ISSN 1598-0170 (Print)
ISSN 2287-1136 (Online)
http://www.jksii.or.kr

클라우드 컴퓨팅에서 ECC 암호를 적용한 안전한 데이터 스토리지 스킴☆

An Efficient and Secure Data Storage Scheme using ECC in Cloud Computing

은 효 춘[1] 논 티라난[2] 이 훈 재[3*]
XiaoChun Yin Non Thiranant HoonJae Lee

요 약

인터넷의 빠른 발전과 더불어 클라우드 컴퓨팅 기술은 가장 기술개발이 시급한 분야가 되고 있다. 클라우드 컴퓨팅은 고객들이 인터넷을 통하여 가상 자원을 제공받게 되며, 가장 시급하게 해결해야할 문제 중의 하나로 클라우드 스토리지를 들 수 있다. 클라우드 컴퓨팅분야의 급격한 증가는 클라우드 스토리지에서 심각한 보안문제를 불러일으키고 있다. 본 논문에서는 인터넷을 통하여 안전하게 데이터를 저장할 수 있고 보안 접근통제가 가능하고 또한 안전하지 않은 인터넷을 통하여 안전하게 다중 사용자끼리 데이터를 공유할 수 있는 스킴을 제안한다. 데이터 스토리지 보안 스킴의 효율을 높이기 위하여 ECC 암호를 데이터 보호 및 인증과정에서 적용한다.

☞ 주제어 : 안전한 데이터 스토리지, 클라우드 컴퓨팅, ECC 암호

ABSTRACT

With the fast development of internet, cloud computing has become the most demanded technology used all over the world. Cloud computing facilitates its consumers by providing virtual resources via internet. One of the prominent services offered in cloud computing is cloud storage. The rapid growth of cloud computing also increases severe security concerns to cloud storage. In this paper, we propose a scheme which allows users not only securely store and access data in the cloud, but also share data with multiple users in a secured way via unsecured internet. We use ECC for cryptography and authentication operation which makes the scheme work in a more efficient way.

☞ Keyword : Secured data storage, Cloud computing, ECC

1. BACKGROUND AND RELATED WORK

Cloud computing is set of resources and services offered through the Internet. Cloud services are delivered from data centers located throughout the world. One of the most prominent services offered by cloud computing is cloud storage. Cloud storage is simply a term that refers to online space that you can use to store your data. A more strict way, cloud storage is a service model in which data is maintained, managed and backed up remotely and made available to users over a network. Compared with hard disc storage, we can consider cloud storage as some kind of network storage, different types of storage devices in the network work together through the cluster, grid or distributed file system functionality to provide the storage space for user.

The biggest concern about cloud storage is security. With cloud storage, users store their data in multiple third party servers. Users worry that data saved on a remote storage

1 Department of Ubiquitous IT, Dongseo University, Busan, 617-716, South Korea Weifang University of Science & Technology, China
2 Department of Ubiquitous IT, Dongseo University, Busan, 617-716, South Korea.
3 Division of Computer and Information Eng., Dongseo University, Busan, 617-716, South Korea
* Corresponding author(hjlee@dongseo.ac.kr)
[Received 30 October 2013, Reviewed 5 November 2013, Accepted 21 January 2014]
☆ A preliminary version of this paper appeared in APIC-IST 2013, Aug 12-14, Jeju Island, Korea. This version is improved considerably from the previous version by including new results and features.

system are vulnerable. There's always the possibility that a hacker will find an electronic back door and access data. Hackers could also attempt to steal the physical machines on which data are stored. In another way, a disgruntled employee could alter or destroy data using his or her authenticated username and password. Cloud storage companies invest a lot of money in security measures in order to limit the possibility of data theft or corruption. Users still aren't likely to entrust their data to the cloud provider without a guarantee that they can access their data information whenever they want and no one else is able to get it. Since all the data are in plaintext format, not only during the transferring between users and cloud servers but also during data being stored on the servers, the data face security threats.

