CHAPTER 6

MOTIVATION TOWARDS SIGNCRYPTION RE-CRYPTOGRAPHY: SECURE AND EFFICIENT APPROACH TOWARDS TRUST PROBLEM

Cryptography is a discipline of computer science that directs the requirement specifications for satisfactory protection mechanism with efficient and smooth functioning in the real world. Signcryption is one of the most promising primitives of cryptography that was proposed by Zheng (1997), that rationally combines digital signature and encryption in a single step, lowers the computational and communications cost when compared with the cost of separate signature and encryption schemes. The concept of proxy re-cryptography was first proposed by Blaze at Eurocrypt (1998), and further be dignified by Ateniese and Hohenberger (2005). They defined their model by using two approaches like proxy re-signature and proxy re-encryption. In this chapter, we have directed towards a probably secure and efficient approach regarding the trust problem for third party, who is not directly involved ‘called proxy’, can be solved by using signcryption re-cryptographic approach. In modern era of cryptography, this is one of the new diverse trends and motivating issues. To solve the crypto logical problems such as trust and ciphertext access control problems, where research focuses on situations under a cryptographic key management by a semi-trusted proxy with special information where data encrypted under one cryptographic key need to be re-encrypted. Further, the proposed work is simulated on AVISPA/SPAN, using the automated formal verification tool.

6.1 INTRODUCTION

Diffie-and-Hellman (1976) [12] first proposed the idea of public key cryptographic protocol wherein the public key infrastructure (PKI) is developed for generating and maintaining the public-keys using the corresponding certificates. However, the PKI suffers from heavy management of public keys and certificates. An alternative solution is Shamir’s identity-based crypto systems (IBC). However, shortcoming of IBC is the key escrow problem [117]. The key escrow is a key exchange process in cryptography where a key is held or escrow, by a third party. The key gets compromised or lost by its original user(s) may be used to decrypt encrypted
matter, and allows restoration of the primary matter to its unencrypted state. Somewhere the third party involved is risky in escrow systems. Key escrow enables us to provide a backup source for cryptographic keys. The modern cryptography is an interdisciplinary approach of computer science focusing on the trust problem which is solved by using the proxy re-cryptographic primitive. The concept of proxy re-cryptography was first proposed by Blaze, Bleumer, and Strauss (1998). This approach was formalized by Ateniese and Hohenberger (2005). It consists of two methods as proxy re-encryption and proxy re-signature. The proxy re-encryption goal is to applied encryption again on generated cipher texts, without believing on honest parties and the proxy re-signature goal is applied to sign to transform into different signature on the same message trustworthy without relying on involved parties. In (2006) they proposed enhanced few proxy re-signature schemes and is discussed the potential applications related to same. They predicted that proxy re-signature and proxy re-encryption is played a crucial role. After that many researchers thoroughly are sparked more light in this area. That’s how some excellent schemes are proposed; where IEEE P1363.3 standardization group is established on proxy re-encryption, which is giving power to proxy re-cryptography approach [118]. A semi-trusted is an entity to convert cipher texts addressed to those which can be decrypted by using some special information.

For primitives of the proxy re-cryptography such as, signcryption proxy re-signature (SCPRS), signcryption proxy re-encryption (SCPRE), and security models are motivated by the same [139], [98].

In this chapter, a more optimized notion of signcryption with proxy re-cryptographic definition and its formal verification have been resented, and its efficiency motivation is specified. Finally, it provides directions for further research in this area in the concluding section.

6.1.1 Trust Problem

To solve the trustworthy problem within the domain of fully trusted authority to build the absolute trust relationship is a challenging issue. The public-key infrastructure certificate authority releases a public-key certificate to bind with the identity [119]. However, how to build offshore trust relationships between honest, trusted authority domains is a difficult task is a
practical problem. The goal is to solve the problem using proxy who allows in transferring certificates, and the proxy can’t generate new certificates. Sometimes it is desired that certificates of authority only transfer in a single direction known as unidirectional transformation. Bidirectional transformation is allowed to authorize in both directions. On the other hand, a trusted domain a requirement is further be extended the process that continues from one of proxy to many more proxies is known as multiuse. Trust problem is the significant asset in cryptographic primitive to solve such problems.

