International Journal of Engineering & Technology, 10 (1) (2021) 57-63

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper



Design and development of a secure certificateless proxy signature based (SE-CLPS) encryption scheme for cloud storage

Ms. K. Sudharani [1] *, Dr. N. K. Sakthivel [2]

[1] Reserach Scholar, Bharathiyar University, Coimbatore, Tamilnadu, India
[2] Vice Principal, Nehru College of Engineering and Research Centre, Pampady – 680 588, Kerala, India
**Corresponding author E-mail: ksudharani.shagthi@gmail.com*

Abstract

Certificateless Public Key Cryptography (CL-PKC) scheme is a new standard that combines Identity (ID)-based cryptography and traditional PKC. It yields better security than the ID-based cryptography scheme without requiring digital certificates. In the CL-PKC scheme, as the Key Generation Center (KGC) generates a public key using a partial secret key, the need for authenticating the public key by a trusted third party is avoided. Due to the lack of authentication, the public key associated with the private key of a user may be replaced by anyone. Therefore, the ciphertext cannot be decrypted accurately. To mitigate this issue, an Enhanced Certificateless Proxy Signature (E-CLPS) is proposed to offer high security guarantee and requires minimum computational cost. In this work, the Hackman tool is used for detecting the dictionary attacks in the cloud. From the experimental analysis, it is observed that the proposed E-CLPS scheme yields better Attack Detection Rate, True Positive Rate, True Negative Rate and Minimum False Positives and False Negatives than the existing schemes.

Keywords: *Certificateless Encryption Scheme; Cloud; GOMAKA Framework; Hackman Tool.*

1. Introduction

PKC is used in a wide range of security applications by providing convenient key exchange services and digital signature based authentication services. There arises a need to authenticate the public keys belonging to the user to deal with the impersonation attack. A trusted authority is introduced to prevent the impersonation attack [1]. Following approaches are developed to prevent this attack

- Certificate-based PKC [2]
- Identity-based PKC [3]
- Certificateless PKC (CL-PKC) [4]

In the certificate-based PKC scheme [2], a digital security certificate assures the legitimacy of a public key. A trusted Certification Authority (CA) generates a signature for combining the public key with the key owner. However, many digital certificates have to be verified in a trust chain to obtain an authentic and valid public key. This results in the increase in the communication and computational overheads. As it requires an infrastructure to issue and manage the certificates, certificate-based PKC is not suitable for the resource-controlled mobile devices. In the Identity (ID)-based PKC scheme [3], the public keys are derived from the user ID to assure the authenticity of the public key. There is no need for the digital security certificates. A Trusted Authority (TA) derives the private keys of the users from the public keys. As the TA knows the private keys of users, there are chances of impersonation and unauthorized decryption of the ciphertexts. This is called as key-escrow problem. It is undesirable for most of the security-based applications.

In the Certificateless PKC (CL-PKC) [5-10], a KGC generates a partial private key for users according to their identities. The private key is generated from its secret value associated with the public key and the partial private key. The public key can be self- authenticated without requiring a certificate, as the user ID is used with its public key. The users can generate their own key pairs. The TA could not figure out the private key corresponding to the public key of user, as the secret value is not computable from others. CL-PKC cryptographic scheme removes the key escrow issues. This scheme requires high computational cost as it depends on the bilinear pairing scheme. Due to their large computational and communication overheads, these schemes are not well suitable for the mobile devices.

In our previous research work, a new SE-CLPS scheme is proposed. It solves the data security issues in the existing CLPS scheme by resisting the public key replacement and malicious KGC attacks. It offers high data security and requires minimum computational cost [11]. In this work, the Hackman tool is used for detecting the dictionary attacks in the cloud. GOMAKA framework incorporates various mathematical models to enhance the security and optimize the execution time of the cryptographic algorithms. The SE-CLPS scheme is used for securing the data in the cloud storage. The proposed scheme is provably secure in the standard model. The proposed cryptographic

Copyright © Ms. K. Sudharani, Dr. N. K. Sakthivel. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

scheme yields better performance in terms of computation cost and size of public key, when compared with the cryptographic schemes without random oracles.
The remaining sections in the manuscript are systematized as follows: Section II provides a brief summary of the existing certificateless encryption schemes. Section III describes about the preliminaries of the CL-PKE scheme, GOMAKA framework and Hackman tool. Section IV shows the comparative analysis result of the proposed encryption scheme with the existing Trust-based, Price-based MaxMin and Fair allocation schemes. The concluding statements of the proposed work are stated in Section V.

