International Journal of Security and Its Applications
Vol. 10, No. 7 (2016) pp.103-118
http://dx.doi.org/10.14257/ijsia.2016.10.7.10

Certificateless Public Key Cryptography: A Research Survey

Jacob Sayid[1], Isaac Sayid[2] and Jayaprakash Kar[3]

[1]*Department of Information Technology*
[2]*Department of Information Systems,*
Information Security Research Group
Faculty of Computing & Information Technology
King Abdul-Aziz University, Kingdom of Saudi Arabia, Jeddah-21589
jgopabandhu@kau.edu.sa

Abstract

Certificateless Public Key Cryptography is a scheme that provides security by avoiding the key escrow and eliminating the drawback of identity-based cryptography. Several certificateless models have been proposed to enhance the efficiency and overcome adversaries attacks. In this paper, we survey various public key encryption schemes on certificateless setting with the security model and discuss the performance. Also, we present the generic model of Certificateless Public Key Encryption (CL-PKE) scheme proposed by various authors.

Keywords*: Certificateless Cryptography, Public Key, Identity-Based Cryptography, Certificateless signatures*

1. Introduction

Public-key encryption scheme was first come up by Rivest, Shamir and Adleman in 1978 [1]. It was based on public-key cryptography concept proposed by Diffie and Hellman in 1976 [2]. The scheme allows communicating parties to securely share a message without having previously shared-secret key. This is accomplished by generating two related keys: public-key, which is used for encryption and publicly known to others, and private-key, which is private for the user and used for decryption. However, public-key cryptography has shown some security issues. The sender has to authenticate the public key of the receiver in real world scenario. Otherwise incorrect public keys would be used to impersonate the receiver. An effective approach to provide authentication for public keys is to have a key management system called Public-Key Infrastructure. PKI is a network of trusted entities known as certificate authority (CA) that are used to authenticate the public key of the users. CAs map users' identity with their public keys and provide other key management functionalities such as certificate generations and revocations. This approach is expensive and inefficient process for any system that employs it.

1.2. Identity-Based Cryptography

To dispense with the need for PKI, identity-based cryptography was introduced by Shamir [3] in 1985. The idea is to eliminate the need for certificates by utilizing digitally unique information for public keys like email address. Obviously, this approach requires a trusted third party to generate private keys for its users. It is called Private Key Generator (PKG) and it distributes a master public key *mpk* and holds the corresponding master secret key *msk*. For a user to obtain a private key, he/she first communicates with the PKG, which in turn uses its *msk* to create the private key to the user. Nevertheless, identity-based cryptography experiences the key escrow drawback which allows a PKG to

ISSN: 1738-9976 IJSIA

disclose messages of its users because it has their private keys. Moreover, there are functional problems related to the key management's tasks such as key revocation [4].

1.2. Certificateless Cryptography

Certificateless cryptography provides security by solving the key escrow issue and avoids the problems of PKI-based schemes. Al-Riyami and Paterson [5] come out with certificateless cryptography concept in 2003. In this scheme, public keys do not require authentication. In other words, certificates and PKI are not required; instead, semi-trusted third party Key Generation Center (KGC) works to issue partial private keys D_{ID} similarly to the identity-based cryptography. To generate the actual private key, the user combines a chosen secret value with his/her D_{ID} value. Hence, the key escrow drawback is resolved in certificateless cryptography and the actual private key is no longer stored in the KGC.

The upcoming sections of this paper are structured as follows. In Section 2, we define the acronyms and notations used in the whole paper. In Section 3, we present the general definition of the certificateless schemes. Next, we describe the security models and details of some well-known certificateless schemes. Next, we survey and analysis some of the well-known certificateless schemes. Finally, we conclude the survey by comparisons of the certificateless schemes performance and security.

