International Journal of Security and Its Applications
Vol.8, No.2 (2014), pp.193-202
http://dx.doi.org/10.14257/ijsia.2014.8.2.20

Secure Streaming Media Data Management Protocol

Jeong-Min Do[1] and You-Jin Song[2*]
[1,2]*Dongguk University*
[1]*havdrim@hotmail.com*, [2]*song@dongguk.ac.kr*

Abstract

This paper is intended to solve the problem of high capacity of streaming media and data access privilege management in a broadcasting environment. For this, an access privilege sharing management technique that enables a safe streaming media service is designed by using AONT-based XOR threshold Secret Sharing on the basis of cloud computing. Specifically, encryption in protocol is composed of the first encryption that maintains the confidentiality of data by performing the AES encryption of streaming media data, and the second encryption that makes an access privilege management function carried out and satisfies variability and redundancy by the distribution storage of AES key. The proposed scheme assures security against a collusion attack between the malicious users and cloud servers due to the decryption privilege sharing because the first encryption and the second encryption output (header, body) is stored and shared through other channels (Privilege Manager Group, Media Service Provider) respectively.

Keywords: *AONT(All Or Nothing Transform), XOR threshold Secret Sharing, XOR Share, Recovery Share, Access Privilege Management*

1. Introduction

Recently, the consumers of broadcasting contents want to watch a clear picture. The size of media data is becoming larger in order to improve the quality of media contents in consideration of needs in the consumer market. An efficiency problem is compelled to be considered in order to apply continuously increasing high-capacity media contents to the streaming service. Studies using cloud computing technology are in active progress in order to solve this efficiency problem. Typically, there is a solution such as Hadoop and MapReduce that solves the problem of high capacity, that is to say, big data, by making a distributed processing system[1-3].

The big data problem occurred as data was increased in geometric progression, due to the development of sensor and internet. Big data is characterized by 4Vs adding up high volume, high variety, high velocity, and value. That is to say, big data doesn't only mean a increase in volume, but also is good-quality data that contains useful information. This big data can be used for satisfying the various requirements of consumers with correct service[2, 3].

Besides, customer information leakage examples in the company database by outsiders or personal information leakage examples by malicious insiders occurred, and therefore an issue of security for personal information stored in the external storage space is brought up. An encryption technique is proposed in order to solve this security problem[4], but there is no access privilege management technique that provides a safe

* Corresponding Author

ISSN: 1738-9976 IJSIA

streaming service for the user who has fair contents use privilege in the cloud computing environment. The threat of information leakage is increasing, as the volume of data of various sorts such as media and medical information shared in the cloud computing environment is increasing.

That is to say, a safe protocol design is necessary in order to solve the high capacity problem of streaming media in the broadcasting environment and the problem of managing the privilege that a user with fair privilege accesses the contents. This paper approaches the high capacity problem in the cloud computing environment where necessary computing resources can be used as much as required. Besides, it is intended to solve the problem of managing the privilege of access to streaming media contents by using a cryptographic technique.

The proposed scheme makes ciphertext (body) by encrypting streaming media data with AES[5], and forms Recovery Share (header) and XOR Share (owned by service user) by the AONT-based XOR Threshold Secret Sharing [6] of AES key. The formed body, header, and XOR Share is transmitted to the cloud provider, Privilege Manager Group, and service user respectively, and then is stored and managed. A collusion attack between malicious attacker and cloud server can be prevented by guaranteeing the confidentiality of data through the distribution storage of header and body, and by implementing the function of access privilege management to surely get admission from Privilege Manager Group when User group accesses data.

The contents of text are as follows. In Chapter 2, the related studies forming the proposed scheme are introduced. And in Chapter 3, an encryption protocol by using AONT-based XOR threshold Secret Sharing that is the proposed scheme is designed. In Chapter 4, the characteristics of proposed scheme are analyzed. Finally, in Chapter 5, the conclusion of this paper is formed.

2. Related Work

This chapter describes the overview and the detailed algorithm of AONT-based XOR threshold Secret Sharing.