2. Related works

He et al. [10] proposed a CL Public Auditing (CLPA) scheme for the cloud-assisted Wireless Body Area Network (WBAN). From the security analysis, it is proven that the proposed CLPA scheme is highly secure against the Type-I and Type-II adversaries in the certificateless cryptography environment. The proposed CLPA scheme yields better security performance than the existing scheme. Balakrishnan and Raj [12] developed a secure email system based on the CLPKC scheme for public key exchange. The message is encrypted using a symmetric key generated from a secret value, the public and private keys of both the sender and receiver. The proposed system is highly secure against standard security model. This avoids a man-in-the-middle attack to obtain details of encryption/decryption key. Hassouna et al. [13] intro- duced an integrated hierarchal certificateless scheme with a level- three TA by combining the traditional Public Key Infrastructure (PKI) hierarchy and the certificateless technology. This integrated scheme is independent from the scalability and certificate management issues. The proposed scheme yields better security and key management features than the CL-PKI scheme. An improved Al-Riyami-Paterson framework is devised by adding random oracle model or trapdoor hash functions in the standard model. The basic technique for achieving the high data security depends on the improved security model for the certificateless schemes. High security is achieved by applying the AP framework to the traditional certificateless scheme.
He et al. [14] proposed a CL Provable Data Possession (CL-PDP) scheme for the smart grid applications in the cloud environment. The proposed CL-PDP scheme is secure and requires low computational cost. Yao et al. [15] developed an Elliptic Curve Cryptography (ECC)-based CL-PKC scheme for mobile devices. This scheme does not require a security certificate to prove the legitimacy of a public key. As this scheme is constructed on the ECC instead of the bilinear pairing, it is lightweight and requires minimum energy consumption. The proposed scheme avoided the key escrow issue and achieved robust security than the existing PKC scheme. Hence, it is highly suitable for the mobile devices. Padma et al. [16] proposed a CL remote authentication protocol with efficient key revocation for the large-scale WBAN. This protocol is computationally efficient and highly secure against the forgery attacks than the existing cryptographic schemes. Gondake et al. [17] formulated a mediated CL-PKE (mCL-PKE) scheme for secure data sharing in the network. A semantic encryption algorithm is used for encrypting the data items and the encrypted items are uploaded to the network. This scheme solved the key escrow and certificate revocation issues. The mCL-PKE scheme supports immediate key revocation and assures high data confidentiality. Jia et al. [18] developed a CL Signature Scheme (CLSS) for improved data security in the Internet of Things (IoT) devices. The CLSS scheme cannot resist the public key replacement attacks. The forgeability of the proposed signature scheme is proven against the super adversaries in the random oracle model. From the experimental results, it is observed that the efficiency of the proposed scheme is higher than the Yeh's scheme without requiring more computational and communication costs. He et al. [19] introduced a privacy-preserving CL-PDP scheme for the cloud storage. This scheme addressed certificate management and key escrow issues and ensured better data privacy protection. Ma et al. [20] designed a novel secure CL-PKE with multiple keywords scheme for Industrial Internet of Things (IIoT) data in the cloud server. The data security of the proposed scheme is found to be maximum against Type I and Type II adversaries. High computational efficiency is achieved with minimum communication cost.
Karati et al. [21] proposed a novel CLSS using ECC scheme that does not require bilinear pairing scheme. This CLSS scheme is highly secure against the Type-I and Type-II adversaries. Islam et al. [22] devised a CL multi-signature scheme for better data security against the adaptive chosen message and identity attacks. Li et al. [23] created a CLSS with the authority trust level. A CL homomorphic signature scheme and a public auditing scheme are proposed in this work. Thus, the security of the CLSS scheme is proven in the random oracle model. The trustworthiness and adequacy of the cloud computing environment are improved. Dhon- gade et al. [24] formulated a new approach to assure high confidentiality level of the data stored in public networks while enforcing data access control requirements using the Advanced Encryption Standard (AES)-128-256 scheme. The computational over-head is reduced significantly, as the proposed approach does not depend on the pairing-based operation. Also, this scheme does not suffer from the key escrow issues, as the KGC resides in the public network.
Charati and Ingle [25] proposed an efficient certificateless encryption scheme with Hash-based Message Authentication Code Secure Hash Algorithm 1 (HMAC SHA1) signature for secure data sharing and verification in public cloud. Our augmented scheme requires the information owner to encode the encryption key just once and to provide some additional data to the cloud so that approved clients can decode the data using their private keys. Srinivasan and Rangan [26] created a secure CL proxy re-encryption scheme by extending the PKI for achieving data security. Zhou et al. [27] introduced a security model for the certificateless signcryption schemes. The proposed scheme is found to be efficient and secure against the passive KGC attacks under the Gap Bilinear Diffie–Hellman (GBDH) and Computational Diffie-Hellman (CDH) assumptions. Zhou et al. [28] presented a new CL signcryption scheme without requiring any bilinear pairing scheme. The proposed scheme is more efficient and secure than the existing signcryption schemes without requiring more computational cost and ciphertext length.

