International Journal of Engineering & Technology, 7 (3.24) (2018) 321-325

International Journal of Engineering & Technology

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

Research paper

Privacy-Enhanced Deduplication Technique in Closed Circuit Television Video Cloud Service Environment

Namje Park*

**Department of Computer Education, Teachers College, Jeju National University,*
61 Iljudong-ro, Jeju-si, Jeju Special Self-Governing Province, 690-781, Korea
Corresponding author E-mail:namjepark@jejunu.ac.kr

Abstract

Background/Objectives: With the recent surging popularity of cloud services, various cloud service providers (CSP) are coming into the picture. On cloud platforms, many different forms of services are available, among which storage-based solutions are the most commonly adopted. The cloud storage market also is likely to continue to expand in the future.
Methods/Statistical analysis:Presently, quite a few cloud storage environments are already implementing the deduplication technique. Given the characteristics of cloud environments, storage servers capable of accommodating large quantities of storage are necessary. With increases in storage capacity emerges the need for additional storage solutions. Utilizing deduplication can solve the cost problem fundamentally.
Findings:The privacy problem can be extremely serious if the files in question deal with "sensitive" issues such as political orientation, ideas, particular diseases, and sexuality. When the list of the file uploaders is obtained on the cloud server, the cloud environment itself may act as a surveillance system in the future, hence the unavoidable risk.
Improvements/Applications:This paper indicates the privacy breach problem and proposes a new technique of deduplication to solve it. In the proposed technique, the map structures of the user and files are not stored on the server and thus the list of file uploaders cannot be secured through the analysis of meta-information on the server, hence the privacy of the users intact. Furthermore, the new technique can solve the issue of file ownership by using PIN (Personal Identification Number) and offers merits such as safety against insider breaches and sniffing attacks.

Keywords: *CIosed Circuit Tele Vision Video, Deduplication Technique, Security Framework, CCTV, Video Security*

1. Introduction

With the recent surging popularity of cloud services, various cloud service providers (CSP) are coming into the picture. On cloud platforms, many different forms of services are available, among which storage-based solutions are the most commonly adopted. The cloud storage market also is likely to continue to expand in the future[1,2].

Of note, the size of data in the upcoming Fourth Industrial Revolution (4IR) era is expected to increase to astronomical proportions, which will ultimately lead to cost increases resulting from the expansion of storage capacity and will be a massive burden on the service providers.

Data deduplication, therefore, is one of the extremely important core technologies in cloud environments. Deduplication refers to a technique that avoids storing the same data (duplicates) uploaded by multiple users. Using the technology has a merit that data storage capacity can be reduced to unprecedented levels[3,4].

Presently, quite a few cloud storage environments are already implementing the deduplication technique. Given the characteristics of cloud environments, storage servers capable of accommodating large quantities of storage are necessary. With increases in storage capacity emerges the need for additional storage solutions. Utilizing deduplication can solve the cost problem fundamentally.

Deduplication, however, poses a problem for privacy because of its very structure. In order to implement the technique, the map structures of the user and files are stored as metainformation. In that process, a list of the users who uploaded a certain file(s) can be obtained through the meta analysis on the server. Of note, the privacy problem can be extremely serious if the files in question deal with "sensitive" issues such as political orientation, ideas, particular diseases, and sexuality. When the list of the file uploaders is obtained on the cloud server, the cloud environment itself may act as a surveillance system in the future, hence the unavoidable risk[5,6,7].

This paper indicates the privacy breach problem and proposes a new technique of deduplication to solve it. In the proposed technique, the map structures of the user and files are not stored on the server and thus the list of file uploaders cannot be secured through the analysis of metainformation on the server, hence the privacy of the users intact. Furthermore, the new technique can solve the issue of file ownership by using PIN (Personal Identification Number) and offers merits such as safety against insider breaches and sniffing attacks.

2. Privacy Issues Inherent in Deduplication Techniques

2.1 Problems Resulting From User-To-File Mapping Structure

Basically, implementing deduplication for cloud storage will require the establishment of user-to-file mapping structure as mentioned in Chapter 2. However, in the user-to-file mapping structure lie a serious issue with privacy breaches. Looking at the mapping structure alone may not fully convey exactly what kind of privacy problem is expected. With files that are highly sensitive to the user, the user-to-file mapping structure itself can give rise to privacy breach threats for the user.