2.1. Overview of AONT based XOR Threshold Secret Sharing

AONT-based XOR threshold Secret Sharing [6] is a scheme to guarantee both variability and redundancy of data, by combining AONT [7, 8] with XOR threshold Secret Sharing [9-11].

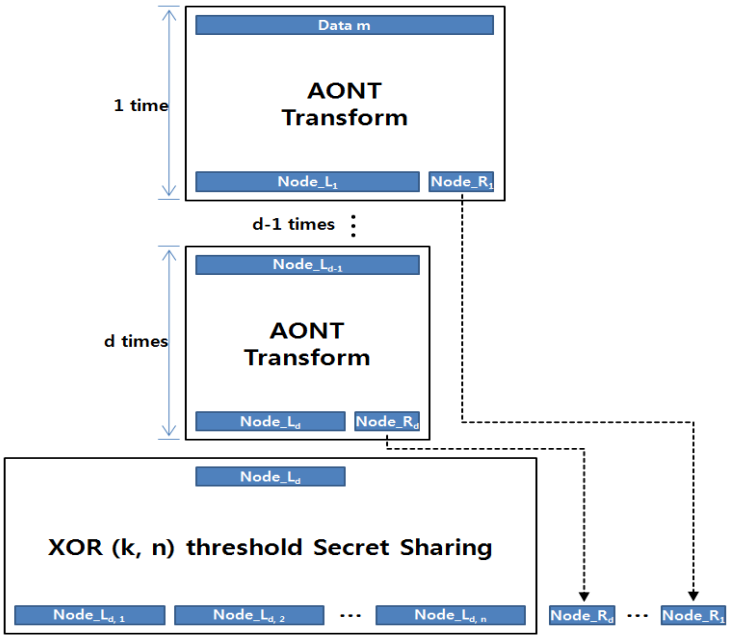

Figure 1. AONT based XOR (k, n) Threshold Secret Sharing

The size of Node_L_i(i=1,...,d-1) is made to decrease by repeatedly applying AONT to secret information like Figure 1, and then XOR threshold Secret Sharing is applied to Node_L_d (share after AONT transformation is performed d times). It is designed so as to obtain variability to split data in the required block size by applying AONT transformation to secret information and redundancy in preparation for data loss by applying XOR threshold Secret Sharing to Node_L_d.

A point to which attention should be paid is the application of Recovery Share and XOR Share. Recovery Share is a share generated in case of AONT transformation, and is an essentially necessary share in case of recovery (Node_R_1, Node_R_2, ..., Node_R_d). And XOR Share is a share generated by XOR threshold Secret Sharing, and is a share (Node_$L_{d,1}$, Node_$L_{d,2}$,..., Node_$L_{d,n}$) that has redundancy (if there are $k$shares, then the loss of the remaining shares doesn't matter).

In the protocol of proposed scheme, Recovery Share is owned by Privilege Manager Group, and XOR Share is owned by User. When User wants to read data, the recovery of Node_L_d to k XOR Shares, and the recovery of AES key by receiving the Recovery Share from Privilege Manager Group is utilized as an access admission procedure (access privilege management).

The specific examples of this application are shown in Figure 2.