2.1. Drawbacks of CL-PKC schemes

- In the CL-PKC scheme, a secure channel is required for the delivery of the partial private keys.
- Key revocation is a potential issue.
- This scheme does not attain complete security, since the TA may cheat.
- Certificateless cryptographic scheme does not provide a mechanism to inform about the expiry and invalidity of the key. This is a major issue for the practical deployment of this scheme.

3. Proposed work

3.1. CL-PKE scheme

Al-Riyami and Paterson [4] developed a CL-PKC scheme by integrating the partial private key and a secret value chosen by the user.

Setup: The inputs for the CL-PKE scheme are a security parameter 'k' and the system parameters 'P' and a secret master key 'msk' are obtained as output. The system parameters are freely available to all users [29].
Set private key: It considers the system parameters and identity 'ID' as input and outputs the secret value of the user 〖SK〗_ID. Each user executes this algorithm.
Set public key: It returns the public key 〖PK〗_ID of the user by considering the system parameters and secret value of the user.
SI Key Extraction: Each user performs registration of the own identity and public key to the KGC. The KGC verifies the knowledge of the user about the private key corresponding to the public key. The KGC considers the system parameters, secret master key and user ID as input and generates a SI-key corresponding to the ID required during the decryption time by the Security Intermediary (SI). The KGC executes this algorithm for each user. It is assumed that the SI key is securely distributed to the SI.
Encryption: The inputs for the encryption process are system parameters, user ID, 〖PK〗_ID and a message 'M' and either a ciphertext 〖CT〗_ID or a symbol ⊥ indicating the encryption failure is obtained as output. Any entity can execute this algorithm.
SI Decryption: It considers the system parameters, SI key, ciphertext and returns a partial decrypted message 〖C'〗_ID for the user or a symbol ⊥ indicating the decryption failure. The SI only can execute this algorithm using the SI-key.
User Decryption: It considers the system parameters, 〖SK〗_ID, 〖C'〗_ID and returns a fully decrypted message 'M' or a symbol ⊥ indicating the decryption failure. Only the user can execute this cryptographic algorithm using own private key and partial decrypted message by the SI.
The data owner stores the sensitive data to be shared only with the authorized users, in the cloud. The owner requests the cloud to partially decrypt the encrypted data, when the users requesting the confidential data from the owner. The cloud consists of three main stages such as encrypted data storage, KGC and SI. The cloud is trusted to perform the security intermediary service and generate the key correctly. But, this scheme is not trusted for the data confidentiality, as it creates the key escrow problem. The CL-PKE scheme ensures efficient key generation and management in the untrusted cloud environment, without creating any key escrow issue. The KGC cannot acquire the private keys of the users [29]. Figure.1 shows the overall flow diagram of the CL- PKE scheme. The main phases of this cryptographic scheme are stated below

3.1.1. Cloud setup

The KGC executes the Setup process of the CL-PKE scheme and generates the master key 'MK' and parameters 'P'. It is to be noted that the setup operation is a one-time task.

3.1.2. User registration

Initially, each user generates own private key and public keys by using the SetPrivateKey and SetPublicKey operations. The KGC generates two partial keys such as SI-key and U-key and a public key KGC-key for the data user. The SI-key is stored at the SI in the cloud environment and U-key is given to the user. The KGC- key comprises the public keys generated by the users and KGC. The KGC-key is used for encrypting data. The partial private key and the public key for the user 'i' are denoted as 〖SI-key〗_i, 〖U-key〗_i and 〖KGC-key〗_i respectively. The SI-key, U-key and SK are used together for the data decryption.