6.1.2 Trusted Server Problem

This problem is emerged with the cloud computing that reduces the cost of software and hardware resources. Almost all cloud storage servers are exerting and are responsible for sensitive information, like electronic storage user’s data, and access the cloud server over the data access. In usual the cloud access control server is considered to be fully trusted, but particular requirement doesn’t met due two practical reasons. First one is that the provider(s) of control service can’t be assumed to be fully trustworthy, the other being it could be corrupted in situations.

A possible solution is to store the encrypted plaintext at the server of cloud storage. The trusted server problem can be solved easily through this. The encrypted cipher texts need to be shared with others and no right to perform decryption by the access control server. Under this condition, the following solution can be conceived: - as the control server access right is to transform the cipher texts therefore only delegated users decrypts cipher texts, but control server access can’t decrypt cipher texts. If the access control server under Encryptor authorization can stored information on the cloud storage server in a new form then only designated receivers can decrypt, this is one of the specific case of proxy re-encryption [120]-[121].

6.1.3 Ciphertext Access Control Problem

The data processed under the specific circumstances is somewhere are intended to be stored for the set of users as a security concerns. The most motivating solution is owner data lays down in plaintext at the storage server, and rights for each user’s access are designed. The each user
specified by the access control lists services and linked to the access the message through control server. Therefore, security and trust issues are important issues in practice.

Data storage is a trivial method that stored into ciphertext. However, the current encryption system can’t allow being efficient shared among a user group on cipher text. It is becoming essential to develop a flexible and efficient method that directly share data based on encrypted plaintexts and it also includes the access policy control services. Bethencourt proposed a ciphertext-policy attribute-based encryption (CP-ABE) [122] approach that is appropriately initiated in solving the ciphertext access control problem [123].

6.2 SIGNCRYPTION

Signcryption is one of the cryptographic primitives, proposed by Zheng (1997), which logically combines digital signature and encryption in a single step on low communication and computational cost [113], [114]. This brings savings in communication and computation. There are various and huge applications of signcryption available that are being widely used for electronic commerce in sheltered and substantiated transactions, invulnerable and validated message delivery, safe, fast and non-repudiable transportation services.

After that many schemes are proposed with their own problems and limitations while they are offering different levels of computational costs and security services. Through the algorithm confidentiality and integrity is achieved [124]. The digital signature (DS) is a fully demonstrates with the mathematical explanation for the authenticity of its message digests. This DS scheme generally consists of the three steps:

The key generation that selects a personal key at random from the possible set of particular keys, that output's private key and its corresponding public value.

i. On behalf of the message and private key it produces the signature and

ii. After this the verification phrase occurs on the message, public keys and signature.

A signcryption scheme that includes DS as well as encryption consists of typically five phases, such as: Setup, Key Generation by Sender, Key Generation by Responder, Signcryption, and Unsigncrypt. Signcryption is extensively accepted in many application areas in ability to
connection to Internet, PDAs digital phones, session key establishment on ATM networks, etc. [27].

**6.3 PROXY RE-CRYPTOGRAPHY**

This is used to establish the trust relationship in an unsecured environment instead of that there are many applications such as digital-right management (DRM) that prevents the illegal redistribution of digital content. In 2006, Taban [125] proposed an entirely new interoperability architecture or modern module in the existing DRM called the domain interoperability manager (DIM). It applies a unique signature scheme and a particular public key encryption scheme. The traditional public key encryption and signature don’t support transformation, but using proxy re-cryptography this can be easily implemented. This scheme contains the two phases as: proxy re-signature and proxy re-encryption. Each phase contains its own properties and definition. A pictorial proxy re-cryptography digests approach is shown in Figure 6.1.

