International Journal of Security and Its Applications
Vol. 10, No. 9 (2016), pp.285-294
http://dx.doi.org/10.14257/ijsia.2016.10.9.27

TEAM: Virtual Synchronized File-based Transparent and Privacy-Enhanced Storage System

Hye-Lim Jeong and Ki-Woong Park*

Dept. of Information Security, Sejong University, Seoul, Korea
woongbak@sejong.ac.kr

Abstract

Under the revised personal information security law in Korea, users may face inconvenience with managing privacy files. This paper proposes a privacy protection system, called TEAM, enhanced with a 'Virtual Synchronized File' concept, which keep convenience by providing users with a transparent user experience, while complying with the personal information security law. The proposed system periodically detects and compresses files containing personal information and encrypts them. The encrypted files are transmitted to a remote storage server. Then the client PC replaces the original privacy file with a Virtual Synchronized File termed VSF. From the users' perspective, the VSF alleviates the inconveniences and maximizes efficiency of personal information management as it (1) allows the users to access to the privacy files in a transparent manner in comparison to an access scheme for a normal document file; (2) internally performs encryption/decryption for the file; and (3) separately stores the files with minimized users' interventions. Consequently, TEAM makes it possible to comply with the law while providing users with transparent user experience.

Keywords*: Privacy protection system, File access control, File virtualization*

1. Introduction

Since 2011, about 160 million personal information has been leaked from the domestic financial institutions in Korea [1]. Among them, 120 million of privacy leakage accident which was leaked from three financial companies in January 2014, has caused substantial damage to the national economy and reliability of financial institutions [2]. The Government of the Republic of Korea, considering these accidents as a national issue, have decided to plan security policy by enforcing the personal information security law [3] as follows. First, a document including personal information must be encrypted. Second, a document including personal information must be stored on an isolated storage. Third, when a document including personal information needs to be deleted, the document must be deleted permanently in a secure way [4, 5].

However, these laws provide inconvenience for users managing personal information. In regards to the first law, a user has to manually encrypt documents including personal information. In regards to the second law, a user has to pick out all of documents including personal information and store them on an isolated storage. In regards to the third law, a user has to permanently delete a document that contains personal information in a secure way when deleting the document. Therefore, the laws induce user-obstructive inconvenience for users managing personal information.

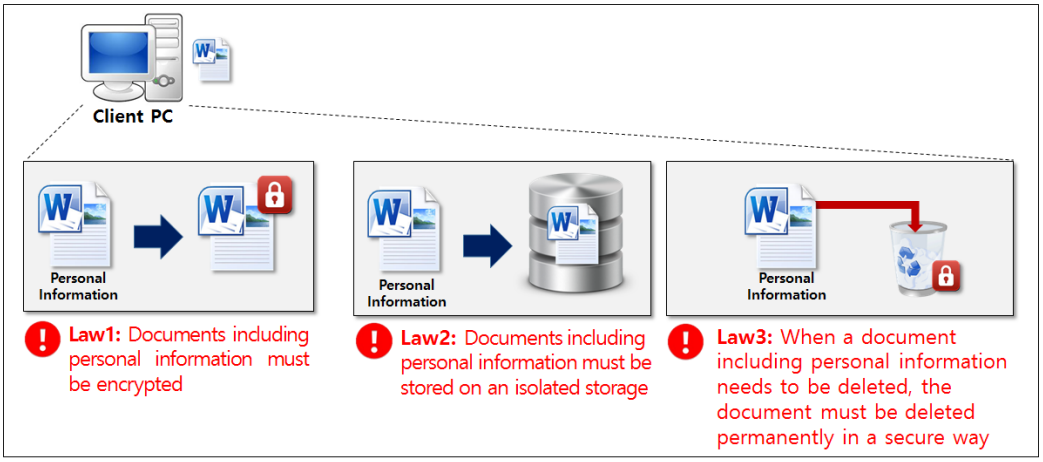
† TEAM: Transparent and Privacy-Enhanced Access Mechanism
* Ki-Woong Park: Corresponding Author (woongbak@sejong.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (2016R1A4A1011761).