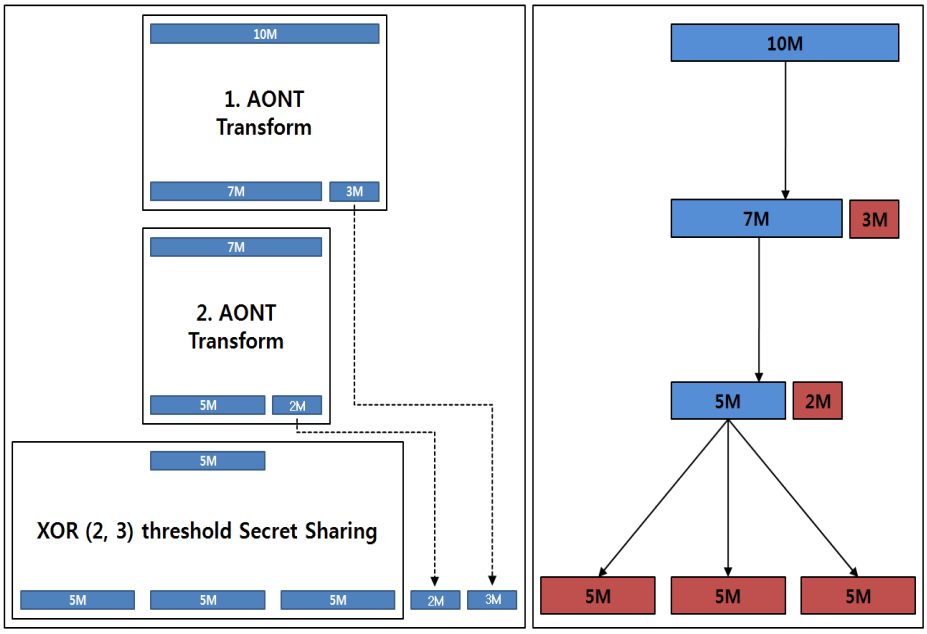

Figure 2. AONT based XOR $(2,3)$ Threshold Secret Sharing

Plaintext of 10M (Mega byte) is splitted by AONT transformation. In the first execution, 10M is splitted into 7M and 3M. And in the second execution, 7M is splitted into 5M and 2M. And it is shared out to 3 XOR Shares by applying $(2,3)$ threshold Secret Sharing to 5M. And enable recovery to its original condition if there are just 2 XOR Shares.

If plaintext of 10M is shared only by XOR $(2,3)$ threshold Secret Sharing, then storage space of 30M is necessary. By contrast, if AONT-based XOR $(2,3)$ threshold Secret Sharing is used, then storage space of 20M is necessary. Besides, if AONT is repeatedly applied, then it is possible to more reduce the volume of the share distributed by Secret Sharing. That is to say, storage space is effectively utilized, and security against data loss is also guaranteed.

2.2. The Detailed Algorithm of AONT based XOR Threshold Secret Sharing

① System Setup: Step to set up a system

· $h : \{0,1\}^{l(s-1)} \rightarrow \{0,1\}^{l}$ is hash function, $g : \{0,1\}^{l} \rightarrow \{0,1\}^{l(s-1)}$ is generator function

· l is the size of a block, s is the number of blocks

② Encryption: Step to encrypt data m with AES

· Form a ciphertext C by encrypting data m with AES.

③ Key Splitting: Step to split AES key by AONT-based XOR (k,n) threshold Secret Sharing

a. AONT Transformation

· Split AES key k into $k_1, k_2, \dots, k_s (k_i \in \{0,1\}^l, i = 1, \dots, s)$ in order to perform AONT transformation.

· Calculate $\mu_s = h(k_1||k_2|| \dots ||k_{s-1})$ by using a hash function h.

· Calculate $g(\mu_s \oplus k_s)$ by using a generator function g for $\mu_s \oplus k_s$.

· Calculate $x_1||x_2|| \dots ||x_{s-1}$ (Node_L_i(i=1,...,d)) by XOR operation of $k_1||k_2|| \dots ||k_{s-1}$ and $g(\mu_s \oplus k_s)$.

$$x_1||x_2|| \dots ||x_{s-1} = (k_1||k_2|| \dots ||k_{s-1}) \oplus g(\mu_s \oplus k_s)$$

· Calculate $h(x_1||x_2|| \dots ||x_{s-1})$ by using a hash function h

· Calculate x_s x_s (Node_R_i) by the XOR operation of $h(x_1||x_2|| \dots ||x_{s-1})$ and $(\mu_s \oplus k_s)$.

$$x_s = (\mu_s \oplus k_s) \oplus h(x_1||x_2|| \dots ||x_{s-1})$$

· Perform the AONT transformation of Node_L_i d times.

· Store all Node_R_i(i=1,...,d) (Recovery Share) in Privilege Manager Group after the final d-th transformation.