2. Preliminaries

Certificateless cryptography is a public-key scheme that provides security without the need for public-keys authentication. Certificateless cryptography avoids the key escrow drawback of identity-based cryptography and replaces the public-key infrastructure by KGC. In this section, we present acronyms and notations used all over the paper. Table. 1 describes the notation. A Certificateless public-key includes three components, the user, the KGC, and a verifier. It is described by a number of probabilistic polynomial-time PPT algorithms [5]. The details of the algorithms are as follows:

- **Setup:** This algorithm takes security parameter 1^k and returns *msk*, *mpk* and *param*. It is run by KGC.
- **Partial-Private-Key-Extract:** The input of this algorithm includes *msk*, *mpk*, *param* and identity ID**.** It returns D_{ID} and the KGC run this algorithm once for each user.
- **Set-Secret-Value:** This algorithm receives *mpk*, *param and* ID as input and returns X_{ID}. It is run by the user.
- **Set-Private-Key:** This algorithm takes *mpk*, D_{ID} and X_{ID} as input and outputs sk_{ID}. It is run once by the user**.**
- **Set-Public-Key:** This algorithm takes *mpk* and X_{ID} to return pk_{ID}. It is run once by the user**.**
- **Sign (Signcrypt):** The input of this algorithm includes *mpk*, *ID*, pk_{ID} and *message*. It outputs cipher text C or an error. It is run by the user/signer.
- **Verify (Unsigncrypt):** The input of this algorithm includes *mpk*, sk_{ID} and cipher text C. It outputs the original message or an error. It is run by the user/signer**.**

3. Security Models of Certificateless Cryptography

Security models are essential in cryptography, as they are used to analyze and define the attack power of different adversaries. Moreover, security models are used to simulate the adversary's actions and ensure their goals. In certificateless cryptography, security

models are used to theoretically prove whether a scheme is secure or not and provide estimation to possible attacks in practice. A certificateless scheme is alleged to be secure by providing a valid proof on its matching security game. A security game is a challenger where the adversaries try to submit queries to win this game if described conditions are met. Officially, in certificateless cryptography, the adversaries are categorized as Type I and II adversaries [5][6]. Following subsections provide definition and information of each type.

3.1. Adversary Types

A secure certificateless scheme should counter both Type I and Type II adversaries. A Type I adversary A_1 is an outsider, not KGC nor legal user, who has the ability to selectively modify public keys of any user and may be able to get some partial private-key values but cannot disclose *msk* value. In other hand, a Type II adversary A_2 is a KGC which is able to view *msk* value but cannot change public keys of any user[24][25].

3.2. Type I and Type II Attack Variations

There have been different attempts to specifically describe the actions involved for adversary A_1 as well as A_2 in the security model. In general, a certificateless encryption scheme is considered secure with variations of Type I adversaries if every PPT attacker has minimal advantage in winning the indistinguishable chosen ciphertext attack 2 game. A CL scheme would be secure against Type II attack variations, if every PPT attacker, with auxiliary information equals the master private key of KGC, has minimal advantage to win the INDCCA2 game. Table 3 summarizes the attack variations for Type I and Type II [5][21].

3.3. Adversary Power and Goals

We first define weakest adversary goals and strongest power [7][22]

- **Existential forgery goal**: when the adversary has the ability to generate at least one correct signature for a message that has not been signed before.
- **Strong forgery goal**: when the adversary has the ability to generate at least one correct but distinct signature for a message that has been signed before.
- **Adaptively chosen-message attack power**: when the adversary has the ability to select signed messages adaptively in the course of attack, where the signer can be used to sign messages while the attack time.

Table 1. Notation

Notation	Meaning	Notation	Meaning
MPK	Master Public Key	A_1	Type I adversary
MSK	Master Secret (Private) Key	A_2	Type II adversary
D_{ID}	Partial Private Key	*PARAM*	System parameter
pk_{ID}	User Public Key	*PPT*	Probabilistic, Polynomial-Time
sk_{ID}	Actual Private Key	IND-CCA	Indistinguishable Chosen Ciphertext Attack
X_{ID}	User Secret Value	IND-CPA	Indistinguishable Chosen Plaintext Attack
ID	User Identity		
KGC	Key Generation Center		
PKG	Private Key Generator		

Table 2. Different Variations of Type I and Type II Adversaries

Name	Name	Name
Strong Type I	Weak Type Ib	Weak Type II
Weak Type Ia	Weak Type Ic	Strong Type II

Combining the previous weak adversary goals and strongest adversary power, a secure certificateless scheme is treated as (1) existentially or (2) strong existentially unforgeable under adaptively chosen-message attacks. For the first scenario, the adversary goal is to get a faked signature σ^* using identity ID^* and message m^* under the following circumstances:

1. σ^* is valid signature for identity ID^* and message m^*
2. (ID^*, m^*) has not been sent to the signing oracle any time before
3. σ^* has never been outputted.