![Proxy Re-Cryptography Diagram](image)

**Figure 6.1:** Proxy Re-Cryptography Digest

**6.3.1 Proxy Re-Signature (PRS)**

In this scheme, a delegate’s signature transforms his/her signature using a semi-trusted proxy to a delegatee’s on the same message by using some additional information. The proxy can’t generate an arbitrary signature on behalf of either the delegate or the delegatee.
6.3.1.1 Properties of Proxy Re-Signature

(i) A Unidirectional or Bi-directional: The proxy is to allow for re-signature key either in uni-directional or in bidirectional transformation.

(ii) Multiuse: In this case, the proxy transformed the signature can be re-transformed again by a proxy. Even so, the signature does not transform a single use.

(iii) Private Proxy: In private proxy, the re-signature keys to be secret in scheme.

(iv) Transparent: The scheme should be see-through so that the user(s) does not know about the where proxy is existed.

(v) Key-Optimal: In this, a user is required to protect and store only a small constant amount of secrets, no matter how many signature delegations the user gives for acceptance.

(vi) Non-interactive: The parties involved are an idle and are not required during the commission process.

(vii) Non-transitive: Other than the two, signature can’t be generated from anywhere in any case for the same.

(viii) Temporary: The right of re-signing is interim. It is necessary to specify the involment of right to access or right to expire at particular moment.

(ix) Collusion resistance: Via proxy, the delegator consigns the signing rights to the entrusted delegate, instead of keeping the rights decryption for the same public key.

6.3.1.2 Definition of Proxy Re-Signature

The proxy re-signature follows the following five steps:

(i) Key Generation: The security parameter $\lambda$ is taken as input, and that returns a verification key $pk$ and a signing key $sk$.

(ii) Re-Key Generation: It takes as an input delegate key pair $(pk_A, sk_A)$, and a delegatee key $(pk_B, sk_B)$, and returns a re-signature key $rk_{A\rightarrow B}$ for the proxy. If the scheme is unidirectional, the delegates signing key are not included in the input. But in the case of bidirectional, the proxy can be easily obtained by $rk_{B\rightarrow A}$ from $rk_{A\rightarrow B}$. In many bidirectional schemes $rk_{A\rightarrow B} = 1/rk_{B\rightarrow A}$.
(iii) **Signature:** It takes as input a signing key $sk$, a positive integer $l$, and a message $m$ from message space, and returns a signature $\sigma$ at level $l$. If this scheme is for single use, then $l \in \{1,2\}$.

(iv) **Re-signature:** It takes as input a re-signature key $rk_{A\rightarrow B}$, and a signature $\sigma_A$, taking place message $m$ under $pk_A$, on level $l$, and returns the signature $\sigma_B$ on the same message $m$ under $pk_B$, at level $l + 1$ if verified $(pk_A, m, pk_B, l) = 1$, or reject otherwise. If the scheme is for single use $l = 1$.

(v) **Verify:** This takes as input a verification key $pk$, the message $m$ from the message space, the signature $\sigma$ and a positive integer $l$, and returns 1 if $\sigma$ is a valid signature under $pk$ at level $l$ or otherwise.

### 6.4 SIGNCRYPTION WITH PROXY RE-ENCRYPTION

The proxy signcryption scheme has the general condition, which is divided into three parties such as delegate signer, proxy signer and the delegatee recipient. In this scheme, the delegate signer generates a proxy credential to the signing authority to a proxy signer. The proxy there after generates signcrypted message using a secret key and its own proxy credentials. Finally, the proxy sends the signcrypted message to an assigned recipient through a network. After receiving the signcrypted message, the recipient recovers the content from the same and also verifies its validity. If any dispute arises, the recipient is free to announce the signature of proxy for public verification.

The notion of signcryption [126] with proxy re-encryption [127] has been presented here. This scheme consists of proxy re-encryption, authenticity and confidentiality in a very efficient way. This primitive have various applications, such as:

(i) Email is the best candidate for applying signcryption. An application of signcryption of proxy re-encryption (SCPRE) is to allow and forward the message for authentication using signcrypted message to be directed to a person when the original receiver is unavailable.