· The Recovery Share stored in Privilege Manager Group is transmitted in a bundle in case of requesting decryption.

b. XOR (k, n) threshold Secret Sharing (for the purpose of explanation, set k for 2, and n for 3)

· Split Node_L_d into $K_0||K_1 (K_i = \{0,1\}^l (i = 1,2))$.

· Generate 2 random numbers $R_0, R_1 (R_i = \{0,1\}^l (i = 1,2))$.

· Generate the following Node_$L_{d,i}$(i=1,2,3) by XOR operation of K_0, K_1 and R_0, R_1

Node_$L_{d,1}$=$(K_0 \oplus R_0 \oplus R_1,\ K_1 \oplus R_1)$

Node_$L_{d,2}$=$(K_0 \oplus R_0,\ K_1 \oplus R_0 \oplus R_1)$

Node_$L_{d,3}$=$(R_0,\ R_1)$

· Transmit Node_$L_{d,i}$ to each user.

④ Decryption: Step to recover AES key by AONT-based XOR (k, n) threshold Secret Sharing, to decrypt a ciphertext C by the recovered ASE key, and to derive data m

a. XOR secret recovery

· User collects threshold k Node_$L_{d,i}$(i=1,2,3), and then Node_L_d is recovered as follows, through XOR operation (Use Node_$L_{d,1}$ and Node_$L_{d,2}$, for the purpose of explanation)

$$K_0 = K_0 \oplus R_0 \oplus R_0$$
$$K_1 = K_1 \oplus R_0 \oplus R_1 \oplus R_0 \oplus R_1$$

·Form Node_L_d=$(K_0||K_1)$

b. AONT inverse transformation

· User is authenticated by Privilege Manager Group, and receives all Node_R_i(i=1,...,d), and the process of AONT inverse transformation is performed d times, and then AES key is recovered.

c. Decryption of ciphertext C

· Decrypt a ciphertext C with AES key recovered by AONT-based XOR threshold Secret Sharing.

3. The Proposed Scheme

This chapter describes a protocol to encrypt media data by using AONT-based threshold Secret Sharing.

3.1. Overview

Data Owner want that only User who have fair privilege should use their media data. The proposed scheme provides a function to grant the privilege of access to media data after a credible institution judges whether a consumer is a fair user in spite of the condition that Data Owner is absent.

Encryption in protocol is composed of the first encryption that maintains the confidentiality of data by performing the AES encryption of streaming media data, and the second encryption that makes an access privilege management function carried out and satisfies variability and redundancy by the distribution storage of AES key. In addition, the proposed scheme assures security against a collusion attack between the malicious users and cloud servers due to the decryption privilege sharing because the first encryption and the second encryption output (header, body) is stored and shared through other channels (Privilege Manager Group, Media Service Provider) respectively.