3.1.3. Data encryption and uploading

The data owner obtains the keys of the user from the KGC and performs symmetrical encryption of each data item, to which the data access control policy is implemented using a random session key 'K'. The owner can encrypt the key 'K' by using the KGC keys and uploads the encrypted key and access control list to the cloud. The access control list signed by the data owner is stored in the SI.

3.1.4. Data recovery and decryption

When a user wants to read data, a request is sent to the SI to obtain the partially decrypted data. The SI checks the presence of user in the access control list and availability of the KGC-key encrypted data of the user in the cloud storage. If the verification is successful, the SI retrieves the data from the cloud and performs partial decryption of the retrieved data using the SI-key for the user. The partial data decryption at the SI reduces the computational load on the users. The user decrypts the data completely, using the SK and U-key.
To enhance the efficiency of the CL-PKE scheme, the SI stores the partially decrypted data in the cloud storage, after completing the primary partial decryption for each user. If a user is revoked, the owner updates the access control list at the SI, to deny the future access requests from the revoked user. During the addition of a new user, the owner performs data encryption using the public key of the user and uploads the data and updated access control list to the cloud. The existing users are not affected by the user revocation and addition of new users [11].

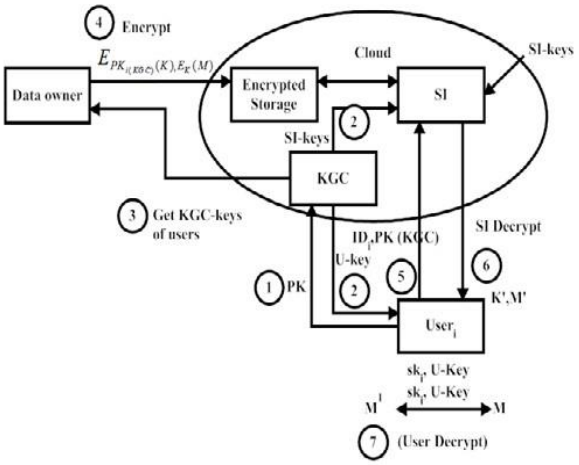


Fig. 1: Overall Flow Diagram of CL-PKE Scheme.

3.2. GOMAKA framework

GOMAKA framework comprises five packages such as application, maths, codec, io, and util. If a user proposes any model, the algorithm of the corresponding model is submitted as an application through the application package in the form of a word document. The Extended Lexical Analyzer (ELA) reads and converts the word document into a code in the type of Pre-Compiled Header (PCH) file stored in Block Cipher Cryptographic Class (BCCC). In the GOMAKA framework comprises various integrating modules such as Rivest-Shamir-Adleman (RSA) algorithm, ElGamal algorithm, etc. If a user wants to encrypt the message, the algorithm is stored in the BCCC. The maths package is used for generating random number, large arithmetic operations, etc. The io package is used for storing the plaintext file, encrypted file, etc. The util package holds the All Block Cipher (ABC) Universal Hackman tool that reads the encrypted data file and uses the dictionary attack for measuring the data security level.

3.3. Hackman tool

The hackman.exe is the core application that supplies a set of the plaintext files to the application and decodes each encoded file for all plaintext file. Hackman has inbuilt hacking mechanism designed with a set of attacks. These attacks can be enabled or disabled with the help of BCCC.LIB. The depth of the attack can be changed by the programmer. But, the default level is the maxi- mum strength hacking. If one plans to evaluate the strength of a cryptography algorithm, then it is strongly recommended NOT TO CHANGE the default settings. The Hackman tool are used for the following attacks

- Default Protocol Attack
- Brutal Attack
- Selective Predictive Dictionary Attack
- Hash reverse Attack
- Sinkhole Attack
- Bitwise Brutal Attack
- Security Protocol Attack
- Dictionary Attack
- Wormhole Attack
- Rapid Tentative Attack

3.4. Attack model

Dictionary attack is the way of hacking a secure computer or data server through the systematic entry of each word in a dictionary as a password to decrypt a message or file. Dictionary attacks are barely successful against the cloud systems that use multiple-word phrases and arbitrary combinations of the uppercase and lowercase letters mixed up with the numbers. Susceptibility to the password attacks or decryption-key attacks can be reduced by controlling the number of attempts allowed within a given time period and choosing the password or key.