In accordance with Article 23-1 (Limitation to Processing of Sensitive Information) of the Personal Information Protection Act, the personal information controller defines sensitive information as the one that concerns ideology/beliefs, joining or leaving a labor union or a political party, political opinions, health, sexuality, etc. or that may significantly compromise the privacy of the person in question. Furthermore, Article 23-2 of the same act stipulates that necessary countermeasures must be taken to ensure the security of the sensitive information so defined.

In other words, when a file to be uploaded contains the sensitive information or the disclosure of upload details itself constitutes the sensitive-information situation, strong security measures are needed. Especially when a file containing documents in favor of a certain political party, ideology-related information, particular diseases, or sexuality is uploaded, the user-to-file mapping structure itself can act as a privacy breach during deduplication. Of note, deduplication environments are structured such that multiple users and a single copy of file are mapped, which means the server can obtain all lists of the users who uploaded certain files. This characteristic can be a serious threat to privacy. Even worse, a possibility that the cloud server may play the role of a surveillance operator cannot be ruled out.

File upload details concerning a user(s) remains strictly private for the individual in question. Even for a cloud server manager, if he/she secures all lists of the files uploaded by an individual or, conversely, the lists of the users who uploaded a particular file, such act can be regarded as an aggressive breach of privacy.

2.2 Incompatibility between Confidentiality and Deduplication

In cloud systems, it is common practice for the server to provide data confidentiality services that are based on encryption. In other words, when the client receives plaintexts, the server would normally encrypt them and keep them in storage as a way to prepare against hacking.

In such practice, when the user requests data downloads, the server will send him/her the decrypted plaintexts, meaning the server has the encryption keys and can even decrypt when necessary.

If necessary, an administrator on the cloud server's side with a high level of authority could recover the files based on the encryption keys[6,7,8,9].

As a powerful security measure to protect the privacy of users, one might consider that the client implements file encryption in advance and then uploads the information to the server. In such cases, the server would surely manage file security but the data deduplication mechanism would be lost. As shown in Fig. 3, despite the same copy of a file User A and User B having uploaded respectively, the file is encrypted with different keys so the server actually recognizes it as two different files, thus rendering deduplication infeasible. In other words, it is extremely difficult in practical terms to fully secure the confidentiality through advance encryption on the client's end and at the same time accomplish the saving of storage space using deduplication.

2.3 File Ownership Issue

The issue of file ownership and file share is related to security policy when viewed from the server's point of view. When Users A and B have each uploaded a single copy of the same file in cloud environments where deduplication is being implemented, what is actually stored therein is a single copy. In such cases, it is necessary to consider to whom the ownership of the copy belongs. Conversely, it is necessary for the download-requesting user to prove that he/she has the ownership to the file in question.

The mentioned problem can be solved by exercising strict control over the lists of the existing uploaders of certain files. But this approach would change nothing regarding the very privacy issue relating to the user-to-file mapping structure as previously mentioned. That a user exercises his/her right to a file means, conversely, the file in question is the one that the user has previously uploaded and put differently, the server can clarify who is the individual that had uploaded that particular file.

Re-phrased using an example, the server can have, through metaanalysis, a full access to the lists of the users who had uploaded a range of sensitive information such as PR materials in favor of particular political parties, information relating to particular diseases and sexuality, and materials addressing particular ideas or beliefs. Hence, any cloud environment wherein the conventional deduplication technique is implemented, is left with the privacy issue due to its own structure. Strictly speaking, all cloud services implementing the current deduplication approach fall short of airtight privacy protection measures on the grounds of the ownership issue[10,11,12,13].

2.4 Susceptibility to Insider Attacks

Cloud environments possess inherent risks of personal information exposure as posed by insiders. As testified by numerous insider-originated data security breaches in the past, insider hacking is in fact considered a huge threat to security in cloud environments.

A typical cloud environment provides the minimal amount of security through data encryption. What lies in such environment is a very real, unexcludable scenario where an insider administrator with the highest level of authority could collect the file decryption keys, and even if such administrator has no knowledge about the encryption keys, he/she could potentially crack the file-to-user mapping relationships through metadata analysis. This means all lists of the users who uploaded a certain file(s) can be secured or, conversely, all lists of the files uploaded by a certain user(s) can be obtained through metadata analysis[14].

As discussed, resolving the issues of deduplication and privacy-protection simultaneously is a daunting task in practical terms. This paper proposes a way to solve both problems concurrently.

3. Proposal of New Deduplication Technique

This chapter proposes a new technique to solve the privacy-breach problem inherent in the conventional deduplication mechanism.