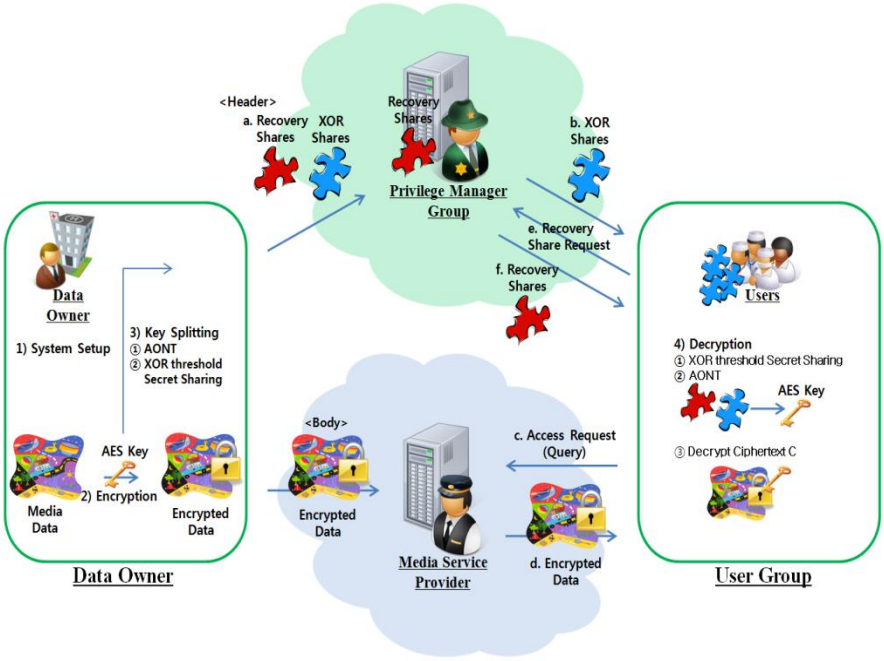

Figure 3. The Protocol of Proposed Scheme

The protocol of proposed scheme is composed of 4 groups (Data Owner, User, Privilege Manager Group, Media Service Provider) and the function of each group.

Data Owner is a group such as media data producer that practically owns the copyright of media data, and plays a role in forming a header and body through encryption in order to provide media data for User. User is a user or potential user who uses Data Owner's service, and uses media data by possessing XOR Share and obtaining admission from Privilege Manager Group.

Media Service Provider is a cloud provider group that owns data storage server only for media data, and manages a ciphertext and the identifier of header corresponding to it. Privilege Manager Group is a credible accredited institution that judges whether User is a fair user in case of access to data, and manages Recovery Share and the identifier of XOR Share corresponding to it.

Specific encryption and access privilege management process is explained in the following chapter.

3.2. Encryption of Media Data

This chapter explains the process of media data encryption. As mentioned before, the encryption algorithm uses AES encryption algorithm and the AONT-based XOR threshold Secret Sharing of related studies.

Above all, as the first encryption, Data Owner forms encrypted streaming media data (body) by encrypting streaming media data with AES key. The same identifier is attached to the encrypted streaming media data and AES key. And a list of identifiers for each ciphertext is managed by Media Service Provider. An identifier is used for searching each secret key corresponding to the ciphertext for decryption.

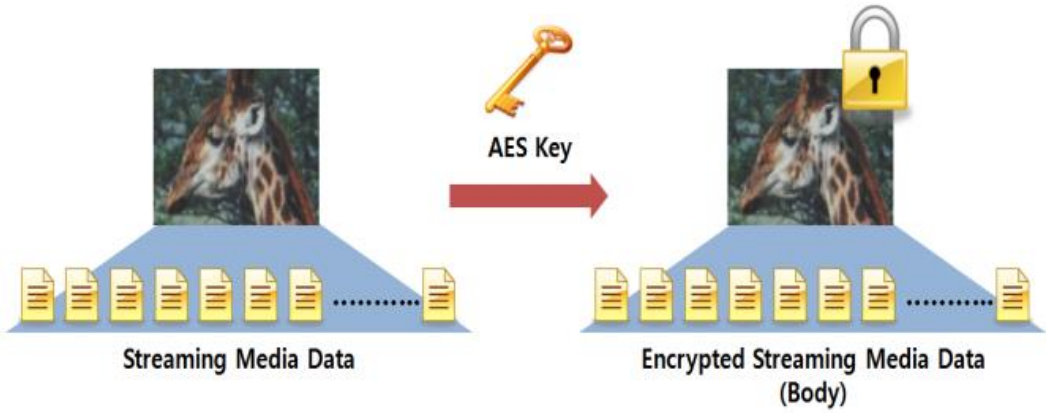

Figure 4. First Encryption

As the second encryption, Recovery Share and XOR Share is formed by the AONT-based XOR threshold Secret Sharing [6] of AES key to which an identifier is attached. The same identifier is given to XOR Share and Recovery Share. And a list of identifiers for each share is kept by Privilege Manager Group. Recovery Share is stored in Privilege Manager Group. And XOR Share is transmitted to User. A point that the minimum privilege to recover AES key is given only if XOR Shares (the same identifier) more than threshold decided in XOR threshold Secret Sharing are gathered becomes a clue to make Privilege Manager Group judge whether to transmit Recovery Share.