(ii) Another well-known application for secure and authentic distributed storage that can be extended whenever the content stored for authentication is desirable.

The signcryption of proxy re-encryption scheme follows the following steps:

i. **Setup:** The algorithm accepts a security parameter $l$ and outputs a master secret key $s$.

ii. **Extraction:** The algorithm accepts an identity $ID_u$, and outputs the secret key $S_u$. 

121
iii. Extract-rekey: It accepts two identities $ID_1$ and $ID_2$, and outputs the rekey from $ID_1$ and $ID_2$.

iv. Signcryption: The signcryption accepts messages $m$, and two identities $ID_1$ and $ID_2$, and outputs the signcryption for $m$ from $ID_1$ and $ID_2$.

v. De-signcrypt: This accepts a signcryption message $\varphi$ and identity $ID_r$, and outputs the de-signcryption of $\varphi$ by $ID_r$.

vi. Re-encryption: It accepts a signcryption $\varphi$, and an identity $ID_r$, and outputs the re-encrypted signcryption $\varphi'$ of $\varphi$ to $ID_r$.

vii. De-re-encrypt: This accepts a second-level signcryption $\varphi'$ and $ID_d$, and outputs the de-signcryption of $\varphi'$ by way of $ID_d$.

6.4.1 The Scheme of signcryption proxy re-encryption (SCPRE)

The SCPRE scheme is derived from the identity-based signcryption scheme; the scheme is presented as follows:

**Setup**

Let $l$ be the security parameter of the system. Let $G_1$ and $G_2$ be two prime ordered groups of order $q = \theta(2^l)$, where $G_1$ be represented additively, and $G_2$ be represented multiplicatively. Let $P$ be a generator of $G_1$.

Let $e : G_1 \times G_2 \rightarrow G_2$, be a bilinear pairing. We assume that the Bilinear Computational Diffie-Hellman (BCDH) assumption holds in $< e, G_1, G_2 >$.

It uses four hash functions $H_0, H_1, H_2$ and $H_3$, where

- $H_0 : \{0,1\}^\ast \rightarrow G_1$,
- $H_1 : G_1 \times \{0,1\}^n \rightarrow Z_q^\ast$.
- $H_2 : G_2 \rightarrow \{0,1\}^{n+t}$
- $H_3 : G_1 \times \{0,1\}^\ast \rightarrow G_1$

The $n$ is the number of bits in the message, and $t$ is the number of bits used to represent an element in $G_1$. The private key generator (PKG) chooses the master secret key $s \in R Z_q^\ast$ and sets the master public key $P_{pub} = sP$. The published public parameters are
< e, G_1, G_2, n, q, P, P_{pub}, H_0, H_1, H_2 >. Each user has his/her identity ID_u, and public key. He/she gets two secret keys S_u, and S_{ul||delegatee}, by providing ID_u and ID_u||"delegatee".

**Extract (ID_u)**

The public key generator (PKG) computes the secret key as S_u = s * H_0(ID_u), where H_0(ID_u), is generally denoted by Q_u

**Signcrypt (m, S_A, ID_B)**

User A is to signcrypt a message m from delegator A to delegate B by using the following steps as:

1. Choose \( r \in R Z_q^* \)
2. Compute \( X = rQ_A \) and \( h = H_1(X||m) \)
3. Compute the signature \( Z = (r + h)S_A \)
4. Choose \( k \in R G_2 \)
5. Compute \( Z = e(S_A, Q_B)^\tau \), and set \( \lambda = w.k \)
6. \( y = H_2(k) \oplus (m||Z) \)
7. The signcryption is \( \emptyset = < X, y, ID_A >. \)

**De-signcrypt (\( \emptyset = < X, y, \lambda, ID_A >, S_B \))**

The delegatee receiver B, after receiving the signcryption \( \emptyset \), does the following.