3.1 Overview of the Proposed Technique

The privacy issue in the conventional deduplication applications basically arises from the inherent user-to-file mapping structure. In fact, a complete removal of such mapping structure would preclude the obtainment of the lists of users who uploaded certain files or conversely, the lists of files uploaded by certain users. Hence, this paper proposes a technique that will fundamentally sever the user-to-file list and file mapping relationships and as a result prevent the tracking of the users through metadata and file structure analysis on the server alone[15,16].

In the proposed system(in Fig. 1), users can upload or download their files using PIN (Personal Identification Number), RefID (Reference ID), and fHV. PIN is the number known only to the user; RefID is the value stored on the meta server and is composed based on the PIN information; and fHV refers to the values obtained by rehashing the hash values of files. RefID cannot be cracked by using fHV; conversely, fHV or PIN cannot be broken based on RefID.

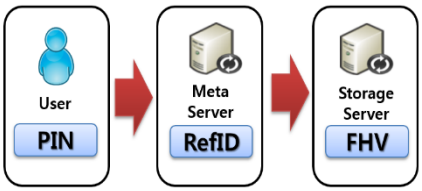

Fig. 1: Overview of the proposed scheme

In sum, no user-to-file mapping information can be obtained using the meta server- and storage server-stored RefID and fHV alone. Figuring out fHV based on RefID would mandatorily require the PIN information known only to users.

Table 1 lists the abbreviations to describe the proposed method.

Table 1: Notation

Abbreviation	Description
UserID	User's ID
SID	Session ID
PK	Pre-shared encryption key
File	File to upload / download
Filehash	The hash value of the original file
FHV	Hash value for Filehash
FilePath	The full path of the destination file
PIN	Value required to verify user ownership
RefID	Reference value stored in the meta server
$H(\cdot)^K$	Result of hashing a specific value
$E(\cdot)^K$	Result of encryption processing with key K
$D(\cdot)^K$	Result of decryption processing with key K

4.2 System Architecture

Figure 2 illustrates the architecture of the client and cloud server. From the client, access to the cloud server can be gained via various devices such as mobile devices, laptops, and PCs.

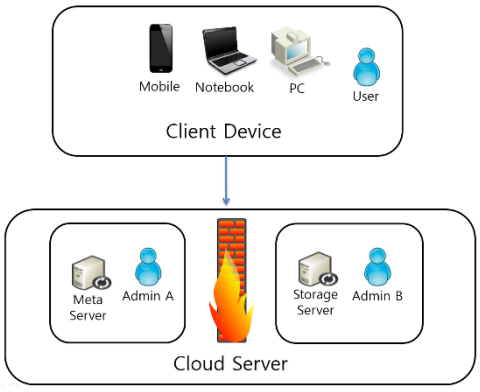

Fig. 2: System configuration

The cloud server includes the meta server and storage server, and these two must be strictly segregated.

In other words, the meta and storage server cannot be located on a single server, meaning they must be placed on different servers. Furthermore, it is advised that the meta server and storage server be split physically as well. Of note, separate administrators should be assigned to manage these two servers.

4.3 File Redundancy Check Protocol

The file redundancy check protocol refers to the phase wherein the actual presence of files on the server is verified. When the same copy of a file is found on the server, efficiency is intact because the file-uploading process is not needed (in fig. 3). The file redundancy check protocol is described below:

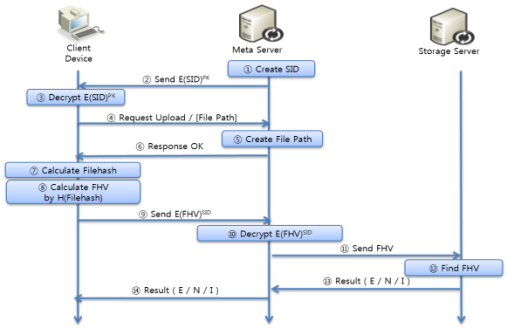

Fig. 3: File duplicate check protocol

① The meta server generates SIDs (session ID).

② The meta server encrypts the SIDs into PK that is previously shared and sends the information to the client server.

③ The client obtains SIDs through decryption.

④ The client requests uploads to a particular path.

⑤ The meta server generates the requested file upload path.

⑥ The client is notified of the completion of path generation.

⑦ The client hashes the file(s) to be uploaded and obtains the file hash values.