Moreover, for the second scenario strong unforgeability, the conditions are the same except for the second one:

2. (ID^*, m^*) has been sent before

3.4. Adversaries and Oracles

Oracles are used as responses to the adversaries' requests in the security model. There are five kinds of oracles in certificateless cryptography [8], and the access for each one is based on the security game specifications. Table 3 describes the oracles

Table 3. Adversaries and Oracles

ORACLE	INPUT	OUTPUT
CREATE-USER	ID	IF ID HAS NOT BEEN CONSTRUCTED BEFORE, THE ORACLE EXECUTES THE FOLLOWING AND ADDS THE RESULTS OF EACH TO A LIST L. D_{ID} ← PARTIAL-PRIVATE-KEY-EXTRACT X_{ID} ← SET-SECRET-VALUE pk_{ID} ← SET-PUBLIC-KEY
PUBLIC-KEY-REPLACE	1-ID 2-pk'_{ID}	-THE USER ID'S PUBLIC KEY IS REPLACED WITH pk'_{ID} -UPDATES THE RELATED RECORD IN THE LIST L
SECRET-VALUE-EXTRACT	ID	SECRET VALUE X_{ID}
PARTIAL-PRIVATE-KEY-EXTRACT	ID	IF ID HAS BEEN CONSTRUCTED BEFORE, THIS EXTRACTS D_{ID} FROM LIST L
SIGN (SIGNCRYPT)	1- *M* 2- *ID* 3-pk_{ID}	SIGNATURE σ← IF ID HAS BEEN CONSTRUCTED BEFORE

4. Survey of Proposed Protocols

In this section, we review a number of proposed schemes for certificateless public key encryption. For each scheme, we describe the algorithms, the security model and the performance. There are two types of CL-PKE scheme constructions: concrete and generic. Concrete construction of scheme is a distinct and fully described scheme while the generic one is based on and derived from other primitives.

4.1. Al-Riyami 1/Yum-Lee version CL-PKE [9]

The first scheme for (CL-PKE) was proposed by [5]. He defined nomenclature, basic notations and the first framework for CL-PKE. This scheme is considered generic and is based on PKE scheme and identity-based one. The scheme is built using elliptic curve pairings.

Algorithms:

The algorithm of the scheme comprises the following:

- **Initialization setup**: CL-PKE scheme is initiated by KGC. This process is done once per user and outputs *mpk* and *msk*. The security parameter is the input for the process.
- **Partial private key generation**: This process is executed again by KGC and once per user. It takes the output of initialization setup as input as well as the identity of the user to construct a partial private key belongs to the user.
- **Secret value setup**: The user run this process taking *mpk* and ID as input to generate a secret value that is required as input for other processes.
- **Private key**: This process generates a private key by taking (D_{ID}, *mpk*, X_{ID}) as input. The generated private key is going to be known only for the generator.

- **Public key**: This process generates public key by taking as input the X_{ID} of the user and the *mpk* that is extracted in the initial setup by KGC. This is done by the user.
- **Encrypt**: The input for encryption process is the *mpk*, user's ID, pk_{ID} and the message. It produces as an output an encrypted message that can be decrypted only by user. Note that in this version, the encrypted message is produced by first applying encryption on the message via public key scheme and then via identity scheme.
- **Decrypt**: The input for this process is sk_{ID} value of the user, *mpk* and the encrypted message, this process can decrypt and output the original decrypted message.

Security Model:

For type I model, the scheme uses Weak Type 1b model. For type II model, the scheme uses Weak Type II model. Note that partial private key process is not accessible by the attacker. This scheme was proven insecure by [10][11]

4.2. Al-Riyami 2 CL-PKE

A second version of the first scheme is proposed by [2] also. Again, the scheme is considered generic and is based on PKE scheme and Identity-based one. The scheme is built using elliptic curve pairings.

Algorithms:

The algorithm of the second version of the proposed scheme is almost similar to the first version. However, it's different in the following elements:

1- **Encrypt:** The input for encryption process is the *mpk*, user's ID, pk_{ID} and the message. It returns an encrypted message that can be decrypted only by user. Note that in Al-Riyami 2 version, the encrypted message is produced by first applying encryption on the message via identity scheme and then via public key scheme.