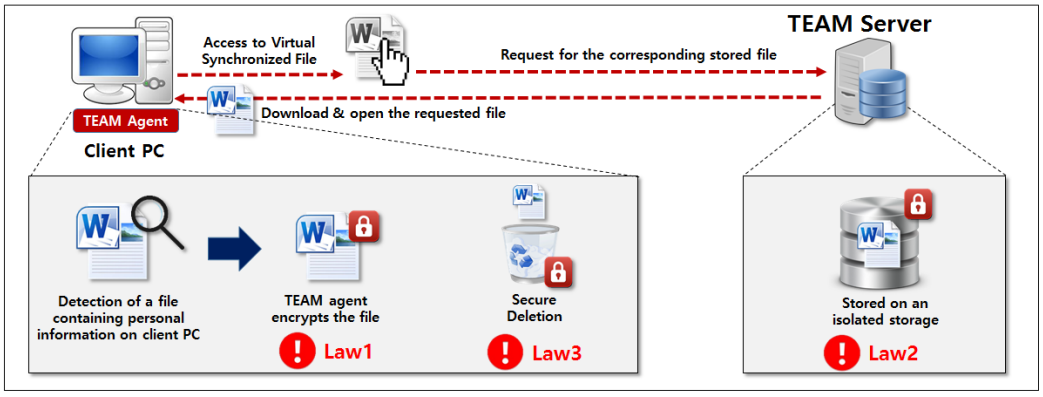
ISSN: 1738-9976 IJSIA

Figure 1-(a) describes what users have to do for complying with the personal information security law. Following the laws, users have to encrypt a document containing personal information. If the file needs to be eliminated, the file must be deleted in an unrecoverable manner. In addition, the files containing personal information must be stored on an isolated storage system.

This paper proposes a virtual synchronized file-based transparent and privacy-enhanced storage system a system, termed *TEAM*, to alleviate such inconveniences through the process as described in Figure 1-(b). *TEAM* automatically complies with the law by detecting, encrypting, and uploading documents containing personal information at the *TEAM* server which is a secure storage server. When a document including personal information needs to be deleted, the *TEAM* agent on the client PC permanently deletes documents containing personal information. In addition, the proposed 'Virtual Synchronized File' (VSF) enables users to access for personal information documents on the *TEAM* server in a transparent manner of a file access scheme for a normal document.



(a) Required actions for complying with the personal information security law



(b) File access procedure with a proposed system, TEAM

Figure 1. File Access Scheme Comparison

The remainder of this paper is organized as follows: Section 2 discusses the previous work on conventional privacy protection systems. Section 3 describes the overall system design of the proposed *TEAM* and its operational flow in more details. Section 4 presents an implementation and experimental results of our system from the performance perspective. Finally, Section 5 presents the conclusion and further works.

2. Related Work

In the last few years, several studies have been conducted on mechanisms for personal information protection. M. Mow Bray and S. Pearson has proposed a client-based privacy manager for cloud computing [6]. It is run on a client PC, which is enhanced with the client-based privacy manager to alleviate the risk of leakage in cloud computing environment. The user would upload personal information to a cloud server through the client-based privacy manager.

Yi-mu Ji presents a hybrid privacy solution based on a policy-attribute-based encryption [7]. The proposed system classifies personal information by a predefined protection level. The classified personal information is uploaded to the server by the user. The user is able to access personal information stored in the server by accessing to the server.

Above two studies propose personal information protection systems to enhance security of personal information. However, the proposed *TEAM* of this paper has a competitive edge in efficiency and the level of security compared to those studies. The proposed system in this paper is able to resolve the problems of the previous studies in the following manner:

1. The TEAM automatically encrypts documents containing personal information and uploads them to the TEAM server.

2. The VSF concept provides users with a transparent user experience of the file access scheme in conventional file systems.

3. TEAM protects privacy by storing the encrypted documents in an isolated storage server (TEAM server), while the encryption key is kept on the client PC.

4. The system requires user authentication through a daemon when accessing document including personal information. The daemon protects the document from being accessed by a third party

3. Overall Operation Flow of *TEAM*

This paper presents privacy protection system, called *TEAM* which enhanced with a virtual synchronized file (VSF) concept. A VSF has a specific extension with a specific execution rule. And each VSF is mapped to a specific document which is stored at the *TEAM* server. A user of the client PC can access a document stored at the *TEAM* server by double clicking VSF mapped to the document file like a normal document. Consequently, VSF provides users with a transparent user experience by offering an identical document access scheme, which differs from conventional secure storage systems [8, 9].