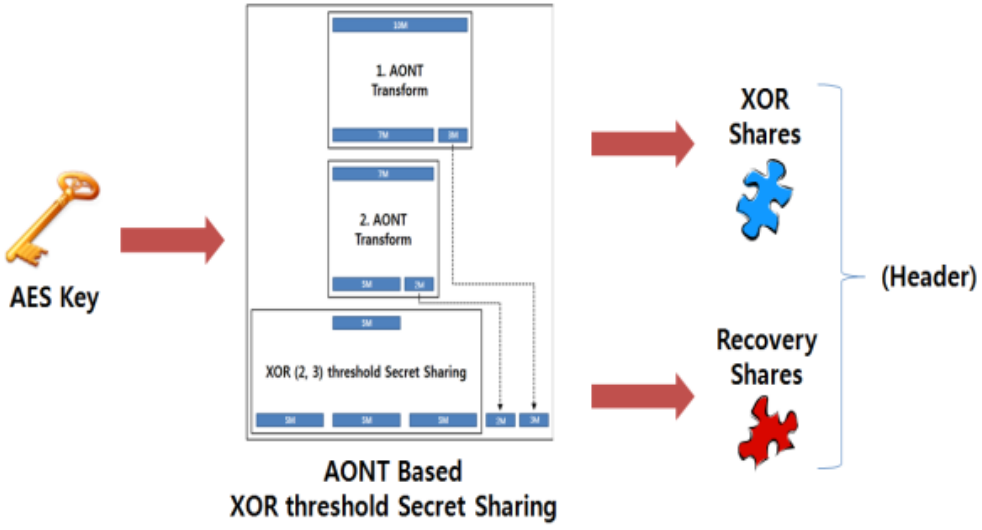

Figure 5. Second Encryption

The output of encryption is XOR Share, Recovery Share, and encrypted streaming media data. The privilege to save and manage this data is given to User, Privilege Manager Group, and Media Service Provider. It is safe against a collusion attack because data can be used only after their consent is given by all of User, Privilege Manager Group, and Media Service Provider.

3.3. Decryption of Media Data

Data Owner encrypts streaming media data with AES, and splits·shares AES key by AONT-based XOR threshold Secret Sharing. A ciphertext encrypted with AES is transmitted to Media Service Provider, and all Recovery Share splitted by the AONT-based XOR threshold Secret Sharing of AES key is transmitted to Privilege Manager Group, and XOR Share is distributed to User. If User wants to use media data, a fair use is authenticated (access admission) by Privilege Manager Group through XOR Share (the same identifier) more than threshold decided in the XOR threshold Secret Sharing, and all Recovery Share (Recovery Share corresponding to the identifier of XOR Share) is acquired. AES key is acquired by performing the process of AONT-based XOR threshold Secret Sharing inverse transformation. And the encrypted streaming media data corresponding to the identifier of AES key is received and decrypted.

4. Analysis

4.1. Collusion Attack of Media Service Provider and Privilege Manager Group

The proposed scheme shares and stores header and body that is decryption privilege. Besides, data is encrypted with AES. And AES key is splitted shared (d Recovery Shares, n XOR Shares for Node_L_d) by AONT-based XOR threshold Secret Sharing. A ciphertext is shared out stored to Media Service Provider, d Recovery Shares to Privilege Manager Group that is a credible institution, and n XOR Shares for Node_L_d to User respectively. Privilege Manager Group authenticates a fair user, and provides decryption privilege (secret key) for ciphertext. That is to say, a procedure to authenticate User can be made, and the threat of collusion attack can be prevented by the sharing of decryption privilege.

4.2. Efficiency of AONT-based XOR Threshold Secret Sharing Calculation

The proposed scheme splits data into pieces of small size by repeatedly applying AONT to secret key, and applies XOR threshold Secret Sharing to only one share (Share except Recovery Share) among them. It is a universally acknowledged truth that XOR operation is outstanding in the aspect of efficiency. Besides, in case of independently using only Secret Sharing scheme, storage space is ineffectively required as much as a multiple of the number of shares splitted from the original data. However, if AONT technique is utilized, then the size of splitted data can be reduced, and therefore it is effective.