2- **Decrypt:** The decryption process in this scheme is almost identical to version 1. However, the only difference is going to be in the order of underlying operations due to the change in encryption process.

Performance

With regard to encryption process compared to other variations of this scheme, the process is done sequentially which is not efficient.

Security Model

This version of the scheme has no proof of A_1 and A_2 security models. This scheme was proven insecure by [9]

4.3. Al-Riyami 3 CL-PKE

A third version of the first scheme is presented by Al-Riyami. The scheme is of generic type since it is based on -PKE scheme and identity-based one. The scheme is built using elliptic curve pairings.

Algorithms:

The algorithm of the third version of the proposed scheme is almost similar to the first and second versions. However, it's different in the following elements:

- **Encrypt**: The input for encryption process is the *mpk*, user's ID, pk_{ID} and the both shares s1 and s2 of the message to be encrypted m. Where s1 + s2 = m. It outputs an encrypted message that can be decrypted only by user. Note that in Al-Riyami 3, the encrypted message is formed by applying parallel encryption on the shares of the message via identity scheme encryption and the other share via public key scheme.
- **Decrypt:** The decryption process in this scheme is going to be different from the other two versions since the message is split into two shares and each share is encrypted in by different schemes.

Performance:

With regard to both encryption and decryption processes compared to other variations of this scheme, the process is done in parallel which is more efficient.

Security Model:

This version of the scheme has no proof of $A_1$and A_2 security models. This scheme was proven insecure by[11].

4.1 Al-Riyami-Paterson 1 CL-PKE [5]

A concrete type scheme is presented by Al-Riyami-Paterson. The scheme is considered concrete because it is not based on primitives like identity to compute public key. The authors adapted Fujisaki-Okamoto padding technique [12] to the scheme in order to secure it against chosen cipher text attacks.

Algorithms:

The algorithms of the Al-Riyami-Paterson 1 scheme matches the basic CL-PKE scheme pointed to earlier in several elements. These elements include partial private key generation, Secret value setup, private and public keys generation. However, it's different in the following elements:

- **Initialization setup**: The only difference in this process is that on top of what parameters required for this process in basic CL-PKE; there are two additional cryptographic hash functions needed.
- **Encrypt and Decrypt**: The encryption and decryption processes in this scheme involve more steps to be applied including checking parameters and the public key structure.

Performance

The scheme includes more underlying operations in its framework. A single encryption process requires three pairing calculations. Therefore, it can be roughly concluded that it is less efficient than the previous generic schemes and also less efficient than the second Al-Riyami-Paterson 2 scheme.

Security Model

In random security model, the scheme's security was proven. The scheme uses Strong Type I for type I model and Weak Type II for type II model.

3.1 Al-Riyami-Paterson 2 CL-PKE [13]

Details

Two years after Al-Riyami-Paterson 1 scheme, the authors have proposed new generic type scheme that is based on identity-based and PKE scheme. The scheme is proposed to be more efficient than the first Al-Riyami-Patersion 1 scheme and more secure.

Algorithms

Al-Riyami-Paterson 2 scheme is a result of optimizing double encryption construction that consists of both identity based encryption of [14] and ElGamal PKE scheme [15] as the main components or building blocks of the scheme. The scheme algorithm comprises the following:

- **Initialization setup**: The input is security parameter to generate several outputs as groups to extract spaces of allowed values for message and other parameters. Moreover, a master key is generated in this process.
- **Partial private key generation**: The input for this process is the user's ID and generates the D_{ID}.
- Secret value setup: The input for this process is the output of initialization setup as well as the ID to generate a secret value.
- **Private key**: It is generated by taking as input the generated X_{ID}, output of initialization setup and entity's D_{ID}.
- **Public key**: It is generated using the exact same inputs of private key except the D_{ID}.
- **Encrypt and Decrypt**: The encryption processes in this scheme involves single pair computation per encryption which is more efficient than the Al-Riyami-Patersion 1 scheme. However, the cost of decryption is similar in both versions of the scheme. The input of encryption is the user identity and the public key. For decryption, private key is needed as input for the process.