In order to securely store and access data in the cloud, many efforts have been done. Zhu et al. [1] proposed an idea to secure cloud file system with attribute based encryption. It is one of the most effective ways to manage and control file sharing in cloud. The proposed approach shows a secure file sharing scheme based on attribute controlling. A secure and practical attribute based encryption scheme pairings (CP-ABE-WP) was designed [1]. The second approach to secure cloud storage, proposed by Bernd Zwattendorfer [2], is to first secure the authentication process using eIDs. The reliable authentication mechanisms are needed. The STORK framework was proposed and moved to the cloud to apply the full cloud computing paradigm. Qin Liu et al. [3] proposed an efficient sharing of secure cloud storage services using a user hierarchy encryption. The designed scheme enables user at upper level to efficiently share the secure cloud storage services with all users at the lower level. A sender can basically specify recipients by taking the number and public keys as inputs of a hierarchical identity based algorithm. Raluca et al. [4] design a proof-based system which is critical to enabling security guarantees in cloud. It is efficient in detecting and proving violations to these properties by combining cryptographic tools in a novel way to obtain a scalable system.

Having motivated by the ideas above, we propose a scheme to build a trusted cloud storage system, which allow the user to store and access their data securely in the cloud by encrypting the data in the client side and decrypting the data after down loading from the cloud. Since the private key is owned by the user of the data, no one can decrypt the data, even though hackers can get the data through some approaches. This scheme also allows the user to share the data with the authenticated users. If the owner of the data wants to share the data with some authenticated users, the owner only needs to exchange the key with the authenticated users. This scheme can make users assure about the security of data stored in the cloud.

The rest of this paper is organized as follows. We first provide preliminaries in section 2. Then section 3 discusses the proposed scheme. Section 4 provides the implementation of this mechanism. Section 5 gives the security and efficiency analysis. Finally section 6 gives the concluding remark of the whole paper.

2. PRELIMINARIES

To facilitate our proposed scheme, the following items are briefly introduced.

2.1 Elliptic Curve Cryptography (ECC)

The elliptic curve cryptosystem [5-11] was initially proposed by Koblitz and then Miller in 1985 to design public key cryptosystem and presently, it becomes an integral part of the modern cryptography. A brief introduction of ECC is given below:

Let E/Fp denotes an elliptic curve E over a prime finite field Fp, which can be defined by

$$y^2 = x^3 + ax + b \tag{1}$$

where, $a,b \in F_p$ and the discriminate $D = 4a^3 + 27b^2 \neq 0$.

The points on E/ Fp together with an extra point O called the point at infinity used for additive identity form an additive group A as

$$A = \{(x, y) : x, y \in F_p, E(x, y) = 0\} \cup \{0\} \quad (2)$$

Let n, the order of A, is very large and it can be defined as $n \times G \bmod q = O$, where G is the generator of A. Also A be a cyclic additive group under the point addition "+"defined as $P + O = P$, where $P \in A$.

The scalar point multiplication over A can be defined as

$$tP = P + P + \cdots + P \ (t \text{ times}) \quad (3)$$

If P,Q ÎA , the addition P + Q be a point -R (whose inverse is R with only changing the sign of y coordinate value and lies on the curve) on the E/FP such that all the points P, Q and －R lie on the straight line, i.e., the straight line cuts the curve at P, Q and －R points. Note that if P = Q, it becomes a tangent at P or Q, which is assumed to intersect the curve at the point 0.

The security strength of the ECC lies on the difficulty of solving the Elliptic Curve Discrete Logarithm Problem (ECDLP) and it provides same level of security of RSA with less bit size key, which is addressed in the next sub-section.

2.2 Computational problems

Similar to the DLP problem (known as discrete logarithm problem), some computational hard problems on ECC are defined below, which have not any polynomial time algorithm.