⑧ By rehashing the obtained hash values, the client generates fHV.

⑨ The client encrypts and sends the fHV.

⑩ The meta server obtains the fHV information through decryption.

⑪ Sending the fHV to the storage server, the meta server inquires about the existence of the file in question.

⑫ The storage server verifies the registration status of the file.

⑬ The check result is stated either as E, N, or I and is notified to the inquirer. E is short for Exist, meaning the file in question is already existing. N represents Not Exist, meaning the file is non-existent. I stands for Incomplete, indicating the file in question exists only partially.

⑭ The meta server sends the E/N/I result to the client.

Table 2 summarizes the results of file redundancy check. When E is sent, the same copy of the file is already uploaded on the server and thus no additional upload is needed. The client, therefore, can simply execute the mapping process without the uploading. Receiving N or I, on the contrary, means the upload on the server is needed and the client should accordingly carry out the upload[17,18,19,20].

Table 2: Notation

Result	Description
E	File already exists on server
N	File does not exist on the server
I	File has partially uploaded to the server

Of note, I refers to the stage where only a partial upload is

executed, hence the additional upload at this point concerns only the portions of the file that are yet to be uploaded on the server[21,22,23,24,25,26].

4.4 File Upload/Mapping Protocol

The file upload/mapping protocol is as follows: (in fig. 4).

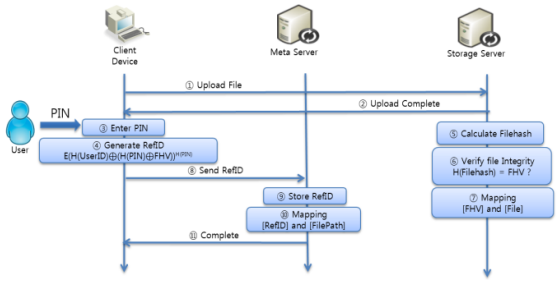

Fig. 4: File upload/mapping protocol

① Steps ① and ② apply only to the file redundancy check result of N and I. In other words, when the result comes out as E, no additional upload is needed. When the result received is N or I, file uploading on the storage server is carried out.

② The client is notified of the upload completion.

③ The user enters PIN through the client.

④ Using PIN, the client generates RefID which is created in the following equation:

$$RefID = E(H(UserID) \oplus (H(PIN) \oplus FHV))^{H(PIN)} \quad (1)$$

⑤ The storage server computes the hash values of the file.

⑥ The storage server verifies the integrity of the file received, by comparing the file hash values against the fHV values.

⑦ The fHV and uploaded file are subjected to mapping and then stored.

⑧ The client sends to the meta server the RefID generated in Step ④.

⑨ The meta server stores the received RefID.

⑩ The meta server maps the RefID and FilePath value.

⑪ The meta server notifies of the upload completion.

4.5 File download protocol

The file download protocol is as follows: (in fig. 5)[27,28,29]

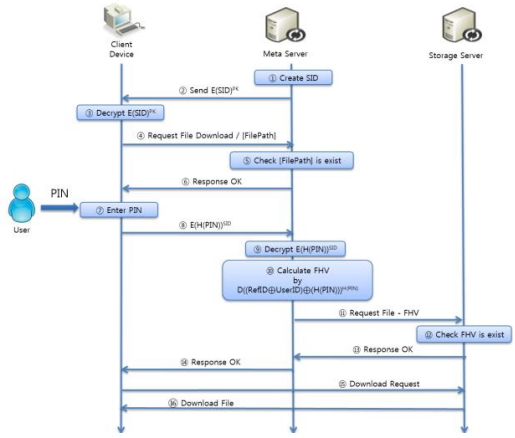

Fig. 5: File download protocol

① The meta server generates SIDs (session ID).

② The meta server encrypts the SIDs into PK and sends the information to the client server.

③ The client obtains SIDs through decryption.

④ The client requests downloads to the meta server.

⑤ The meta server verifies the existence of the file in the path concerned. If the file is non-existent, the session ends.

⑥ The meta server notifies the client of the result that the file exists in the path concerned.

⑦ The user enters PIN through the client.

⑧ The client encrypts the hash value of PIN into a SID key and sends it to the meta server.

⑨ The meta server decrypts H(PIN).

⑩ The meta server computes fHV based on the RefID and H(PIN) value using the equation below:

$$D((RefID \oplus UserID) \oplus (H(PIN)))^{H(PIN)} \quad (2)$$

⑪ The meta server sends the fHV to the storage server.