Table 1. Notations of the Entities and Messages Symbols

Definition of the Entity Symbols	Definition of the Message Symbols
· *C*: Client · *S*: *TEAM* Server	· PU_x = Public key of *x* · PR_x = Private key of *x* · *Nonce* = Generated random data against replay attacks · $K_{x,y}$ = Symmetric key shared between *x* and *y* · *FK* = File Encryption Key · *n* / Splite(*D*, *N*) : n_{th} partial data of *D* that is consists of *N* packets · *PW*: Password · *Fin*: Communication Finish Sign

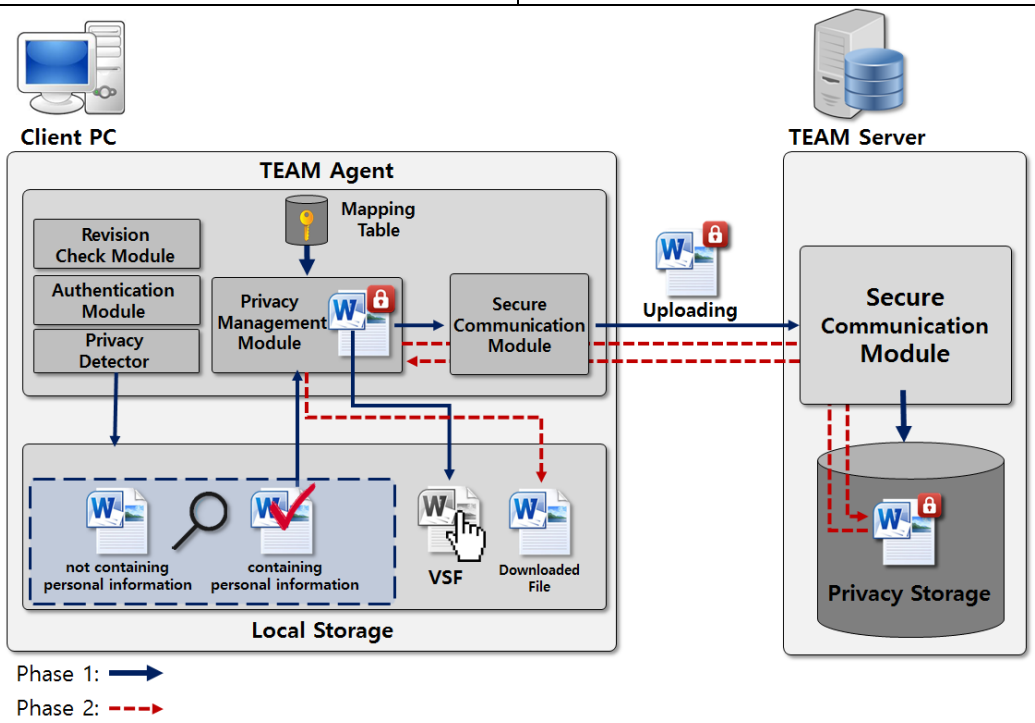
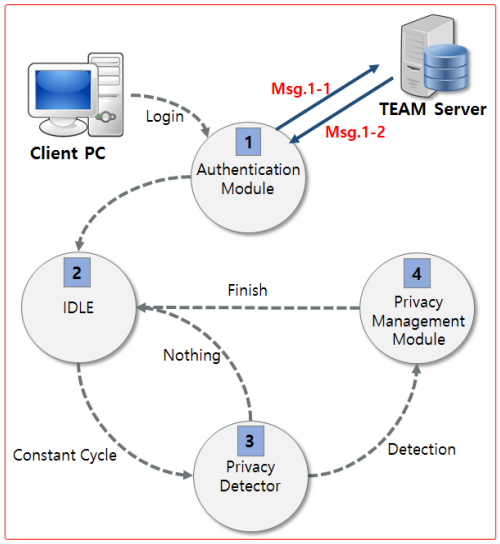