Performance

The second scheme of Al-Riyami-Paterson 2 is more efficient than the first one since single pair computation is needed for encryption process. However, the decryption process costs are similar in both versions of the scheme.

Security Model

In random security model, the scheme's security was proven by the authors. The scheme uses Strong Type I for type I model and Weak Type II for type II model. Nevertheless, the security of this scheme is broken by [11][16][6]

3.2 Baek, Safavi-Naini and Susilo CL-PKE scheme

An improved scheme that doesn't use elliptic curve pairings in its underlying architecture was proposed in [17]. The scheme is considered a modified formulation of Al-Riyami-Paterson scheme. One of the main distinct parts in this scheme is that the public key cannot be computed before partial private key computation. This has a disadvantage of preventing future encryption of messages with a public key before the private key is obtained. In contrast, the scheme with such a formulation prevents "denial of decryption".

Algorithms

Baek scheme has five elements in its algorithms instead of seven. The reason is merging of three elements in Al-Riyami scheme into one element. In other words, private key, public key generation and secret value setup are combined into one element called user key generation. The algorithm of the suggested scheme comprises the following:

- **Initialization setup**: This process is similar to (Al-Riyami 1) initialization setup. It is executed by KGC taking security parameter as input and generating *msk*, *mpk* and other parameters. Moreover, a master key is generated in this process.
- **Partial private key generation**: Same as (Al-Riyami 1) partial private key generation process. The input for this process is the user ID, *msk* and *mpk* to generate D_{ID}.
- **Key Generation**: A process that combines multiple processes in (Al-Riyami 1) scheme. The input for this process is the D_{ID} and the user identity to generate public and private key for users.
- **Encrypt**: Exactly as encryption process in (Al-Riaymi 1) scheme. Takes the pk_{ID}, *mpk* and ID as input to generate an encrypted message.
- **Decrypt**: Different than decryption process in (Al-Riaymi 1) scheme. Takes the user private key and *mpk* as input to generate a decrypted message.

Performance

Compared to (Al-Riyami 1) scheme, this scheme is more efficient since it combines multiple processes of key generation into single process.

Security Model

Obviously, the scheme doesn't rely on secret value which makes Weak Type Ia security model improper. In random security model, the scheme's security was proven by the authors. The scheme uses Strong Type I for type I model and Weak Type II for type II model.

3.3 Libert–Quisquater CL-PKE scheme (concrete version)

An efficient concrete CL-PKE scheme was proposed by [11]. The scheme seems to be a newer version of Al-Riyami-Paterson scheme since the author called it "NewFullCLE". The scheme can be viewed as identity based encryption combined with ElGamal encryption.

Algorithms

Libert–Quisquater scheme has seven elements in its algorithm and it is highly similar to Al-Riyami-Paterson 2 scheme elements with little differences in some underlying operations. The elements in this scheme are initialization setup, partial private key generation, secret value setup, private key generation, public key generation, encrypt and decrypt

Performance

The scheme is comparable to Baek Scheme as the author claimed. It is more efficient than similar schemes proposed such as Shi and Li [18].

Security Model

In random security model, the scheme's security was proven by the authors. The scheme uses Strong Type I for type I model and Weak Type II for type II model.

3.4 Libert–Quisquater CL-PKE scheme (generic version 1)

Libert–Quisquater [11] proposed secured version of Al-Riyami 1 scheme. The proposed scheme applied Fujisaki-Okamoto transformation. This technique is a modified version of the Fujisaki-Ocamoto conversion which converts IND-CPA attacks into IND-CCA ones. The modification is done by including the user/receiver ID including the public key within the input of hash functions used in the scheme algorithms.

Algorithms

This scheme is almost identical to AL-Riyami 1 scheme in terms of efficiency and how the algorithms are executed except two of them. These two algorithms are the encryption and decryption. User identity is included in the hash functions used for both encryption and decryption.

Performance

The scheme is similar Al-Riyami 1 scheme since it extends it. The encryption process compared to other generic versions of Libert-Quiquater is considered less efficient.

Security Model

In random security model, the scheme uses Strong Type I for type I model and Weak Type II for type II model.

