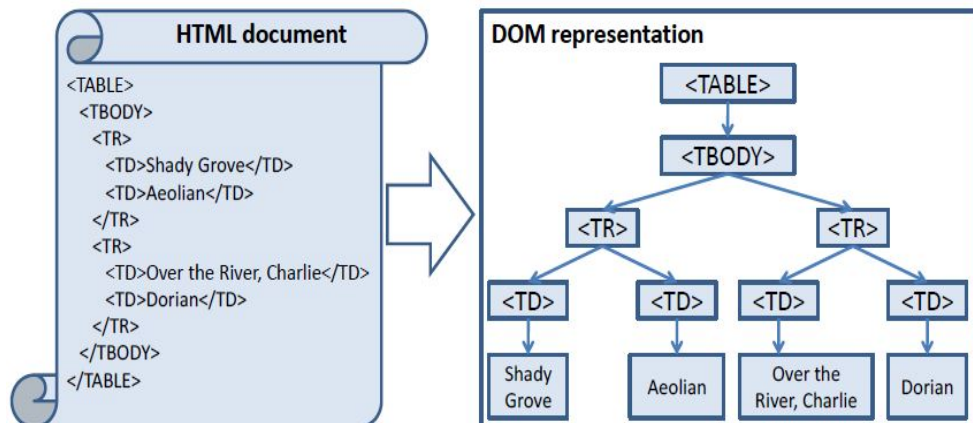


Figure 2.7: Transformation to DOM tree

Chapter 3

Literature Review

3.1 Related Work

Konstantin Beznosov **et.al.** depicted about the web administrations and their security [4] . In this paper, brief outline of the key advances in the territory of web administrations and related security innovations are illustrated. It likewise portray about web administrations building pieces i.e., SOAP, UDDI, and WSDL. Principle consideration of their methodology is on innovations for ensuring SOAP message and conveying security-pertinent data with web administrations, XML security, WS-security and SAML.

Takeshi Imamura **et.al.** demonstrated about the stream-based usage of XML Encryption [5]. They utilized API (Xerces Native Interface) to model a stream based execution of the determination. They additionally assess its execution and contrasted and DOM-Related Implementation. At last it attained a 0.27 % - 26 % decrease in handling time elapsed in encryption of XML records of sizes bigger than 2kb, and 34 % - 88 % for decoding of XML reports with any sizes.

K. Komathy **et.al.** demonstrated the Component based methodology for Securing XML informing administrations [6]. They utilized this model for Extensible Markup Language (XML) arrangement to speak to web transactions, in view of the multi-stage nature of advanced web requisitions. The proposed model joins both computerized mark and encryption to give high security against inactive and dynamic strike. In their approach, remarkable session key is utilized to limit the replay strike. This paper likewise highlights the execution of the proposed model

for charge card approval in a nature.

J. Efrim Boritz **et.al.** talked about the Security in XML-built money related reporting administrations in light of the Internet [7]. This paper addresses about the security in money related reporting administrations. Initially, it portrays about web benefits and conceptualizes fiscal reporting administrations, for example, XBRL and XARL as web administrations. It additionally talked about the security dangers and their limits of current security innovations. At that point, it distinguishes security necessities that ought to be recognized to guarantee solid, dependable XBRL and XARL administrations. At last, the paper demonstrates a few stated security measures and recommends Web Services Security Architecture as a proper security instrument for monetary reporting administrations.

Gwan-Hwan Hwang **et.al.** demonstrated about the operational model and dialect help for securing XML records [8]. In this paper, they show an operational model for XML record security. For a XML record X, they characterized the operational model as the methodology of encoding information and inserting advanced marks, which sign the information in X. The secured XML archive x_s incorporates encoded and decoded information of X, and installed advanced marks. Their operational model additionally characterizes the procedures of unscrambling x_s and checking the advanced marks inserted in x_s . Their proposed model additionally offer a security system which coordinates component insightful encryption and transient based component shrewd computerized marks. Here they characterized another dialect called **document security dialect (Dsl)**. For computerization reasons, the DSL incorporates a definition for the "standard DSL calculation downloading and interfacing convention" which satisfies programmed calculation download and connecting prerequisites in the operational model. Two separate executions further show its practicability: one uses the Java programming dialect to execute the securing apparatus, whilst alternate utilizes the development instrument of XSLT 1.0 to actualize the encryption and unscrambling changes. What's more, they created a DSL manager with an amicable realistic client interface to make it simpler for clients to create DSL reports.