4.3. Redundancy of AONT-based XOR Threshold Secret Sharing

It is possible to solve the problem of lack of redundancy according to the loss of secret key possessed by User. In other words, if a secret key is lost, then it should be issued again, and a consequent problem of overload in calculation is anticipated. From the viewpoint of company, a service is suspended until the secret key is issued again, and therefore it fails to maintain business continuity and wastes time cost. The proposed scheme can recover the secret key if only threshold k XOR Shares exist among n XOR Shares for Node_L_d, which User possesses. That is to say, the possibility of recovery is decided according to the amount of k XOR Shares information required for recovering Node_L_d among n XOR Shares by utilizing XOR threshold Secret Sharing.

5. Conclusion

This paper formed a protocol to be capable of recovering the encrypted streaming media data only if a key is shared · stored by using AONT-based XOR threshold Secret Sharing, and User acquires the Recovery Share (Privilege Manager Group) and XOR Share (User) distributed by Data Owner. Besides, it guarantees security against a collusion attack through realizing the management of access privilege and the management of distribution of decryption privilege by Privilege Manager Group.

Acknowledgements

This work was supported by the MKE (Ministry of Knowledge Economy) [A004700008], Development of realistic sense transmission system with media gateway supporting multimedia and multi-device. This article is a revised and expanded version of a paper entitled [The Encryption with Access Privilege Management for Streaming Media Service] presented at International Symposium on Advanced and Applied Convergence held on November 14-16, 2013 at Seoul, Korea.

References

[1] Y. S. Min, H. Y. Kim and Y. K. Kim, "Distributed File System for Cloud Computing", Korean Institute of Information Scientists and Engineers, Communications of the Korea Information Science Society, vol. 27, no. 5, **(2009)**, pp. 86-94.
[2] D. H. You, S. H. Chung and T. H. Kim, "Korean Institute of Information Scientists and Engineers", Journal of the Korea Information Science Society, vol. 36, no. 5, **(2009)**, pp. 351-359.
[3] O. G. Min, H. Y. Kim and G. H. Nam, "Trends in Technology of Cloud Computing", Electronics and Telecommunications Research Institute, Electronics and Telecommunications Trends, vol. 24, no. 4, **(2009)**, pp. 1-13.
[4] S. C. Yu, C. Wang, K. I. Ren and W. J. Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing", INFOCOM, 2010 Proceedings IEEE, **(2010)**, pp. 321-334.
[5] J. Daemen and V. Rijmen, "AES Proposal: Rijndael", AES Algorithm Submission, **(1999)**.
[6] Y. J. Song and J. M. Do, "Distributed Access Privilege Management for Secure Cloud Business", Korea Information Processing Society, Korea Information Processing Society Review, vol. 18-C, no. 6, **(2011)**, pp. 369-378.
[7] R. L. Rivest, "All-or-nothing encryption and the package transform", Fast Software Encryption, 4th International Workshop(FSE'97), vol. 1267, **(1997)**, pp. 210-218.
[8] H. Kuwakado, "A Study for Encryption Mode to Improve Safety in Crypto System", The Telecommunications Advancement Foundation, Research report, no. 19, **(2004)**, pp. 242-251.
[9] A. Shamir, "How to Share a Secret", Communication of the ACM, vol. 22, no. 11, **(1979)**, pp. 612-613.
[10] H. Ishizu and T. Ogihara, "A Study on Long-term Storage of Electronic Data", The Institute of Electronics, Information and Communication Engineer, Proceedings of the IEICE General Conference, D-9-10, **(2004)**.
[11] Y. J. Song and K. Y. Park, "Distributed Security Management Model of a Large Amount of e-Business Data", Global e-Business Association, e-Business Studies, vol. 11, no. 1, **(2010)**, pp. 325-342.
[12] J. M. Do and Y. J. Song, "The Encryption with Access Privilege Management for Streaming Media Service", The Institute of Internet, Broadcasting and Communication, The International Symposium on Advanced and Applied Convergence, vol. 1, **(2013)**, pp. 155-158.