3.5 Libert–Quisquater CL-PKE scheme (generic version 2)

Libert–Quisquater [11] proposed secured version of Al-Riyami 2 scheme. The proposed scheme also applied Fujisaki-Okamoto transformation. As mentioned earlier, the technique is a modified version of the Fujisaki-Ocamoto conversion which converts IND-CPA attacks into IND-CCA attacks. The modification is done by including the user/receiver ID including the public key within the input of hash functions used in the scheme algorithms.

Algorithms

This scheme is similar to AL-Riyami 2 scheme in terms of efficiency and how the algorithm is executed except two steps which are the encryption and decryption. User identity is included in the hash functions and used for both encryption and decryption.

Performance

The scheme is similar Al-Riyami 2 scheme since it extends it. The encryption process compared to other generic versions of Libert-Quiquater is considered less efficient.

Security Model

In random security model, the scheme uses Strong Type I for type I model and Weak Type II for type II model.

3.6 Libert–Quisquater CL-PKE scheme (generic version 3)

Libert–Quisquater [11] proposed secured version of Al-Riyami 2 scheme. The proposed scheme also applied Fujisaki-Okamoto transformation. The transformation is done by including the user/receiver ID including the public key within the input of hash functions used in the scheme algorithms.

Algorithms

This scheme is similar to AL-Riyami 3 scheme in terms of efficiency and how the algorithms are executed except two elements. These two elements are the encryption and decryption. User identity is included in the hash functions used for both encryption and decryption.

Performance

The scheme is similar Al-Riyami 3 scheme since it extends it. The encryption process compared to other generic versions of Libert-Quiquater is considered more efficient since parallelism is exploited.

Security Model

In random security model, the scheme uses Strong Type I for type I model and Weak Type II for type II model.

3.7 Scheme 11: Dent, Libert and Paterson (concrete version)

Dent, Libert and Paterson [19] proposed concrete CL-PKE scheme that is secure and resist both Strong type I and II attacks. The scheme doesn't require random oracle model. The scheme uses bilinear map groups in its algorithms. It is an extended version of Boyen, Mei and Waters [20] scheme and applies its concepts.

Algorithms:

The algorithm of this scheme comprises the following:

- **Initialization setup:** Using bilinear map groups, this process generate both *mpk* and *msk* for KGC.
- **Partial private key generation**: The input of this process is ID, *mpk* and a random value generated in initialization setup process to generate the D_{ID}.
- **Secret value setup**: This process generates a random value as the secret value X_{ID} taking as input mpk.
- **Private key**: It is generated by taking as input the generated X_{ID}, D_{ID} and *mpk*.
- **Public key**: It is generated taking as input the *mpk* and user X_{ID}.

- **Encrypt:** The encryption process in this scheme takes the message, user's public key, user's ID and *mpk* to generate an encrypted message.
- **Decrypt:** The decryption process in this scheme takes the encrypted message, user's private key and *mpk* to generate decrypted message.

Performance

The author claimed that the public keys and encrypted messages are relative short which make the scheme perform better.

Security Model

The scheme doesn't require random oracle model. It uses Strong Type I for type I model and Strong Type II for type II model.

Table 4. Comparison of Relative Performance between Proposed Schemes

Scheme 1	Relative Performance Comparison	Scheme 2
Al-Riyami 1/Yum-Lee	Is almost equal to	Al-Riyami 2
Al-Riyami 1/Yum-Lee	Is less efficient than	Al-Riyami 3
Al-Riyami 2	Is less efficient than	Al-Riyami 3
Al-Riyami-Paterson 1	Is less efficient than	Al-Riyami-Paterson 1
Al-Riyami 1/Yum-Lee	Is less efficient than	Baek, Safavi-Naini and Susilo
Libert–Quisquater (concrete)	Is more efficient than	Shi and Li
Libert–Quisquater (generic 1)	Is almost equal to	Libert–Quisquater (generic 2)
Libert–Quisquater (generic 1)	Is less efficient than	Libert–Quisquater (generic 3)
Libert–Quisquater (generic 2)	Is less efficient than	Libert–Quisquater (generic 3)