⑫ The storage server verifies if that particular file mapped to the fHV exists or not.

⑬ The storage server notifies the meta server of the result that such file exists.

⑭ The meta server notifies the client of the download-ready status.

⑮ The client requests the file download to the storage server.

⑯ The client downloads the file to the storage server to complete the process.

4. Conclusion

Data deduplication positions itself as an essential core technology in the establishment of cloud storage systems. Capable of significantly reducing the space required for data storage, the technique offers a merit of direct cost saving and as such, will likely remain a core technology in cloud computing. Despite the benefits, deduplication has an inherent problem with user-privacy breaches. The conventional deduplication method tends to focus solely on file encryption.

This research paper pointed out the problem of user privacy that arises from the user-to-file mapping structure and proposed a new technique to complement such weakness. The proposed mechanism is characterized by the storing of RefID and fHV separately on the meta server and storage server, where a relationship cannot be established between these two values when using them alone. Hence, the privacy of users can be guaranteed securely and the data confidentiality remains intact even against meta-information exposure resulting from insider attacks. Furthermore, the utilization of users' PIN information resolves the issues of file ownership and user privacy simultaneously by the new technique.

Deduplication is a vitally necessary technique in creating cloud storage and is being implemented in numbers of cloud services. To provide secure cloud services for the upcoming 4IR era, continued research should be undertaken into the privacy problem that lies inherent in deduplication technology.

Acknowledgment

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) [2017-0-00207, Development of Cloud-

based Intelligent Video Security Incubating Platform].

References

[1] J. R. Douceur, A Adya, W. J. Bolosky, P. Simon, and M. Theimer, "Reclaiming space from duplicate files in a serverless distributed file system", Distributed Computing Systems, Proceedings of 22nd International Conference on. IEEE, 14 pages, Jul. 2002.

[2] Namje Park and Namhi Kang, "Mutual Authentication Scheme in Secure Internet of Things Technology for Comfortable Lifestyle", Sensors, Vol. 16, No. 1, pp. 1-16, Dec. 2015.

[3] Donghyeok Lee and Namje Park, "Geocasting- based synchronization of Almanac on the maritime cloud for distributed smart surveillance", The Journal of Supercomputing, Vol. 73, No. 3, pp. 1103-1118, Mar. 2017.

[4] Kyungsu Park, Ji Eun Eom, Jeongsu Park, and Dong Hoon Lee, "Secure and Efficient Client-side Deduplication for Cloud Storage", Journal of the Korea Institute of Information Security & Cryptology, Vol. 25, No. 1, pp. 83-94, Feb. 2015.

[5] Bellare, Mihir, Sriram Keelveedhi, and Thomas Ristenpart, "DupLESS: Server-Aided Encryption for Deduplicated Storage", IACR Cryptology ePrint Archive, 2013.

[6] Donghyeok Lee and Namje Park, "Teaching Book and Tools of Elementary Network Security Learning using Gamification Mechanism", Journal of the Korea Institute of Information Security & Cryptology, Vol. 26, No. 3, pp. 787-797, Jun. 2016.

[7] Namje Park, Jin Kwak, Seungjoo Kim, Dongho Won, and Howon Kim, "WIPI Mobile Platform with Secure Service for Mobile RFID Network Environment", Advanced Web and Network Technologies, and Applications, LNCS, Vol. 3842, pp. 741-748, Jan. 2006.

[8] Hyun-il Kim, Cheolhee Park, Dowon Hong, and Changho Seo, "Encrypted Data Deduplication Using Key Issuing Server", Korea Information Science Society, Vol. 43, No. 2, pp. 143-151, Feb. 2016.

[9] Cheolhee Park, Dowon Hong, Changho Seo, and Ku-Young Chang, "Privacy Preserving Source Based Deduplication In Cloud Storage", Korea Institute Of Information Security And Cryptology, Vol. 25, No. 1, pp. 123-132, Feb. 2015.

[10] Namje Park and Marie Kim, "Implementation of load management application system using smart grid privacy policy in energy management service environment", Cluster Computing, Vol. 17, No. 3, pp. 653-664, Sep. 2014.

[11] J. Li, X. Chen, M., Li, J. Li, P. P. Lee, and W. Lou, "Secure deduplication with efficient and reliable convergent key management", IEEE transactions on parallel and distributed systems, Vol. 25, No. 6, pp. 1615-1625, Jun. 2014.