4. Performance analysis

$$TPR=TP/(TP+FN) \quad (1)$$

$$TNR=TN/(FP+TN) \quad (2)$$

$$FPR=FP/(FP+TN) \quad (3)$$

$$FNR=FN/(FN+TP) \quad (4)$$

$$Accuracy=(TP+TN)/(FP+FN+TP+TN) \quad (5)$$

The performance metrics are required for evaluating the proposed approach and determining whether the network traffic belongs to the normal or attack category. The True Negative Rate (TNR) denotes the amount of normal machines that are correctly recognized as attackers

in the cloud. It denotes the accuracy of the at- tack detection model that distinguishes the normal machines and attackers. The True Positive Rate (TPR) represents the number of machines that are correctly recognized as attackers. False Positive Rate (FPR) is computed as the ratio of the number of incorrect positive detection results to the total number of negative detection results. The best FPR value is 0.0 and the worst value is 1.0.

- True Positive (TP) is the number of attacks that are correctly classified as attacks.
- False Positive (FP) is the number of normal machines that are incorrectly classified as attacks.
- True Negative (TN) is the number of normal machines that are correctly classified as normal machines.
- False Negative (FN) is the number of attackers that are incorrectly classified as normal machines.

Table 1: System Specifications

Parameter	Value
Number of physical hosts	1
System architecture	x64
Operating system	Windows
Number of VMs	10, 20, 30, 40, 50
Number of CPU cores per VM	4
CPU speed per VM	1000 MIPS
RAM memory per VM	16 GB
Hard drive storage per VM	976:56 GB
Network bandwidth share per VM	50000 Kbit/s

Table I shows the system specifications. Figure.2 shows the comparative analysis of the attack detection rate for the proposed E-CLPS scheme and existing CLPS scheme. The proposed approach yields maximum attack detection rate than the existing CLPS schemes. Figure.3 illustrates the false positive analysis of the proposed E-CLPS scheme and existing CLPS scheme. Figure.4 shows the false negative analysis of the proposed E-CLPS scheme and existing CLPS scheme. The proposed scheme achieves minimum false positive and false negative rates than the existing CLPS scheme. Figure.5 depicts the true positive analysis of the proposed E-CLPS scheme and existing CLPS scheme. Figure.6 illustrates the true negative analysis of the proposed E-CLPS scheme and existing CLPS scheme. From the graphs, it is observed that the proposed approach yields high true positive and true negative rates than the existing schemes.

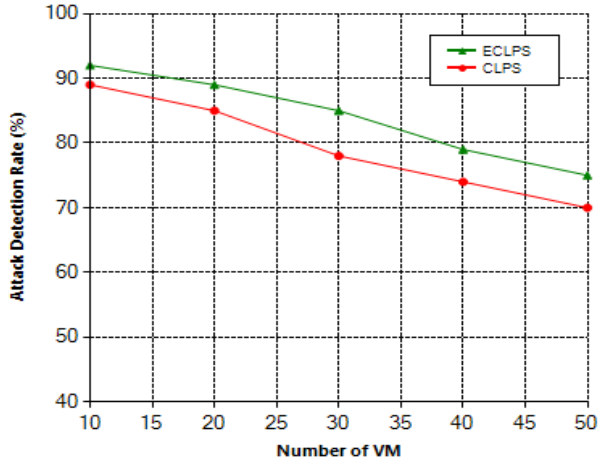


Fig. 2: Attack Detection Rate of Proposed E-CLPS and Existing CLPS Schemes.

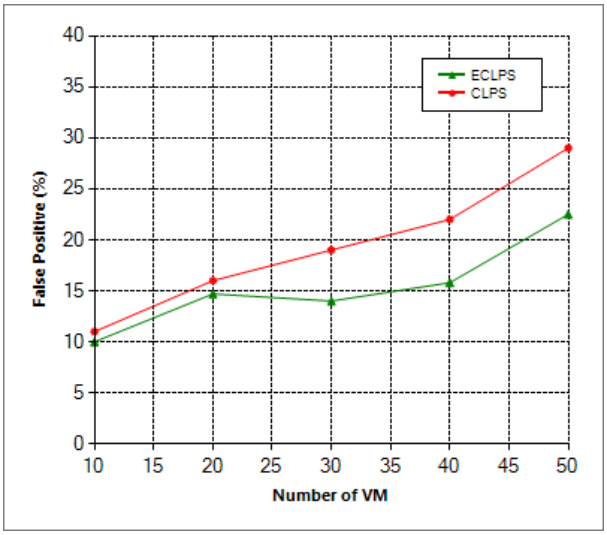


Fig. 3: False Positive Analysis of the Proposed E-CLPS and Existing CLPS Scheme.

