Data Management System based on Privacy Aware File Virtualization

Hye-Lim Jeong
System Security Lab.
Daejeon University
Daejeon, Korea
hyello13@gmail.com

Sung-Kyu Ahn
System Security Lab.
Daejeon University
Daejeon, Korea
yiimfn@gmail.com

Chang-Hoon Lee
System Security Lab.
Daejeon University
Daejeon, Korea
0x80520@gmail.com

Ki-Woong Park*
Dept. Information Security
Daejeon University
Daejeon, Korea
woongbak@dju.kr

***Abstract*— Recently, according to cause personal information leakage accident, personal information protection act is revised. Although previous related works proposed some manners for personal information protection, the revised act is uncomfortable to users. Satisfying both personal information protection and convenience have many problem. We proposed system for solving the problem namely *Pandora*. It detects a file containing personal information. The file is encrypted and send to server storage. In addition, *Pandora* proposed virtual link file. This can access personal information stored in server. A user execute the virtual link file as execution of a normal file. *Pandora* satisfies convenience and personal information protection.**

Keywords— Privacy protection, Data Security, Access Control

I. INTRODUCTION

In early 2014, 20 million of personal information was leaked from three financial institutions in the Republic of Korea [1]. According to the issue of personal information, Korea government enforced personal information protection act in order to solve the problem [2]. Revised personal information protection act gives difficulties of managing personal information to business using personal information. We discussed mainly three acts in personal information protection act given difficulties. One revised personal information protection act requires to encrypt privacy information. As the act, users suffer from discomfort to manually encrypt each documents that contain personal information. Also, users should make sure whether all files contain personal information on user's PC. Another requires that a personal information is not able to recover when a user delete the personal information. So, users check if deleted personal information does not recover. The other requires to separately store a personal information at server. Users have to send personal information to server according to act.

For these reasons, we proposed virtual link file named VLF that provided user as execution. VLF required a personal information file to server, when it executes. VLF provide an interface to user like interface of a normal file execution. VLF works on a system named *Pandora*. *Pandora* gives convenience to user by running some process to comply with acts. Also, *Pandora* ensures transparency to user by keeping user interface. Therefore, we proposed a system to solve the difficulties mentioned above. As shown Fig 1, *Pandora* automatically detects files containing personal information in user's PC. Then *Pandora* encrypts the detected file and sends them to server. Then server provides VLF to a user. To run provided VLF enables a user to access a personal information stored in server. We solved difficulties of three acts through *Pandora*. *Pandora* is convenient by performing all processes automatically, without user intervention. The main contribution of this paper can be itemized as follow:

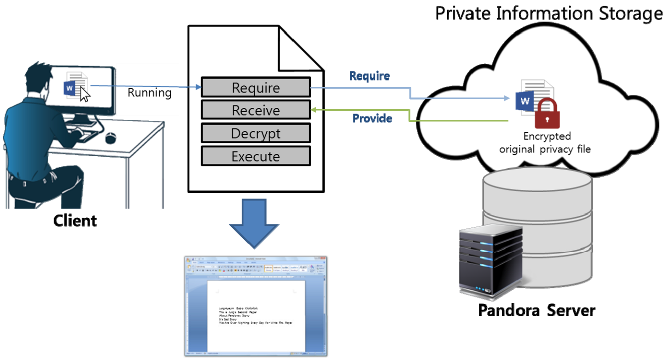

Fig 1. The usage scenario of the proposed system, *Pandora*

- First, we enable *Pandora* to remotely access the personal information in server. Through VLF execution, user can access the personal information over keeping user interface condition because VLF offers normal user interface to user in the manner of transparent.
- Second, the execution latency of virtual link file has a problem that the latency is slow rather than an execution latency of a normal file stored local storage. To solve the problem, we minimized the execution latency by building shared area to share data between client and server.

The remainder of this paper is organized as follows: In Section II, we discuss the previous work on conventional personal information backup systems. In Section III, we describe design and implementation of *Pandora*. We present experimental result of *Pandora* from the performance perspective in Section IV. Finally, the conclusion and future work are given in Section V

*: Corresponding author

This work was supported by the Human Resource Training Program for Regional Innovation and Creativity through the Ministry of Education and National Research Foundation of Korea (2015H1C1A1035859).