[12] Namje Park and Hyo-Chan Bang, "Mobile middleware platform for secure vessel traffic system in IoT service environment", Security and Communication Networks, John Wiley&Sons Ltd, Vol. 9, No. 6, pp. 500-512, Apr. 2016.

[13] Bellare, Mihir, Sriram Keelveedhi, and Thomas Ristenpart, "Message-locked encryption and secure deduplication", Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, Berlin, Heidelberg, 2013.

[14] Namje Park, Hongxin Hu, and Qun Jin, "Security and Privacy Mechanisms for Sensor Middleware and Application in Internet of Things (IoT)", International Journal of Distributed Sensor Networks, Vol. 2016, Article ID 2965438, 3 pages, 2016.

[15] Cheolhee Park, Dowon Hong, and Changho Seo, "A Secure and Practical Encrypted Data De-duplication with Proof of Ownership in Cloud Storage", Korea Information Science Society, Vol. 43, No. 10, pp. 1165-1172, Oct. 2016.

[16] Namje Park, "UHF/HF Dual-Band Integrated Mobile RFID/NFC Linkage Method for Mobile Device-based Business Application", The Journal of The Korean Institute of Communication Sciences, Vol. 38, No. 10, pp. 841-851, Oct. 2013.

[17] Namje Park, "Design and Implementation of Mobile VTS Middleware for Efficient IVEF Service", Journal of KICS, Vol. 39C, No. 6, pp. 466-475, Jun. 2014.

[18] Kaaniche, Nesrine, and Maryline Laurent, "A secure client side deduplication scheme in cloud storage environments", New Technologies, Mobility and Security (NTMS), 2014 6th International Conference on. IEEE, pp.1-7, Mar. 2014.

[19] P. Puzio, R. Molva, M. Onen, and S. Loureiro, "ClouDedup: secure deduplication with encrypted data for cloud storage", Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on. Vol. 1. IEEE, pp. 363-370, Dec. 2013.

[20] Namje Park, Jungsoo Park, and Hyoungjun Kim, "Inter-Authentication and Session Key Sharing Procedure for Secure M2M/IoT Environment", International Information Institute(Tokyo) Informa- tion, Vol. 18, No. 1, pp. 261-266, Jan. 2015.

[21] Donghyeok Lee and Namje Park, "A Secure Almanac Synchronization Method for Open IoT Maritime Cloud Environment", Journal of Korean Institute of Information Technology Vol. 15, No. 2, pp. 79-90, Feb. 2017.

[22] Ma, Jingwei, Gang Wang, and Xiaoguang Liu., "DedupeSwift: Object-Oriented Storage System Based on Data Deduplication", Trustcom/Big Data SE/I SPA, 2016 IEEE, pp.1069-1076, Aug. 2016.

[23] Tang, Yang, and Junfeng Yang, "Secure Deduplication of General Computations", USENIX Annual Technical Conference, pp.319-331, Jul. 2015

[24] Young-Jun Yoo, Sun-Jeong Kim, and Young Woong Ko, "Cloud File Synchronization Scheme using Bidirectional Data Deduplication", Journal of KIIT, Vol. 12, No. 1, pp. 103-110, Jan. 2014.

[25] Byung Kwan Kim, Young Woong Ko, and Kwang Mo Lee, "Performance Enhancement for Data Deduplication Server Using Bloom Filter", Journal of KIIT. Vol. 12, No. 4, pp. 129-136, Apr. 2014.

[26] Donghyeok Lee and Namje Park, "Electronic identity information hiding methods using a secret sharing scheme in multimedia-centric internet of things environment", Personal and Ubiquitous Computing, Vol. 22, No. 1, pp. 3-10, Feb. 2017.

[27] Donghyeok Lee, Namje Park, Geonwoo Kim, and Seunghun Jin, "De-identification of metering data for smart grid personal security in intelligent CCTV-based P2P cloud computing environment", Peer-to-Peer Networking and Applications, DOI 10.1007/s12083-018-0637-1, pp. 1-10, Mar. 2018.

[28] Namje Park, "The Core Competencies of SEL-based Innovative Creativity Education", International Journal of Pure and Applied Mathematics, Vol. 118, No. 19, pp. 837-849, Jan. 2018.

[29] Namje Park, "STEAM Education Program : Training Program for Financial Engineering Career", International Journal of Pure and Applied Mathematics, Vol. 118, No. 19, pp. 819-835, Jan. 2018.