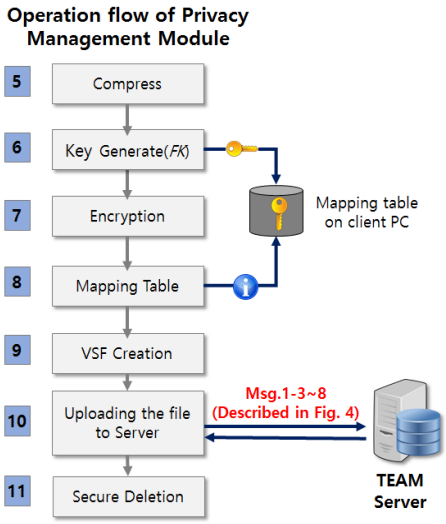
Figure 2. Overall Operation Flow on the basis of TEAM

As shown in Figure 2, *TEAM* consists of a client PC and a *TEAM* server. *TEAM* agent on client PC performs a process to comply with the personal information security law automatically. Overall operation flow on the basis of *TEAM* can be divided into two parts, *Phase1* and *Phase2*. *Phase1* is a process for uploading privacy document in a secure and transparent manner. *Phase2* is a process for accessing the uploaded document with the VSF concept. Section 3.1 and Section 3.2 present the overall operation flow. Table 1 presents the notation of entity and message symbols for describing the upload and access protocol on the basis of *TEAM* in Section 3.1 and Section 3.2

3.1. Phase1: Process for Uploading a Document

As shown in Figure 3, a user needs to login over the authentication module when the user boots the client PC or when the corresponding session value is expired. Authentication module prevents third party for accessing documents containing personal information. After the authentication, the user can interact with the *TEAM* server through the authenticated session (Msg. 1-1, 2 in Figure 4). The privacy detector of *TEAM* agent on the client PC searches documents containing personal information within the local storage of client. If the privacy detector detects a document containing personal information, the privacy management module of *TEAM* agent then compresses the detected document to reduce network traffic and storage space. After the compression, the agent generates a key (*FK*) to encrypt the detected document. Following that, *FK*, file name of the document, and the corresponding hash value are stored at the mapping table of *TEAM* agent on the client PC. The *FK* will be referred when the encrypted document is decrypted. Then, the agent creates VSF which is pointing the encrypted document stored in the *TEAM* server. The encrypted document and the hosting information which is consists of *Hash (File||FileName)* and session data are uploaded to the *TEAM* server (Msg. 1-3~6 in Figure 4). On the reception of the Msg. 1-3, the server stores the hosting information and the encrypted document into the server database to manage the received document containing the personal information. Then, client received an acknowledgement message (Msg. 1-7, 8 in Figure 4). Finally, the original privacy document is deleted permanently in a secure way.





Message 1-1.	Client → Server	$E\{PU_S, ID \| Hash(PW) \| Nonce_1 \| E\{PR_C, ID \| Hash(PW) \| Nonce_1\}\}$
Message 1-2.	Server → Client	$E\{PU_C, Nonce_1+1 \| Session \| E\{PR_S, Session \| K_{user, server}\}\}$
Message 1-3~8	Corresponding messages (3-11) are described as a message flow chart in Fig. 4	