Table 5. Comparison of Certificateless Cryptography Schemes

Scheme	Type of Scheme	Type I security model	Type II security model	Secure?
Al-Riyami 1/Yum-Lee [9][21]	Generic	Weak Type Ib	Weak Type II	No [10][16]
Al-Riyami 2 [9]	Generic	No proof available	No proof available	No
Al-Riyami 3 [9]	Generic	No proof available	No proof available	No [16]
Al-Riyami-Paterson 1 [5][9]	Concrete	Strong Type I	Weak Type II	Yes
Al-Riyami-Paterson 2 [9][13]	Generic	Strong Type I	Weak Type II	No[11][16][6]
Baek, Safavi-Naini and Susilo Certificateless [17]	Concrete	Strong Typi	Weak Type II	Yes
Libert–Quisquater (Concrete)	Concrete	Strong Typi	Weak Type II	Yes
Libert–Quisquater (generic 1)	Generic	Strong Typi	Weak Type II	Yes
Libert–Quisquater (generic 2)	Generic	Strong Typi	Weak Type II	Yes
Libert–Quisquater (generic 3)	Generic	Strong Typi	Weak Type II	Yes
Dent, Libert and Paterson	Concrete	Strong Typi	Strong Type	Yes

4. Conclusion

In this paper, we have presented all version of CL-PKE schemes proposed by by Al-Riyami, Baek, et al and Libert–Quisquater, We have given a comparative study on performance and security of the proposed scheme. From this survey, we can identify the drawbacks of various CL-PKE schemes. This help to construct an efficient and secure scheme that reduces the vulnerability.

Acknowledgements

We would like to thank to our supervisor Dr. Jayaprakash Kar for his valuable suggestions and comments that helped improving this work. This support is greatly appreciated.

References

[1] R. L. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-key Cryptosystems", Commun ACM, vol. 21, no. 2, **(1978)** Feb., pp. 120–126.
[2] W. Diffie and M. Hellman, "New Directions in Cryptography", IEEE Trans Inf Theor, vol. 22, no. 6, **(1976)**, pp. 644–654.
[3] A. Shamir, "Identity-Based Cryptosystems and Signature Schemes", in Advances in Cryptology, G. R. Blakley and D. Chaum, Eds. Springer Berlin Heidelberg, **(1984)**, pp. 47–53.