1) Elliptic Curve Discrete Logarithm Problem (ECDLP)
Given $Q, R \in A$, find an integer $k \in F_p^*$ such that R=k.Q.

2) Computational Diffie-Hellman Assumption (CDHA)
Given $P, xP, yP \in A$, it is hard to compute .

3) Decisional Diffie－Hellman Problem (DDHP)
Given $P, aP, bP, cP \in G$ for any $a, b, c \in F_p^*$, decide whether or not $cP = abP$.

2.3 Elliptic Curve Diffie-Hellman (ECDH) and Elliptic Curve Integrated Encryption Scheme (ECIES)

Theoretical findings related to ECC cannot be used directly, as it is necessary to define data structures and procedures to manage the information. There are several immediate applications for ECC in cryptography, such as Elliptic Curve Diffie-Hellman (ECDH), Elliptic Curve Digital Signature Algorithm (ECDSA), Elliptic Curve Integrated Encryption Scheme (ECIES) etc.[12-33] In this paper, we apply ECDH and ECIES in our mechanism to provide the data encryption-decryption and key exchange; here we only introduce the ECDH and ECIES.

ECDH is an Elliptic Curve variant of the standard Diffie-Hellman algorithm. Comparing with Diffie-Hellman algorithm, it is a key agreement protocol performed using elliptical curves rather than traditional integers. This protocol allows two parties to create a secure channel for communications.

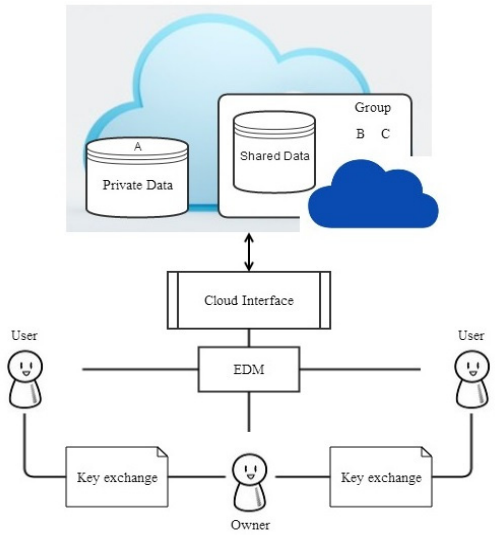
ECIES is an integrated encryption scheme based on elliptic curves that includes public key operations, encryption algorithms, authentication codes and hash computations; more precisely it uses the following functions:

1) Key Agreement (KA): Function used for the generation of a shared secret by two parties.
2) Key Derivation Function (KDF): Mechanism that produces a set of keys from keying material and some optional parameters.
3) Encryption (ENC): Symmetric encryption algorithm
4) Message Authentication Code (MAC): Data used in order to authenticate messages.
5) Hash (HASH): Digest function, used within the KDF and the MAC functions.

3. THE PROPOSED FRAMEWORK

Cloud storage is the most prominent service in cloud computing, with cloud storage, users can store and access their data at any time/anywhere, it brings much convenience to the users, however since the data are stored over the cloud and flow through the network in plaintext format,

users worry about the security of the data, although the cloud providers claim data stored in the cloud are very much of security concerns. We provide a framework, in figure 1; there are two parts for every user's data, the private part and the shared data part. In the private data part, users can store their private and sensitive data which are used only by themselves; in the shared data part, users can store the data which they want to share with other users in their group. This scheme not only allows users store and access their data securely but also allows users share data with multiple users in their group securely.



(Fig. 1) Secured Cloud Storage Framework (SCSF).

3.1 Notation

1) Owner: owner may be a person or an organization that has the data to be stored in the cloud. He has the ownership of the data and he can create a group and share the data with other users.
2) User: user also may be a person or an organization that belongs to some group and he can request to download the data. There is no obvious boundary between user and owner, if a user upload some data, he will become the owner of the data.
3) Pin number: initial number which belongs to the owner, it is a necessary condition for EDM to encrypt and decrypt the data.
4) EDM: encryption and decryption module which are integrated with Elliptic Curve Integrated Encryption Scheme. It is used to encrypt and decrypt data. It needs the owner or the user to enter the Pin number before encryption and decryption.