Figure 3. Operation Flow Chart for Uploading the Privacy File

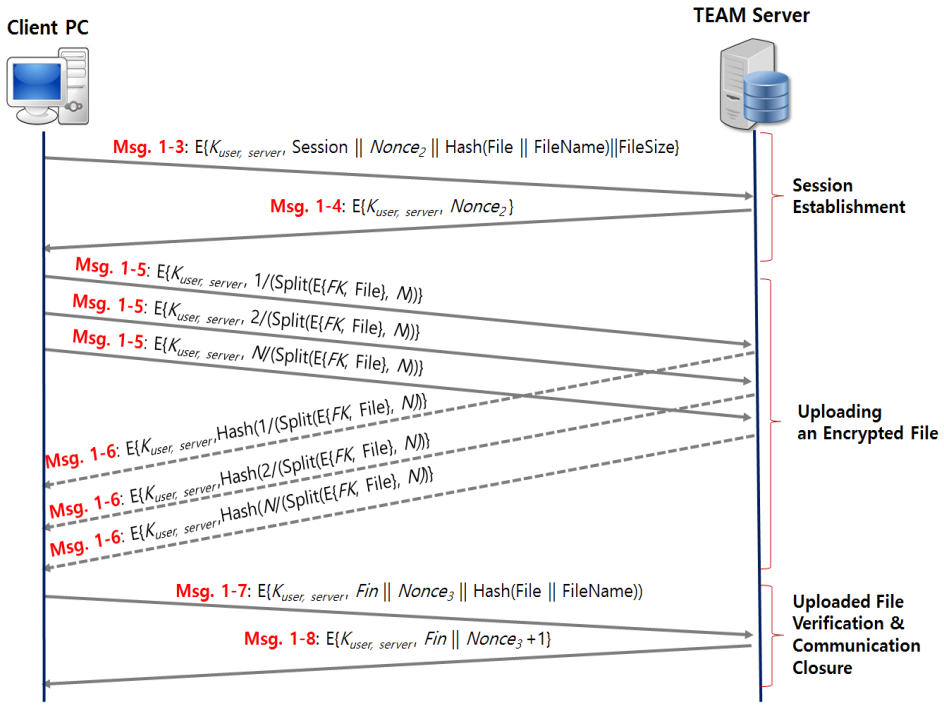
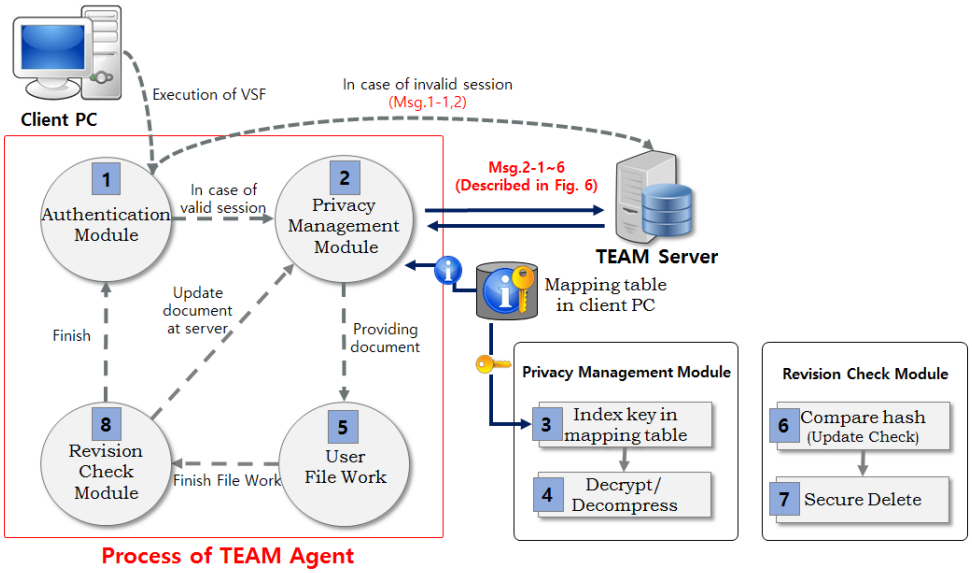


Figure 4. Message Flow Chart for Uploading a Privacy File

3.2. Phase2: Document Access Process

Figure 5 presents the process flow for accessing stored documents on the *TEAM* server when a client executes VSF. If a user executes VSF, the authentication module on client PC checks whether the session value is valid or not. If the value is not valid, the authentication module requests an authentication to user through *Msg. 1-1*. Then a generated valid session value by the server is sent to the *TEAM* agent over *Msg. 1-2*. Because the session value has an expiration time, the server needs to reissue the session value if the session value is expired. This process is to protect personal information from any other third party. After login, the privacy management module of the *TEAM* agent requests a corresponding document of the VSF to server by searching the mapping table on the client PC (Msg. 2-1~6 in Fig. 6). Then, server sends the corresponding document to client. Following that, *TEAM* agent decrypts and decompresses the received document using the stored *FK* on the mapping table of the client PC. After a user finishes the document work, the revision check module of the client PC checks whether the document was modified or not, and the next process varies according to this result. If any part of the document was not modified, it simply finishes with secure deletion. Otherwise, it updates new record into the mapping table on client PC, encrypts the modified document, and sends it to the *TEAM* server.