Tao-Ku Chang **et.al.** have given the thought regarding the outline and execution of a provision program interface for securing XML archives [9]. In this paper, they utilize some true cases to exhibit that, it is important to plan a proper API for protecting arrangement of child tree encryption for XML archives. Their objective is to expand gainfulness and decrease the expense of keeping up this sort of programming, for which they propose an archive security dialect (DSL) API. They likewise depict the usage of the DSL API, and use exploratory results to show its reasonableness.

David C. Yen **et.al.** have given the effect and usage of XML on business-to-business trade [10]. This paper demonstrate the effect investigation of the Extensible Markup Language (XML). In this paper, they additionally highlights on how business accomplice inside a store network will be permitted to produce its information trade design by receiving a XML meta-information administration framework in the nearby side. Trailed a concise presentation of the data innovation for Business to Business (B2b) and Business to Customer (B2c) Electronic Commerce (EC), the effect of XML on the future business world is talked about.

Juha-Miikka Nurmilaakso **et.al.** depicted about XML-based e-business structures and institutionalization [11]. This paper examines the properties and institutionalization of 12 noticeable XML based e-business schema's. Their dissection concentrates on the shared characteristics, contrasts and regularities among these e-business schemas and their institutionalization.

Alexandros Kaliontzoglou **et.al.** have given A protected e-Government stage structural planning for little to medium measured open associations [12]. They take Small to medium measured open associations (Smpos) impart some of their e-Government prerequisites with their

bigger partners, for example, the pending requirements for interoperability, security and ease of use. Moreover, they have some particular needs that are either remarkable in their setting or all the more requesting because of their aspects. These are expense and assets contemplations, improved availability and more amazing adaptability because of the bigger number of natives and organiza-

tions served and robotized handling in light of the confined number of prepared faculty. This paper at first proposes a structural planning for a safe e-Government stage focused around Web Services, which provides the above necessities. Also, a particular administration is based upon the proposed stage, in which a region produces and safely conveys an advanced conception testament to a resident or an alternate district.

Paul Kearney depicted about Message level security for web administrations [13]. This paper gives a rundown of the rising accord on security for communitarian business utilizing web benefits as a part of a nature's turf. The most widely recognized security measure utilizing security at transport layer may be enough for straightforward requisitions. Nonetheless, for more intricate situations, e.g. more than two gatherings, or different web administrations, entire messages or distinctive portions of messages may be scrambled and marked to secure the privacy and trustworthiness of web administration messages.

Hye-Kyeong Ko **et.al.** have concentrated on the effectiveness of secure XML television [14]. In this paper, a marking plan is proposed to help quick recreation of XML reports in the connection of a well-known strategy, called XML pool encryption. The proposed marking plan backs the expedient derivation of structure data in all allotments of the archive. The double depiction of the stated marking plan is additionally examined. In the test comes about, the proposed marking plan is proficient in hunting down the area of unscrambled data.

Carlos Gutierrez **et.al.** have given The useful provision of a procedure for inspiring and outlining security in web administration frameworks [15]. In this paper, they introduce the provision of the Process for Web Service Security (Pwssec), created by the creators, to a genuine web administration based research endeavor. The way in which security in between authoritative data frameworks could be investigated, outlined and actualized by applying Pwssec, which joins a danger investigation and administration, alongside a security building design and a standard-based methodology, is likewise demonstrated. They furthermore display an apparatus fabricated to give backing to the Pwssec proces.