3.2 Owner operation

1) Authentication

Owner must be authenticated to access the service from the cloud provider. Firstly, owner logins with his username and password through the Cloud Interface and then cloud provider checks the authenticity of the owner. After being authorized, if the owner requests to upload data, cloud provider will load module (EDM) to the client end which is responsible for encryption and decryption operation.

2) EDM operation

This module is integrated with ECIES application which is used to encrypt and decrypt the data. This module requires the owner to enter an initial number (Pin number) and choose the encrypt option, and then the EDM can use this Pin number to generate the ephemeral key pair which is used for encryption and decryption.

3) Encryption

The data owner wants to store in the cloud couldn't be in plaintext format, it will face security problem. So the data are required to be transformed into cypher format in the client side, the EDM module uses the ephemeral key pair to encrypt the data that needs to store in the cloud.

4) Decryption

The data that the owner uploaded on the cloud are in ciphertext format. When the owner requests to download the data, the cloud server will send the encrypted data to the client end. If the owner can enter the Pin number correctly, the EDM module will decrypt the data.

3.2 Group user operation

1) Authentication

Same to owner, the user also must be authenticated to access the service from the cloud provider. Firstly, the user logins with his username and password through the Cloud Interface and then cloud provider checks the authenticity of the user. If the user is authorized, he can check the data which are shared in the group. We suppose the user is already in the group that the owner created.

2) Key exchange

If the user is interested in the data shared in the group, he can download the data. Since the data is in cypher format, the user couldn't decrypt the data. So the user requires checking the owner of the data and then asks the owner to get the Pin number.

We use the extended ECDH algorithm to exchange the Pin number between the user and the owner.

3) EDM operation

When the user requests to decrypt the data, the cloud server will load the EDM module to the user end, if the user wants to encrypt the data, it will ask for the Pin number.

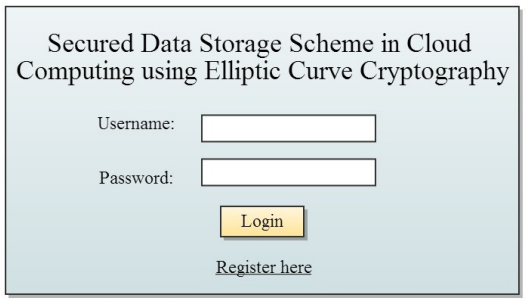
4) Decryption

After downloading the data and getting the Pin number, the user can enter the Pin number and choose the decrypt option, and then the EDM module will decrypt the data.

4. IMPLEMENTATION

4.1 Cloud Interface

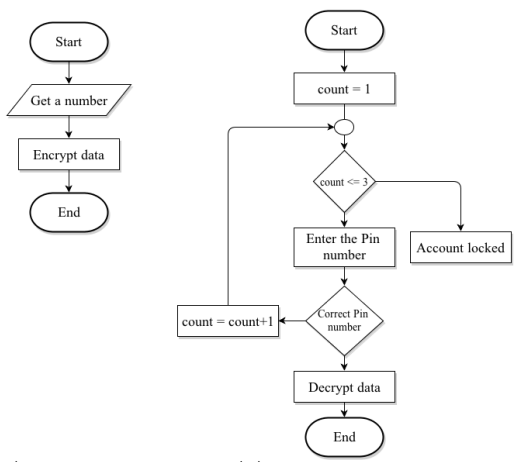
In our scheme, the owner must be authenticated to access the service of the cloud. The commonly used security mechanism for data access is username and password pair. The owner provides the username and password pair to the cloud server through cloud interface which is in the figure 2.