Message 2-1~6.	Corresponding messages (1-6) are described as a message flow chart in Fig. 6

Figure 5. Operation Flow Chart for Accessing the Privacy File

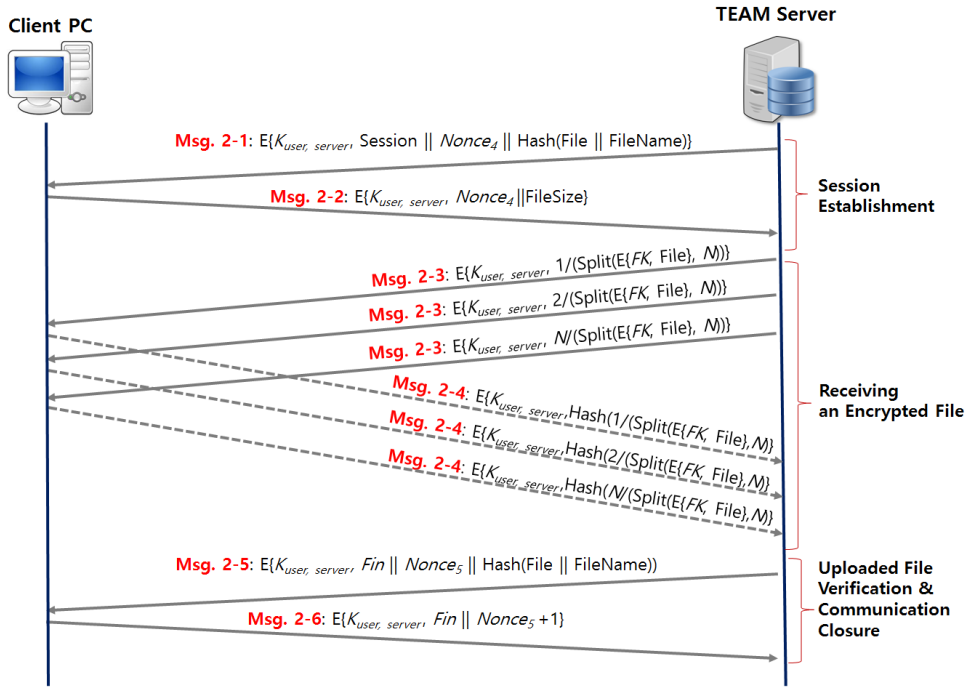


Figure 6. Message Flow Chart for Accessing a Privacy File

4. System Implementation and Performance Evaluation

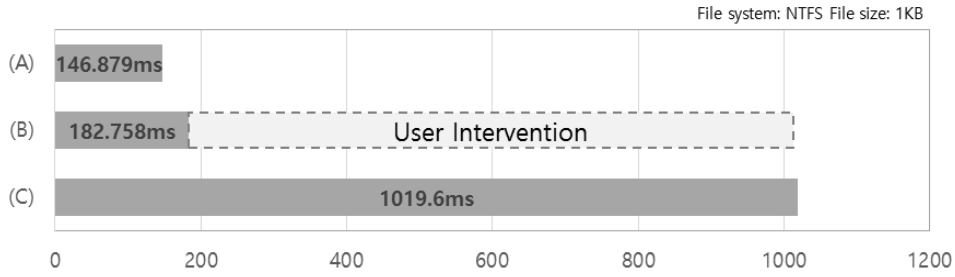
4.1. Implementation of *TEAM*

The *TEAM* server have an *Intel i3-4370* CPU and a 12GB main memory. The client PC has an Intel i5-4690 CPU and an 8GB main memory. The privacy detector on the client PC searches documents including personal information. If the detector detects a document, the detector sends the document to privacy management module through inter process communication. On the reception of the document, the privacy management

module compresses the document by using *gzip* [10] algorithm. To encrypt the compressed document, the privacy management module generates key (*FK*) and encrypts the document by using *AES* algorithm [11]. Following that, the module creates metadata of the document. The metadata includes document name, hash value of the document, encryption key (*FK*), and time stamp. The metadata is stored into the mapping table which is SQLite database. *TEAM* agent sends the metadata to the *TEAM* server with the encrypted document. When the *TEAM* agent sends them to the server, the agent creates virtual synchronized file (*VSF*) in accordance with the created metadata. When the transmission of the encrypted document to the server is completed, the agent carries out a secure deletion for the original document including personal information in an unrecoverable manner of Guttmann algorithm [12].