1. \( w = e(X, S_B) \)
2. Compute \( k = \lambda w^{-1} \)
3. Recover \( m||Z = y \oplus H_2(k) \)
4. \( h_1 = H_1(X||m) \)
5. If \( e(Z,P) = e(P_{pub}, X + h_1 \cdot Q_A) \), then \( < m, (X,Z), ID_A > \) This is the output as the message and signature. Otherwise, \( \bot \) is output.

**Rekey-Extract (S_B, ID_C)**

B sends \( r_{k_{B\rightarrow C}} = < -S_B + H_3(e(S_B, Q_{(c||delegatee)})) >. \) to the proxy.

Re-encrypt (\( \emptyset = < X, y, \lambda, ID_A >, r_{k_{B\rightarrow C}}, ID_B, ID_C \))
The proxy computes re-encrypted signcryption $\mathcal{O}' = \langle X, y, \lambda, e(X, rK_{B\rightarrow C}), ID_A, ID_B \rangle$, and sends $\mathcal{O}'$ to $C$.

**De-re-encrypt ($\mathcal{O}' = \langle X, y, \lambda', ID_A, ID_B \rangle, S_{c||delegatee}$)**

On receipt of a level 2 signcryption, $C$ decodes the algorithm as follows:

1. $w = e\left(X, H_3, \left(e(Q_B, S_{c||delegatee})\right)\right)$
2. Compute $k = \lambda' w^{-1}$
3. Recover $m||Z = y \oplus H_2(k)$
4. $h_1 = H_1(X||m)$
5. If $e(Z, P) = e(P_{Pub}, X + h_1 Q_A)$, then output $\langle m, (X, Z), ID_A \rangle$, else output $\bot$.

A collective thought for proxy re-signature and re-encryption schemes is to establish secure applications scenarios on a long term basis.

### 6.5 FORMAL VALIDATION USING AVISPA/SPAN TOOL

We simulated signcrypted proxy re-cryptographic approach in CAS implementation language and it is shown with sender principal pattern information executed on OFMC back end tool. It is a useful tool that allows and checks all participated agents to execute all the specified steps as a honest run participants, resultant in form of SAFE state, as depicted in Figure 6.2.

![Figure 6.2: SPAN on OFMC Back End](image-url)
CL-ATSE is a set of constraints, is working under to find attacks on designed protocols. The intermediate translation and checking are work automatic and internally performed on the same. This has executed same on AtSe tool, as shown in Figure 6.3, which is a presentation with negligible possibility of attack.

**Figure 6.3:** SPAN on AtSe Protocol Check

**Figure 6.4:** Sender pattern principal
The specification is automatically simulated in the proposed approach between delegator and delegatee via a third party of proxy. Here in figure 6.4, the pattern of sender principal is shown according to the above provided definition. The delegator, sends the message to proxy, where secret via proxy is added and sent to the delegatee where it is deciphered.

This permits a CAS+ specification to translation for simple and fast specification of security protocols; interactively building a Message Sequence Chart (MSC) [110], [111]. But, originally messages are sent in the form of encrypted form over algorithm, where it is like to be impossible to decrypt, as depicted in Figure 6.5.

The definition has simulated with the Intruder with its knowledge, in Figure 6.6, with the real sender pattern principle.
Further, in the last but not least, the various additional composition behaviors are also available, as exposed in Figure 6.7.

The analysis of four back-ends are harmonized to each other in a sense for some common back-ends procedure, but these are not equivalent so that should return different results.

![Figure 6.7: With Intruder Real Type Pattern with Emissions](image)

### 6.6 SUMMARY

The presented work is a motivation for the new direction of cryptography using the approach proxy re-cryptography for secure signcryption based protocol. The use of Signcryption approach is the new paradigm that fulfilling the most desired cryptographic applications and demanding due to the most of cost effective in sense of high performance, suits to low memory devices and so on. Further, we are highlighted some of future works as:-(i) to collect for the long-term schemes using proxy re-cryptography into a single location, though researchers can evaluate their suitability for various applications. (ii) The approach for modern cryptography with security requirements have arisen in different distributed environments as the attacks may come either from internal or external objects, (iii) proxy re-cryptography should be the standard model and it is collusion-resistant.