(Fig. 2). Cloud interface login.

4.2 EDM encryption and decryption

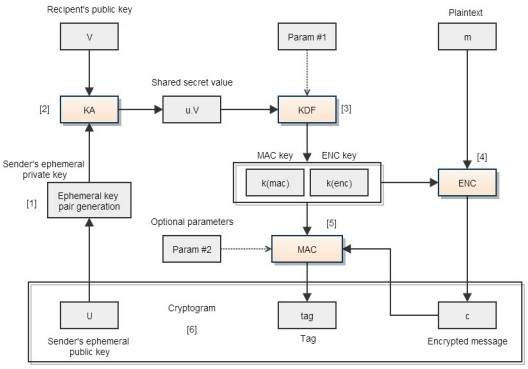
After receiving the login information, the cloud server will check the authenticity of the owner. If the owner is authorized, cloud server will load the cryptographic module (EDM) to the client end which will handle the encryption and decryption process. In the figure 3, we provide the flowchart of the EDM.



(1) Encryption (2) Decryption

(Fig. 3) EDM flowchart.

In order to encrypt the data, the user needs to input the Pin number which is a 4-digit decimal number. The encryption function will encrypt the data depending on the Pin number. The pin number is owned by the owner, even the cloud provider does not know, it can ensure the security and privacy of the data. After encrypting, the data will be uploaded on to the private data part. When the owner wants to access the data, he can download the data and decrypt



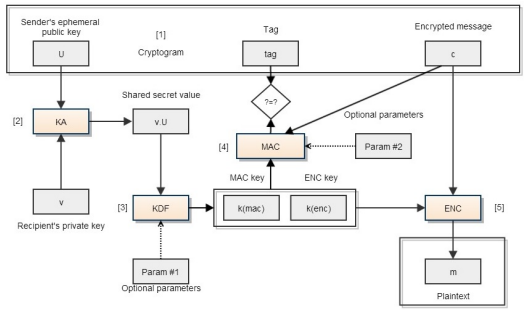
(Fig. 4) EDM encryption diagram.

the data with the same Pin number. In order to defend the Brute-force attack, we account the attempts of inputting Pin number. If the user does not correctly input in three times, the account will be locked.

The user will not be able to input within 24 hours. If the user can input the Pin number correctly, he/she can go to the next step to choose the option of encryption and decryption which is integrated with the ECIES.

Figure 4 represents a graphic description of the EDM encryption process, including the elements and functions involved in the procedure. The steps that must be taken in order to complete the encryption of the data are:

1) The sender must create a pair of temporary keys. The temporary private key will be denoted as u, and the temporary public key as U.
2) The sender will use the key agreement function, KA, in order to create a shared secret value, which is the product of the sender's temporary private key and the recipient's public key, V.
3) The sender will take the shared secret value (and optionally other parameters as input data for the key derivation function, denoted as KDF. The output of this function is the concatenation of the encryption key, k_{ENC}, and the MAC key, k MAC.



(Fig. 5) EDM decryption diagram.

4) The sender will encrypt the clear message, m, using the ENC symmetric algorithm and the encryption key k_{ENC}. The encrypted message will be represented a sc.
5) Taking the encrypted message and the MAC key (and optionally other parameters), the sender will use the selected MAC function in order to produce a tag.
6) Finally, the sender will take the temporary public key, the encrypted message and the tag, and will send the cryptogram consisting of those three concatenated elements (U‖tag‖c) to the recipient of the message.