Eric Jui-Lin Lu **et.al.** proposed a XML multi-signature plan [16]. In this paper, by using the coherent XML structure and Wu's plan, they stated a XML multi-signature plan. In this plan, it takes the benefits of Wu's plan, further enhances the effectiveness in multi-signature era by marking the principles as opposed to the sub-documents, gives fine-grained control at the component level, and additionally is perfect with the standard XML Signature. Also, this plan proposes another mark era approach that is focused around the structure of the marked report instead of the marked archive alone.

F.song **et.al.** have given electronic voting plan about elgamal visually impaired marks focused around XML [17]. This paper display an electronic voting calculation about Elgamal visually impaired mark focused around XML and dissect its security, bookkeeping to the present electronic voting plan and the Elgamal visually impaired mark calculation. The project utilizes the particular of XML advanced mark and the engineering of Elgamal visually impaired mark calculation and has great security and viable significance.

Peter Buneman **et.al.** discussed about the keys utilized for XML security [18]. In this paper they talked about the meaning of keys for XML reports, giving careful consideration to the idea of a relative key, which is regularly utilized within progressively organized reports and exploratory databases.

Eric Jui-Lin Lu **et.al.** examined an exact investigation of XML/EDI [19]. In this paper, the creator altogether mulled over and dissected the XML/EDI skeleton proposed by XML/EDI gather, and outlined and additionally actualized a XML/EDI model. This paper additionally proposes a general usage strategy for the commercial ventures that are in the provision of XML/EDI. Dr Andrew Blyth has given the thought regarding a XML-based construction modeling to perform information incorporation and information unification in weakness evaluations [20]. In this paper the creator exhibited a structural engineering focused around the encoding of data inside a XML archive. He likewise exhibited how, through requisition of the structural engineering, extensive amounts of security-related data might be caught inside a solitary database construction.

Ernesto Damiani **et.al.** have given the thought regarding configuration and usage of a right to gain entrance control processor for XML reports [21]. In this paper an Access Control System for XML is depicted considering definition and requirement of access limitations specifically on the structure and substance of XML reports, in this manner giving a basic and compelling path for clients to secure data at the same granularity level gave by the dialect itself.

Peter Michalek has given the thought regarding analyzing requisition security of XML Schema's [22]. This article depicts the state of the workmanship and conceivable bearings later on. It outlines industry endeavors and focal points on requisition security related XML patterns being created inside Oasis(advancing Open Standards for the Information Society). Denis Trcek has given an integral framework for the security management of information systems [23]. This article shows an endeavor at administration of E-business frameworks security that is focused around coordinating existing methodologies in an adjusted manner. To cultivate handy utilization of the applied model in this paper, short foundation learning in related zones is given.

Alfredo Cuzzocrea **et.al.** have given the security saving OLAP over dispersed XML information [24]. This paper stated a novel Secure Multiparty Computation (SMC)-based security saving OLAP system for appropriated accumulations of XML records. The system has numerous novel characteristics extending from pleasant hypothetical properties to a compelling and effective convention, called Secure Distributed OLAP conglomeration convention (SDO). The proficiency of this methodology has been accepted by an exploratory assessment over disseminated accumulations of engineered, benchmark and genuine XML reports.

Jae-Gil Lee **et.al.** portrayed about secure inquiry transforming against scrambled XML information utilizing inquiry mindful unscrambling [25]. In this paper, they proposed the thought of Query- Mindful Decryption for effective handling of questions against scrambled XML information.

Dr Renato Iannella has given the thought regarding the Odr1(open Digital Rights Language), XML for computerized right administration [26]. This paper

gives a short outline of Drm(digital Right Management) emulated by a nitty gritty take a gander at the ODRL dialect and its utilization of XML. At long last, the OMA profile of ODRL has been audited to show how XML-based Rels(rights Expression Languages) are constantly utilized and stretched out by the group.