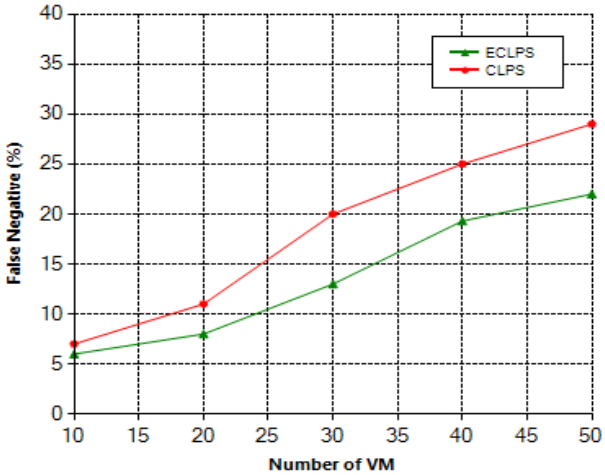


Fig. 4: False negative analysis of the proposed E-CLPS and existing CLPS schemes

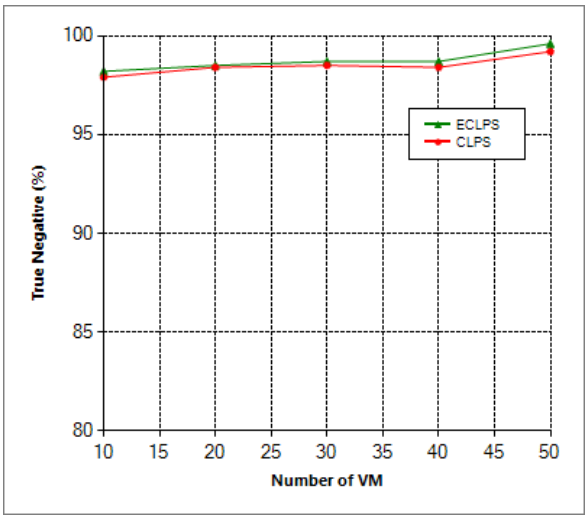


Fig. 5: True Negative analysis of the proposed E-CLPS and existing CLPS schemes

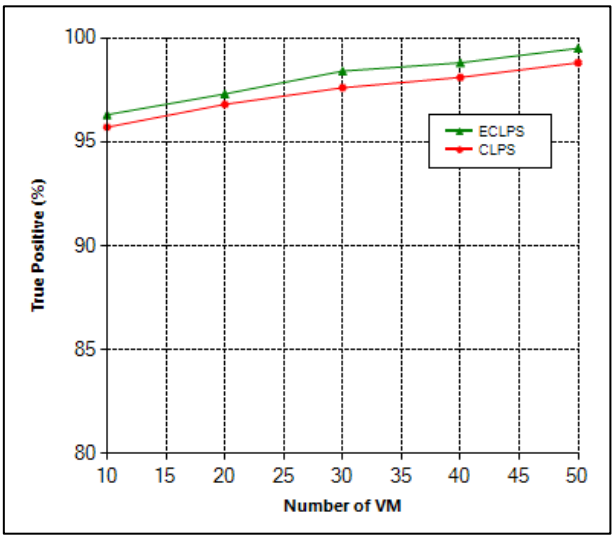


Fig. 6: True Positive Rate of the Proposed E-CLPS and Existing CLPS Schemes.

5. Conclusion

In our previous work, a new secure encryption scheme based on the Certificateless Proxy Signature (SE-CLPS) is proposed. It solves the security flaws in the existing CLPS scheme by resisting the public key replacement and malicious KGC attacks. It offers high security

guarantee and requires minimum computational cost. In this work, the Hackman tool is used for detecting the dictionary attacks in the cloud. GOMAKA framework incorporates various mathematical models to enhance the security and optimize the execution time of the cryptographic algorithms. The proposed E-CLSPS approach is compared with the existing CLPS scheme. The proposed E-CLPS scheme yields better attack detection rate, true positive rate, true negative rate and minimum false positives and false negatives than the existing schemes.