4.2. Performance Evaluation of *TEAM*

The latency performance is evaluated by measuring a time interval between the time of VSF execution and the decrypted open time of the document. As shown in Figure 7, the time to open a normal document is 146.878ms. The time to open an encrypted document file is 182.758ms. However, the 182.758ms does not include the latency by a user intervention (ex. Time to type a password for the document to be accessed). Through the execution of VSF, the average decryption and execution latency, which brings privacy document to the client PC and decrypts the received document using key stored in client PC, is 1019.6ms. Although the latency to access the document over the VSF (1019.6ms) is longer than the time to access the document in a normal manner without any security considerations (146.879ms), *TEAM* has advantages that automatically carries out without user intervention by using key stored in client database only with an avg. 872.721ms.



(A): The time to open a normal document **(B)**: The time to open an encrypted document **(C)**: The time to execution of VSF

Figure 7. VSF Execution Latency on Situation-specific

5. Conclusion

Under the revised personal information security law in Korea, persons handling the personal information experience inconveniences. As a remedy to this problem, we proposed a privacy protection system, called TEAM, enhanced with a 'Virtual Synchronized File' concept, which keep convenience by providing users with a transparent user experience interface, while complying with the personal information security law. The VSF file has a specific extension with a specific execution rule so that users may open the VSF like a normal document. Consequently, VSF provides users with a transparent user experience by providing identical document access scheme in comparison to the conventional file system.

References

[1] M.-J. Kim, N. Heo and J. Yoo, "A Study on the Stock Price Fluctuation of Information Security Companies in Personal Information Leakage", Journal of the Korea Institute of Information Security and Cryptology, vol. 26, no. 1, **(2016)** Febuary, pp. 275-283(9 pages).
[2] J.-H. Eom and M.-J. Kim, "Effect of Information Security Incident on Outcome of Investment by Type of Investors: Case of Personal Information Leakage Incident", Journal of the Korea Institute of Information Security & Cryptology, vol. 26, no. 2, **(2016)** April, pp. 463-474 (12pages).
[3] Korea Personal Information Protection Act. url: http://www.law.go.kr/(10465)
[4] Mowbray, Miranda, and Siani Pearson, "A client-based privacy manager for cloud computing", Proceedings of the fourth international ICST conference on Communication
[5] K.-P. Yee, "Aligning security and usability", IEEE Security & Privacy, vol. 5, **(2004)**, pp. 48-55
[6] R. Yigal, U. Mattsson and R. Ortega, "Assignment of security contexts to define access permissions for file system objects", U.S. Patent No. 9,230,128, **(2016)** Jan. 5.
[7] Y.-M. Ji, "A Privacy Protection Method Based on CP-ABE and KP-ABE for Cloud Computing", Journal of Software, vol. 9, no. 6, **(2014)**, pp. 1367-137.
[8] Y. Yu, "Remote data possession checking with enhanced security for cloud storage", Future Generation Computer Systems, vol. 52, **(2015)**, pp. 77-85.
[9] H.-T. Yeh, "User authentication of smart mobile devices and cloud storage services", Innovation in Design, Communication and Engineering: Proceedings of the 2014 3rd International Conference on Innovation, Communication and Engineering (ICICE 2014), Guiyang, Guizhou, PR China, October 17-22, 2014. CRC Press, **(2015)**.
[10] L. P. Deutsch, "GZIP file format specification version 4.3.", **(1996)**.
[11] J. Daemen and V. Rijmen, "AES proposal: Rijndael", **(1999)**.
[12] Y. Ka-Ping, "Aligning security and usability", IEEE Security & Privacy, vol. 5, **(2004)**, pp. 48-55.

Authors

Hye-Lim Jeong, She received the BS degree in the department of information security from Daejong University in 2015, and the MS degrees in the department of information security from Daejon University 2017. She is a Ph.D Student of Sejong Univerity. Her research interests include system security and secure storage system.

Ki-Woong Park, He received the BS degree in computer science from Yonsei University in 2005, and the MS and PhD degrees in electrical engineering from the KAIST in 2007 and 2012, respectively. He is an assistant professor in the information security department at Sejong University. He worked as a researcher at National Security Research Institute in 2012. His research interests include system security issues for a real cloud and mobile computing systems. He is a member of the IEEE and ACM.