3.2 Motivation

As XML is widely used in business transactions, so the security of these XML data is essential to meet the business requirements. Data in transit and in servers can be effectively secured by XML Signature and XML Encryption techniques. Conventional techniques uses SSL/TLS which is not sufficient because it does not provide the security of data once it reaches the server side. This drawback can be removed by using XML Signature and XML Encryption following W3C standards. Here we are focusing on XML Encryption. Different author previously introduced different techniques like the concept of Document security language etc but there were some demerits with those technique. The effectiveness of XML signing and encrypting also depends on Parsers like DOM etc. Here we have used AES and 3DES for Encryption.

3.3 Objective

The objective of this thesis is to apply AES and 3DES algorithm to encrypt sensitive data. As these are symmetric key algorithms so to encrypt secret key RSA algorithm is used. Time elapsed in Encrypting is also calculated using these algorithms. Here we have used DOM parser. Time analysis for encryption and decryption is also shown by graph.

Chapter 4

Implementation and Results

4.1 Introduction

We have applied AES and Triple-DES algorithms to encrypt XML data. As these are symmetric key algorithms, So the key used in encrypting data is encrypted again using RSA algorithm. Time for encryption and decryption is also calculated. Graphs for encryption and decryption time are also drawn for different xml files.

4.2 Implementation

Language Used: "JavaScript"

Libraries Used:

- **"xml-encryption"** for XML Encryption

Platform Used: "Node.js"

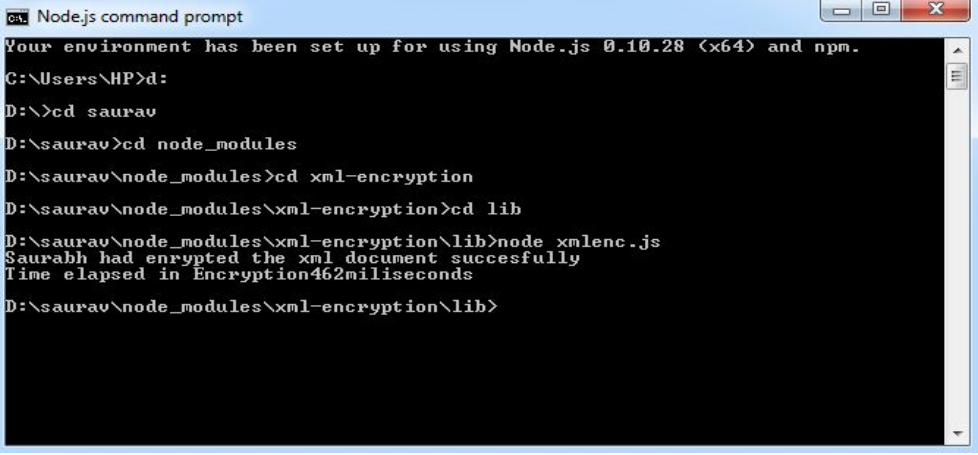
Node.js is a software platform for adaptable server-side and systems administration requisitions. Node.js provisions are composed in Javascript, and could be run inside the Node.js runtime on Mac OS X, Windows and Linux with no progressions. Node.js requisitions are intended to expand throughput and effectiveness. Here it provides an environment to sign and encrypt XML document.

4.3 Results

Figure 4.1 shows the input XML file.. For Encryption we have used AES-128, AES-256 and 3DES in CBC mode to encrypt the xml file. To encrypt the key,

```
<saaurabh>
  <book>
    <name>XML Security</name>
    <price>300</price>
    <publication>TMH</publication>
    <py>2005</py>
  </book>
</saaurabh>
```

Figure 4.1: Input XML File



```
Node.js command prompt
Your environment has been set up for using Node.js 0.10.28 (x64) and npm.
C:\Users\HP>d:
D:\>cd saurav
D:\saurav>cd node_modules
D:\saurav\node_modules>cd xml-encryption
D:\saurav\node_modules\xml-encryption>cd lib
D:\saurav\node_modules\xml-encryption\lib>node xmlenc.js
Saurabh had encrypted the xml document successfully
Time elapsed in Encryption462milliseconds
D:\saurav\node_modules\xml-encryption\lib>
```