References

[1] Z. Cheng, L. Chen, L. Ling, and R. Comley, "General and efficient certificateless public key encryption constructions," in International Conference on Pairing-Based Cryptography, 2007, pp. 83-107. https://doi.org/10.1007/978-3-540-73489-5_6.

[2] C. Gentry, "Certificate-based encryption and the certificate revocation problem," in International Conference on the Theory and Applications of Cryptographic Techniques, 2003, pp. 272-293. https://doi.org/10.1007/3-540-39200-9_17.

[3] A. Shamir, "Identity-based cryptosystems and signature schemes," in Workshop on the theory and application of cryptographic techniques, 1984, pp. 47-53. https://doi.org/10.1007/3-540-39568-7_5.

[4] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in International Conference on the Theory and Application of Cryptology and Information Security, 2003, pp.452-473. https://doi.org/10.1007/978-3-540-40061-5_29.

[5] M. Toorani, "Certificateless Public-Key Cryptography," 2011.

[6] F. Wang and Y. Zhang, "A new provably secure authentication and key agreement mechanism for SIP using certificateless public-key cryptography," Computer Communications, vol. 31, pp. 2142-2149, 2008. https://doi.org/10.1016/j.comcom.2008.01.054.

[7] M. Barbosa and P. Farshim, "Certificateless signcryption," in Proceedings of the 2008 ACM symposium on Information, computer and communications security, 2008, pp. 369-372. https://doi.org/10.1145/1368310.1368364.

[8] M. Luo, Y. Wen, and H. Zhao, "An enhanced authentication and key agreement mechanism for SIP using certificateless public-key cryptography," in The 9th International Conference for Young Computer Scientists, 2008. ICYCS 2008., 2008, pp.1577-1582. https://doi.org/10.1109/ICYCS.2008.311.

[9] I. Memon, M. R. Mohammed, R. Akhtar, H. Memon, M. H. Memon, and R. A. Shaikh, "Design and implementation to authentication over a GSM system using certificate-less public key cryptography (CL-PKC)," Wireless personal communications, vol. 79, pp. 661-686, 2014. https://doi.org/10.1007/s11277-014-1879-8.

[10] D. He, S. Zeadally, and L. Wu, "Certificateless public auditing scheme for cloud-assisted wireless body area networks," IEEE Systems Journal, 2015.

[11] K. Sudharani and P. Sakthivel, "A Secure Encryption Scheme Based on Certificateless Proxy Signature," in Advances in Big Data and Cloud Computing, ed: Springer, 2018, pp. 277-285. https://doi.org/10.1007/978-981-10-7200-0_25.

[12] S. K. Balakrishnan and V. J. Raj, "Practical Implementation of a Secure Email System Using Certificateless Cryptography and Domain Name System," IJ Network Security, vol. 18, pp. 99-107, 2016.

[13] M. Hassouna, B. I. Barry, and E. Bashier, "A New Level 3 Trust Hierarchal Certificateless Public Key Cryptography Scheme in the Random Oracle Model," IJ Network Security, vol. 19, pp. 551-558, 2017.

[14] D. He, N. Kumar, S. Zeadally, and H. Wang, "Certificateless Provable Data Possession Scheme for Cloud-Based Smart Grid Data Management Systems," IEEE Transactions on Industrial Informatics, vol. 14, pp. 1232-1241, 2018. https://doi.org/10.1109/TII.2017.2761806.

[15] X. Yao, X. Han, and X. Du, "A light-weight certificate-less public key cryptography scheme based on ECC," in Computer Communication and Networks (ICCCN), 2014 23rd International Conference on, 2014, pp. 1-8. https://doi.org/10.1109/ICCCN.2014.6911773.

[16] M. S. Padma, D. J. W. Wise, M. S. Malaiarasan, and M. N. Rajapriya, "Ensuring Authenticity and Revocability for Wireless Body Area Network using Certificateless Cryptography," 2016.

[17] P. Gondake, P. Khandagale, V. Tanpure, and S. Said, "Capability of Certificateless Cryptography for Secure Data Sharing Over the Network," Imperial Journal of Interdisciplinary Re- search, vol. 2, 2016.