Figure 5 presents a graphic description of the EDM decryption process, including the elements and functions involved in the procedure. The steps that must be taken in order to complete the decryption of encrypted data:

1) After receiving the cryptogram (U‖tag‖c) from the sender, the receiver must retrieve the ephemeral public key U, the tag, and the encrypted message c, so he can deal with those elements separately.
2) Using the retrieved ephemeral public key, U, and his own private key, v, the receiver will multiply both elements in order to produce the shared secret value v·U.
3) Taking as input the shared secret value v·U and the same optional parameters that sender used, the receiver must produce the same encryption and MAC keys by means of the KDF procedure.
4) With the key k_{MAC}, the encrypted message c, and the same optional parameters used by the sender, the receiver will first compute the element tag*, and then

he will compare its value with the tag that he received as part of the cryptogram. If the values are different, the receiver must reject the cryptogram due to a failure in MAC verification procedure.

5) If the tag value generated by the receiver is the correct one, then he will continue the process by deciphering the encrypted message c using the symmetric ENC algorithm and k_{ENC}. At the end of the decryption process, the receiver will be able to access the plaintext that the sender intended to send to him.

4.3 Key exchange algorithm

If the owner wants to share the data with the users in the group, he needs to encrypt the data with the shared Pin number and exchange the Pin number with the user in his group. Based on the ECDH, we design the Key Exchange Algorithm. We suppose A is the owner of the data, B is one of the user in A's group, who is interested in A's data and wants to know the Pin number. The following algorithm is to exchange the Pin number between A and B

1: procedure
2: A publishes the elliptic curve equation and base point G in the group public domain;
3: A chooses private key K_A from the Galois field;
4: A computes its public key P_A:{K_A*G};
5: A sends P_A to B;
6: B chooses private key K_B from the Galois field;
7: B computes its public key P_B:{K_B*G};
8: B sends P_B to A;
9: A encodes Pin number as the x coordinate of Pm;
10: A generates the Cm: {Pm + K_A*P_B}:
11: A sends the Cm to B;
12: B computes Pm: {Pm+ K_A*P_B - K_B*P_A};
13: end procedure

5. SECURITY AND EFFICIENCY ANALYSIS

5.1 Security analysis

In our proposed secured cloud storage scheme, initially, user should be authenticated to Cloud Interface to ensure that the user is the legal customer. Meanwhile, if the user wants to decrypt the data, he should enter the right Pin number to the EDM module. And once the user is authenticated by the EDM module, the mutual authentication of both Cloud Interface and EDM module must also be established. Since the data are encrypted and decrypted by the EDM module in the user end, and flow through the network in ciphertext format, no one can access the data except the user, even some hackers or some employees of the cloud provider get the data through some approaches, they can't get the key and decrypt the data. Thus the proposed key exchange algorithm involves the following cryptographic operations, which are briefly shown in the section 4.3. The user and the data owner exchange the key using the shared secret. The applying of the Elliptic Curve Diffie-Hellman (ECDH) increases the difficulty of the computational hard problem.

5.2 Efficiency analysis

The proposed ECC-based cloud storage scheme is more efficient than the existing RSA-based schemes due to the following reasons:

1) Provides comparable security with small key-length: In general, it is seen that 160-bit key in ECC is equivalent in security with 1024-bit key in RSA. This is because the existing RSA based PKI uses Diffie - Hellman key exchange protocol, in which the public challenges generated with key-size is at least 1024 bits, otherwise it is assumed that RSA is compromised. On the other hand, in ECC, the public challenges are of 160 bits key length, which is not easily compromised due to the unique properties of ECC.
2) Requires less computation cost: Since the main computation carried out in ECC is the scalar point multiplication, thus it requires much lesser computation cost than RSA, which uses the most costly modular exponentiation operation. In addition, ECC uses all 160-bit operation, but RSA requires 1024-bit manipulation for comparable security.

Therefore, the proposed ECC- based scheme requires less computation cost than the existing RSA based-PKI.