Figure 4.2: Encryption Result

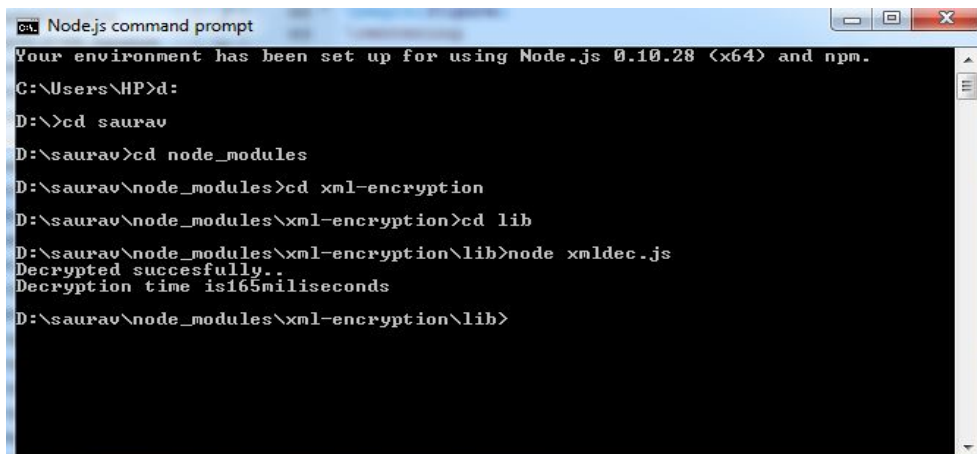
RSA is used and put into the EncryptedKey element. The output file is stored in the system in encrypted syntax. After encrypting console shows the message that it is encrypted successfully. Encryption time is also shown in the console. Figure 4.2 shows the snapshot of the console output after encrypting the given XML. After Encryptin, the encrypted file generates which contains the encrypted data in standard format. Figure 4.3 shows the output encrypted file. Decryption process results in original xml which is encrypted. Figure 4.4 shows the console output for decryption. Different XML files are taken having different number of elements in the input file for time analysis of encryption and decryption. As the number of elements in the input XML file increases the time for corresponding operations are shown by graph. Generally, time is proportional to the number of

```

<xenc:EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element"
  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
  <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#aes-256-cbc" />
  <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
    <e:EncryptedKey xmlns:e="http://www.w3.org/2001/04/xmlenc#">
      <e:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-oaep-mgf1p">
        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
      </e:EncryptionMethod>
      <KeyInfo>
        <X509Data><X509Certificate>MIIEDzCCAg3uaLv1AUo= </X509Certificate></X509Data>
      </KeyInfo>
      <e:CipherData>
        <e:CipherValue>sGH0hhzkjmlWYYY0gyQMampDMgewHMBtZafk1MHh9A= </e:CipherValue>
      </e:CipherData>
    </e:EncryptedKey>
  </KeyInfo>
  <xenc:CipherData>
    <xenc:CipherValue>V3Vb1vD1055Lp92zvK kNzP6xTu7/L9EMAEU </xenc:CipherValue>
  </xenc:CipherData>
</xenc:EncryptedData>

```

Figure 4.3: Encryption Result



```

Node.js command prompt
Your environment has been set up for using Node.js 0.10.28 (x64) and npm.
C:\Users\NHP>d:
D:\>cd saurav
D:\saurav>cd node_modules
D:\saurav\node_modules>cd xml-encryption
D:\saurav\node_modules\xml-encryption>cd lib
D:\saurav\node_modules\xml-encryption\lib>node xmldec.js
Decrypted succesfully..
Decryption time is165miliseconds
D:\saurav\node_modules\xml-encryption\lib>