[18] X. Jia, D. He, Q. Liu, and K.-K. R. Choo, "An Efficient Provably-Secure Certificateless Signature Scheme for Internet-of- Things Deployment," Ad Hoc Networks, 2018. https://doi.org/10.1016/j.adhoc.2018.01.001.

[19] D. He, N. Kumar, H. Wang, L. Wang, and K.-K. R. Choo, "Privacy-preserving certificateless provable data possession scheme for big data storage on cloud," Applied Mathematics and Computation, vol. 314, pp. 31-43, 2017. https://doi.org/10.1016/j.amc.2017.07.008.

[20] M. Ma, D. He, N. Kumar, K.-K. R. Choo, and J. Chen, "Certificateless searchable public key encryption scheme for industrial internet of things," IEEE Transactions on Industrial Informatics, vol. 14, pp. 759-767, 2018. https://doi.org/10.1109/TII.2017.2703922.

[21] A. Karati, S. H. Islam, and G. Biswas, "A pairing-free and provably secure certificateless signature scheme," Information Sciences, vol. 450, pp. 378-391, 2018. https://doi.org/10.1016/j.ins.2018.03.053.

[22] S. Hafizul Islam, M. Sabzinejad Farash, G. Biswas, M. Khurram Khan, and M. S. Obaidat, "A pairing-free certificateless digital multisignature scheme using elliptic curve cryptography," International Journal of Computer Mathematics, vol. 94, pp. 39-55, 2017. https://doi.org/10.1080/00207160.2015.1088148.

[23] F. Li, D. Xie, W. Gao, K. Chen, G. Wang, and R. Metere, "A certificateless signature scheme and a certificateless public auditing scheme with authority trust level 3+," Journal of Ambient Intelligence and Humanized Computing, pp. 1-10, 2017. https://doi.org/10.1007/s12652-017-0553-x.

[24] S. Dhongade, S. Bhandare, A. Davare, and R. Chandel, "An Efficient Certificateless Encryption for Secure Data Sharing Over the Network Using AES-128 and AES-256," 2015.

[25] N. Charati and M. Ingle, "An Efficient Certificateless Encryption with Signature for Secure Data Sharing and Verification in Public Clouds," International Journal of Engineering Science, vol. 13873, 2017.

[26] A. Srinivasan and C. P. Rangan, "Certificateless proxy re-encryption without pairing: revisited," in Proceedings of the 3rd International Workshop on Security in Cloud Computing, 2015, pp. 41-52. https://doi.org/10.1145/2732516.2732519.

[27] C. Zhou, W. Zhou, and X. Dong, "Provable certificateless generalized signcryption scheme," Designs, codes and cryptography, vol. 71, pp. 331-346, 2014. https://doi.org/10.1007/s10623-012-9734-y.

[28] Y. Zhou, B. Yang, and W. Zhang, "Provably secure and efficient leakage-resilient certificateless signcryption scheme without bilinear pairing," Discrete Applied Mathematics, vol. 204, pp. 185-202, 2016. https://doi.org/10.1016/j.dam.2015.10.018.

[29] S.-H. Seo, M. Nabeel, X. Ding, and E. Bertino, "An efficient certificateless encryption for secure data sharing in public clouds," IEEE Transactions on Knowledge and Data Engineering, vol. 26, pp. 2107-2119, 2014. https://doi.org/10.1109/TKDE.2013.138.

[30] O. A. Wahab, J. Bentahar, H. Otrok, and A. Mourad, "How to distribute the detection load among virtual machines to maximize the detection of distributed attacks in the cloud?," in IEEE International Conference on Services Computing (SCC),, 2016, pp. 316-323. https://doi.org/10.1109/SCC.2016.48.

[31] W. Wang, B. Liang, and B. Li, "Multi-resource fair allocation in heterogeneous cloud computing systems," IEEE Transactions on Parallel and Distributed Systems, vol. 26, pp. 2822-2835, 2015. https://doi.org/10.1109/TPDS.2014.2362139.

[32] G. Wei, A. V. Vasilakos, Y. Zheng, and N. Xiong, "A game- theoretic method of fair resource allocation for cloud computing services," The journal of supercomputing, vol. 54, pp. 252-269, 2010. https://doi.org/10.1007/s11227-009-0318-1.

[33] O. A. Wahab, J. Bentahar, H. Otrok, and A. Mourad, "Optimal Load Distribution for the Detection of VM-based DDoS Attacks in the Cloud," IEEE Transactions on Services Computing, 2017.