6. CONCLUSION

Cloud computing is surrounded by many security issues like securing data and examining the utilization of cloud by the cloud computing vendors. As a key service of cloud computing, cloud storage is also suffering from the security concern. In this paper, we concentrate on solving the problem of data security in cloud storage. We design the SCSSF framework which defines the approach that users securely store and access data in cloud. It can ensure the security of the data in the cloud via Cryptography operations. Additionally, a key exchange algorithm is also provided which can ensue sharing the data among multiple users securely. We have used ECC for the cryptography operation because the use of ECC significantly reduces the computation cost, message size and transmission overhead over RSA as 160-bit key size in ECC provides comparable security with 1024-bit key in RSA. At last, based on the framework, we provide the implementation accordingly to study the feasibility and performance of ECC for the secure cloud storage.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (grant number: 2013-071188) and Busan Brain 21 project.

Reference

[1] Zhu, S., Yang, X., & Wu, X.. Secure Cloud File System with Attribute Based Encryption. In Intelligent Networking and Collaborative Systems (INCoS), 2013 5th International Conference on (pp. 99-102). IEEE.

[2] Zwattendorfer, B., & Tauber, A.. Secure cloud authentication using eIDs. In Cloud Computing and Intelligent Systems (CCIS), 2012 IEEE 2nd International Conference on (Vol. 1, pp. 397-401). IEEE.

[3] Liu, Q., Wang, G., & Wu, J.. Efficient sharing of secure cloud storage services. In Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on (pp. 922-929). IEEE.

[4] R. Ada Popa, J. Lorch, D. Molnar, H. J. Wang, and L. Zhuang. Enabling security in cloud storage SLAs with cloudproof. Technical Report MSR-TR-2010-46, Microsoft Research, 2010.

[5] Stallings, W, “Cryptography and Network Security: Principles and Practices”, *Prentice Hall, 4th Edition,* pp 420-430, 2009.

[6] Hankerson, D, Menezes, A, Vanstone, S, “Guide to elliptic curve cryptography”, *Springer-Verlag, New York, USA, 2004.*

[7] Koblitz, N, “Elliptic Curve Cryptosystem”, *Journal of mathematics computation,* Vol. 48, No. 177, pp203-209, 1987.

[8] Miller, V, “Use of elliptic curves in cryptography”, *Proc. of Advances in Cryptology-CRYPTO’ 85, LNCS,* Vol. 218, pp. 417 – 426, 1985.

[9] V.Miller, “Uses of elliptic curves in cryptography”, *Lecture Notes in Computer Science218: Advances in Cryptology- CRYPTO’85,* pages417-426, Springer-Verlag, Berlin, 1986.

[10] N.Koblitz, “Elliptic curve cryptosystems”, *Mathematics of Computation, 48:203-209, 1987.*

[11] Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS), Apr. 2002, http://www.ietf.org/rfc/rfc3278.txt

[12] M. Abdalla, M. Bellare, P. Rogaway, DHIES: An Encryption Scheme Based on the Diffie-

[13] Hellman Problem, *Contribution to IEEE P1363a, 1998,* http://cseweb.ucsd.edu/users/mihir/papers/dhaes.pdf.

[14] M. Abdalla, M. Bellare, P. Rogaway, The oracle Diffie-Hellman assmptions and an analysis of DHIES, *Lecture Notes in Comput. Sci.2020(2001),* 143 – 158.

[15] American National Standards Institute, Public Key Cryptography for the Financial Services Industry:

The Elliptic Curve Digital Signature Algorithm (ECDSA), 1998.

[16] V. Gayoso Martínez, L. Hernández Encinas, and C. Sánchez Ávila, *Journal of Computer Science and Engineering,* Volum 2, Issue 2, August

[17] Brainpool, ECC Brainpool Standard Curves and Curve Generation, 2005, http://www.ecc-brainpool.org/download/Domain-parameters.pdf.

[18] Bundesamt fur Sicherheit in der Information stechnik, *Elliptic Curve Cryptography, 2009,* https://www.bsi.bund.de/cln 183/EN/Home/home node.html.