[4] A. W. Dent, "A survey of certificateless encryption schemes and security models", Int. J. Inf. Secur., vol. 7, no. 5, **(2008)** May, pp. 349–377.
[5] S. S. Al-Riyami and K. G. Paterson, "Certificateless Public Key Cryptography", in Advances in Cryptology - ASIACRYPT 2003, C.-S. Laih, Ed. Springer Berlin Heidelberg, **(2003)**, pp. 452–473.
[6] Z. Zhang, D. S. Wong, J. Xu and D. Feng, "Certificateless Public-Key Signature: Security Model and Efficient Construction", in Applied Cryptography and Network Security, J. Zhou, M. Yung, and F. Bao, Eds. Springer Berlin Heidelberg, **(2006)**, pp. 293–308.
[7] S. Goldwasser, S. Micali and R. L. Rivest, "A Digital Signature Scheme Secure Against Adaptive Chosen-message Attacks", SIAM J Comput, vol. 17, no. 2, **(1988)** Apr., pp. 281–308.
[8] X. Huang, Y. Mu, W. Susilo, D. S. Wong and W. Wu, "Certificateless Signature Revisited", in Information Security and Privacy, J. Pieprzyk, H. Ghodosi, and E. Dawson, Eds. Springer Berlin Heidelberg, **(2007)**, pp. 308–322.
[9] S. Al-Riyami, Cryptographic schemes based on elliptic curve pairings. University of London, **(2004)**.
[10] D. Galindo, P. Morillo and C. Ràfols, "Breaking Yum and Lee Generic Constructions of Certificate-Less and Certificate-Based Encryption Schemes", in Public Key Infrastructure, A. S. Atzeni and A. Lioy, Eds. Springer Berlin Heidelberg, **(2006)**, pp. 81–91.
[11] B. Libert and J.-J. Quisquater, "On Constructing Certificateless Cryptosystems from Identity Based Encryption", in Public Key Cryptography - PKC 2006, M. Yung, Y. Dodis, A. Kiayias, and T. Malkin, Eds. Springer Berlin Heidelberg, **(2006)**, pp. 474–490.
[12] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes", in Crypto, vol. 99, **(1999)**, pp. 537–554.
[13] S. S. Al-Riyami and K. G. Paterson, "CBE from CL-PKE: A Generic Construction and Efficient Schemes", in Public Key Cryptography - PKC 2005, S. Vaudenay, Ed. Springer Berlin Heidelberg, **(2005)**, pp. 398–415.
[14] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing", in Advances in Cryptology — CRYPTO 2001, J. Kilian, Ed. Springer Berlin Heidelberg, **(2001)**, pp. 213–229.
[15] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", in Advances in Cryptology, G. R. Blakley and D. Chaum, Eds. Springer Berlin Heidelberg, **(1984)**, pp. 10–18.
[16] Z. Zhang and D. Feng, "On the security of a certificateless public-key encryption (2005)", Available Fro M Httpeprint Iacr Org2005426 Proof Lemma, vol. 2.
[17] J. Baek, R. Safavi-Naini and W. Susilo, "Certificateless Public Key Encryption Without Pairing", in Information Security, J. Zhou, J. Lopez, R. H. Deng, and F. Bao, Eds. Springer Berlin Heidelberg, **(2005)**, pp. 134–148.
[18] Y. Shi and J. Li, "Provable Efficient Certificateless Public Key Encryption", IACR Cryptol. EPrint Arch., vol. 2005, **(2005)**, p. 287.
[19] A. W. Dent, B. Libert and K. G. Paterson, "Certificateless encryption schemes strongly secure in the standard model", in Public Key Cryptography–PKC 2008, Springer, **(2008)**, pp. 344–359.
[20] X. Boyen, Q. Mei and B. Waters, "Direct chosen ciphertext security from identity-based techniques", in Proceedings of the 12th ACM conference on Computer and communications security, **(2005)**, pp. 320–329.
[21] J. Kar, "Provably Secure Identity-Based Aggregate Signcryption Scheme in Random Oracles", *International Journal of Network Security*, Taiwan, vol. 17, no.5, **(2015)**, pp. 580-587.
[22] J. Kar, "Deniable Authentication Protocol based on Discrete Logarithms and Integer Factorization Problems", *ICIC Express Letter*s, Japan, 2012. vol. 7, no. 7, **(2013)** July, pp. 2061-2067.
[23] D. H. Yum and P. J. Lee, "Generic Construction of Certificateless Encryption", in Computational Science and Its Applications – ICCSA 2004, A. Laganá, M. L. Gavrilova, V. Kumar, Y. Mun, C. J. K. Tan, and O. Gervasi, Eds. Springer Berlin Heidelberg, **(2004)**, pp. 802–811.
[24] J.Kar, "Non-interactive Deniable Authentication Protocol using generalized ECDSA Signature Scheme", *International Journal of Smart Home*. Korea, vol. 5, no. 4, **(2011)** Oct., pp. 39-49.
[25] J. Kar & D. M. Alghazzawi, "On Construction of Signcryption Scheme for Smart Card Security", *IEEE International Conference on Intelligence and Security Informatics (IEEE ISI 2015),* US, **(2015)**, pp. 109-113.

Authors

Yaqoob S. Ikram has completed bachelor of science (IT) and pursuing M.S in Information Technology in Faculty of Computing & Information Technology, King Abdul-Aziz University, Kingdom of Saudi Arabia, in 2013. His research interests include cloud security, host-based intrusion detection systems and improved agile models for software development. Mr.Yaqoob worked as a developer in different government project at Kingdom of Saudi Arabia and other countries.

Isaac Sayid has completed bachelor of science (IT) and pursuing M.S in Information Technology in Faculty of Computing & Information Technology, King Abdul-Aziz University, Kingdom of Saudi Arabia, in 2013. His current research interests include Instrustion detection and Prevention system and Cryptography.

Jayaprakash Kar has received his M.Sc and M.Phil in Mathematics from Sambalpur University, M.Tech and Ph.D in Computer Science (Cryptographic Protocols) from Utkal University, India. Currently he is working as Assistant Professor in the Department of Information Systems, Faculty of Computing and Information Technology. He is actively associated with Information Security Research Group, King Abdulaziz University, Saudi Arabia. His current research interests is in development and design of provably secure cryptographic protocols and primitives using Elliptic Curve and Pairing based Cryptography