```

Figure 4.4: Decryption output

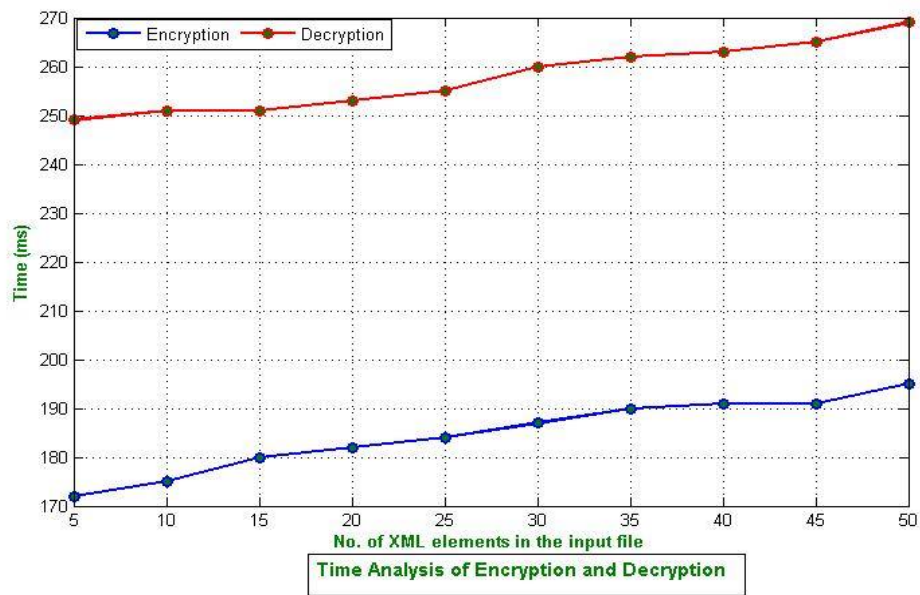


Figure 4.5: Time Analysis of Encryption and Decryption

elements in the input XML file. Here DOM parser is used. As the running time depends on current CPU utilization, number of applications running at that time. So to find a time, concerned code is executed 10 times and average is calculated to find nearly exact time. Similarly time is calculated for different xml files by increasing the number of elements in the input file.

Figure 4.5 shows the time analysis for encryption and decryption.

Chapter 5

Conclusion and Future Work

In this thesis we have applied AES and 3DES algorithm in CBC mode. We can specify the algorithm in the code which we want to use. In future we can extend our library for customized algorithm. So the conclusion is that W3C standard can be expanded by using customized or user defined algorithms using this library.

Here we conclude that, as the number of elements in the input XML file increases the execution time also increases. That is execution time is proportional to number of elements in the input XML file.

Future work may be implementing the XML Signature and Encryption with other algorithms which are not defined in W3C standard. Further memory consumption for the same can be calculated.

Bibliography

- [1] D. Eastlake, “Xml encryption syntax and processing,” *W3C Recommendation*, 2003.
- [2] A. C. Weaver, “Secure sockets layer,” *Computer*, vol. 39, no. 4, pp. 88–90, 2006.
- [3] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, and F. Yergeau, “Extensible markup language (xml),” *World Wide Web Consortium Recommendation REC-xml-19980210*. <http://www.w3.org/TR/1998/REC-xml-19980210>, 1998.
- [4] K. Beznosov, D. J. Flinn, S. Kawamoto, and B. Hartman, “Introduction to web services and their security,” *Information Security Technical Report*, vol. 10, no. 1, pp. 2–14, 2005.
- [5] T. Imamura, A. Clark, and H. Maruyama, “A stream-based implementation of xml encryption,” in *Proceedings of the 2002 ACM workshop on XML security*, pp. 11–17, 2002.
- [6] K. Komathy, V. Ramachandran, and P. Vivekanandan, “Security for xml messaging services a component-based approach,” *Journal of network and computer applications*, vol. 26, no. 2, pp. 197–211, 2003.
- [7] J. Efrim Boritz and W. G. No, “Security in xml-based financial reporting services on the internet,” *Journal of Accounting and Public Policy*, vol. 24, no. 1, pp. 11–35, 2005.