[19] W. Diffie, M.E. Hellman, New directions in cryptography, *IEEE Trans. Inform. Theory 22(1976), 644-654.*

[20] Arjun Kumar, Byung Gook Lee, HoonJae Lee, Secure Storage and Access of Data in Cloud Computing, *ICT Convergence (ICTC), 2012 International Conference on Oct. 15-17, 2012*

[21] Institute of Electrical and Electronics Engineers, *Standard Specifications for Public Key Cryptography -Amendment 1: Additional Techniques, 2004.*

[22] International Organization for Standardization / International Electro technical Commission, *Information Technology - Security Techniques - Encryption Algorithms - Part 2: Asymmetric Ciphers, 2006.*

[23] N. Koblitz, Elliptic curve cryptosystems, *Math. Comp. 48(1987), 203-209.*

[24] V. Gayoso Mart´ınez, L. Hern´andez Encinas, C. S´anchez´Avila, Security and practical considerations when implementing the Elliptic Curve Integrated Encryption Scheme, preprint, 2010.

[25] V. S. Miller, Use of elliptic curves in cryptography, *Lecture Notes in Comput.Sci.218(1986), 417-426.*

[26] National Institute of Standards and Technology, *Digital Signature Standard (DSS), 2000.*

[27] J. H. Silverman, The Arithmetic of Elliptic Curves, volume 106 of Graduate texts in Mathematics, *Springer-Verlag, New York, NY, USA, 1986.*

[28] Standards for Efficient Cryptography Group, *Test Vectors for SEC 1, 1999, http://www.secg.org/download/aid-390/gec2.pdf.*

[29] Standards for Efficient Cryptography Group, *Elliptic Curve Cryptography, 2000, http://www.secg.org/download/aid-86/sec2 final.pdf.*

[30] Standards for Efficient Cryptography Group, *Recommended Elliptic Curve Domain Parameters, 2000, http://www.secg.org/download/aid-780/sec1-v2.pdf.*

[31] Yunho Lee, “An Analysis on the Vulnerability of Secure Keypads for Mobile Devices,” *Journal of Korean Society for Internet Information*, Vol.14, No.5, pp.15-21, June 2013.

[32] Young Bae Yoon, Junseok Oh, Bong Gyou Lee, “The Important Factors in Security for Introducing the Cloud Services,” *Journal of Korean Society for Internet Information*, Vol. 13, No. 6, pp.33-40, Dec. 2012.

[33] Hee Won Myeong, Jung Ha Paik, Dong Hoon Lee, “Study on implementation of Secure HTML5 Local Storage,” *Journal of Korean Society for Internet Information*, Vol. 13, No. 4, pp.83-93, Aug. 2012.

◐ 저 자 소 개 ◑

은 효 춘(XiaoChun Yin)

She received the B.S. degree in education and technology from Qufu Normal University, Qufu, China in 2004, and received the M.S. degree in education and technology from Nanjing Normal University, Nanjing, China in 2007. She had been working as a lecturer in Weifang University of Science & Technology, China from 2008 to 2012. Currently she is a doctoral candidate in cryptography and network security at Dongseo University, Korea. Her research interests include network security, cloud security, authentication protocol and real-time communication.

논 티라난 (Non Thiranant)

He received the B.S. degree in Information technology from Multimedia University, Malaysia in 2013. He is currently a master candidate in cryptography and network security at Dongseo University, Korea. His research interests include network security, cloud computing, e-Healthcare, and authentication protocol.

이 훈 재 (Hoon Jae Lee)

He received the B.S., M.S. and Ph.D. degree in Electrical Engineering from Kyungpook national university in 1985, 1987 and 1998, respectively. He had been engaged in the research on cryptography and network security at Agency for Defense Development from 1987 to 1998. Since 2002 he has been working for Department of Computer Engineering of Dongseo University as an associate professor, and now he is a full professor. His current research interests are in security communication system, side-channel attack, USN & RFID security. He is a member of the Korea institute of Information security and cryptology, IEEE Computer Society, IEEE Information Theory Society and etc