-
- [8] G.-H. Hwang and T.-K. Chang, “An operational model and language support for securing xml documents,” *Computers & Security*, vol. 23, no. 6, pp. 498–529, 2004.
- [9] T.-K. Chang and G.-H. Hwang, “The design and implementation of an application program interface for securing xml documents,” *Journal of Systems and Software*, vol. 80, no. 8, pp. 1362–1374, 2007.
- [10] D. C. Yen, S.-M. Huang, and C.-Y. Ku, “The impact and implementation of xml on business-to-business commerce,” *Computer Standards & Interfaces*, vol. 24, no. 4, pp. 347–362, 2002.
- [11] J.-M. Nurmilaakso, P. Kotinurmi, and H. Laesvuori, “Xml-based e-business frameworks and standardization,” *Computer Standards & Interfaces*, vol. 28, no. 5, pp. 585–599, 2006.
- [12] A. Kaliontzoglou, P. Sklavos, T. Karantjias, and D. Polemi, “A secure e-government platform architecture for small to medium sized public organizations,” *Electronic Commerce Research and Applications*, vol. 4, no. 2, pp. 174–186, 2005.
- [13] P. Kearney, “Message level security for web services,” *Information Security Technical Report*, vol. 10, no. 1, pp. 41–50, 2005.
- [14] H.-K. Ko, M.-J. Kim, and S. Lee, “On the efficiency of secure xml broadcasting,” *Information Sciences*, vol. 177, no. 24, pp. 5505–5521, 2007.
- [15] C. Gutiérrez, D. G. Rosado, and E. Fernández-Medina, “The practical application of a process for eliciting and designing security in web service systems,” *Information and Software Technology*, vol. 51, no. 12, pp. 1712–1738, 2009.
- [16] E. J.-L. Lu and R.-F. Chen, “An xml multisignature scheme,” *Applied Mathematics and Computation*, vol. 149, no. 1, pp. 1–14, 2004.
- [17] F. Song and Z. Cui, “Electronic voting scheme about elgamal blind-signatures based on xml,” *Procedia Engineering*, vol. 29, pp. 2721–2725, 2012.

-
- [18] P. Buneman, S. Davidson, W. Fan, C. Hara, and W.-C. Tan, “Keys for xml,” *Computer networks*, vol. 39, no. 5, pp. 473–487, 2002.
- [19] E. J.-L. Lu, R.-H. Tsai, and S. Chou, “An empirical study of xml/edi,” *Journal of Systems and Software*, vol. 58, no. 3, pp. 271–279, 2001.
- [20] A. Blyth, “An xml-based architecture to perform data integration and data unification in vulnerability assessments,” *Information Security Technical Report*, vol. 8, no. 4, pp. 14–25, 2003.
- [21] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati, “Design and implementation of an access control processor for xml documents,” *Computer Networks*, vol. 33, no. 1, pp. 59–75, 2000.
- [22] P. Michalek, “Dissecting application security xml schemas: Avdl, was, oval—state of the xml security standards report,” *Information Security Technical Report*, vol. 9, no. 3, pp. 66–76, 2004.
- [23] D. Trèek, “An integral framework for information systems security management,” *Computers & Security*, vol. 22, no. 4, pp. 337–360, 2003.
- [24] A. Cuzzocrea and E. Bertino, “Privacy preserving olap over distributed xml data: A theoretically-sound secure-multiparty-computation approach,” *Journal of Computer and System Sciences*, vol. 77, no. 6, pp. 965–987, 2011.
- [25] J.-G. Lee and K.-Y. Whang, “Secure query processing against encrypted xml data using query-aware decryption,” *Information sciences*, vol. 176, no. 13, pp. 1928–1947, 2006.
- [26] R. Iannella, “The open digital rights language: Xml for digital rights management,” *Information Security Technical Report*, vol. 9, no. 3, pp. 47–55, 2004.