II. Related Work

We proposed *Pandora* for solving difficulties of act observation as stated above. *Pandora* automatically observed revised act without user's intervention. In addition, *Pandora* is comfortable because *Pandora* undisturbed user interface. Also, proposed *Pandora* protects personal information. In the last few years, several works have been to the study of personal information protection system. In this Section, we discuss about proposed manner from previous related works.

As in a client-based privacy manager for cloud computing [5], M. Mow bray and S. Pearson has proposed a client-based privacy manager for cloud computing. The paper's method comprises the privacy manager in the client to reduce the risk of sensitive data leakage and privacy loss in cloud environment. Before a user send data to the cloud, the privacy manager obfuscate data and then transmits encrypted data to cloud. After a user send data to the cloud, the privacy manager provides de-obfuscate given data from the cloud.

A Privacy Protection Method Based on CP-ABE and KP-ABE for Cloud Computing [6] presents a Hybrid privacy solution based on Policy-attribute-based encryption (ABE) and ciphers policy-ABE. Hybrid privacy solutions design different security levels depending on the security requirements of the user's personal data stored in the cloud system. Depending on the characteristics to generate Access Control Tree that controls the user's access. This solution provides the ability to securely store the user's personal information.

According to this method, it can protect the privacy data in the cloud. Since, if sensitive data be leaked, it is impossible to de-obfuscate data because the key is on the client. Given overall configuration and flow, the paper's method is similar to the *Pandora*'s method. However, that method is not suitable for the protection of privacy data for the following reasons: it is not consider about data existing in the user's PC. Whether or not it upload data to the cloud, protection of personal data is not built well because of original of sensitive data still remaining in the user's PC. In addition because cloud server covers only uploaded data on the cloud, privacy data is not stored and managed separately from the normal data. *Pandora* is able to improve the problems of that study is as follows:

- To protect privacy data within a user's PC, client detects privacy data existing in the user's PC, generates the key and encrypts the detected data. Then, backup the encrypted data to the cloud server and then delete that with the original data safely. It all operate actively and have advantage that no user's intervention is required.
- To separate privacy data from normal data and against data leakage, the cloud server generates a specific link file associated with received data and transmits it to the client. Therefore, not only key and the encrypted data is physically separate like A Client-Based Privacy Manager for Cloud Computing' method but also the original data does not exist in both the user's PC and the cloud storage.

III. *Pandora* Design & Implementation

A. *Design of Pandora*

Pandora is consist of two parts, the client, and the server. Personal information detector detects files containing personal information and sends detected them to server. Detected personal information files in client are deleted by overwriting. Then server creates VLF. VLF is able to create client PC according to system configuration environment. The VLF sends to client in order that a user can access personal information files stored in server.

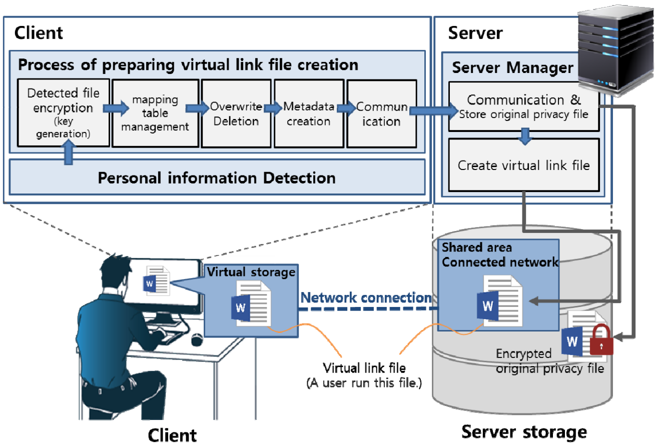
Client
Process of preparing virtual link file creation
Detected file encryption (key generation)
mapping table management
Overwrite Deletion
Metadata creation
Commun ication
Personal information Detection
Server
Server Manager
Communication & Store original privacy file
Create virtual link file
Virtual storage
Network connection
Shared area Connected network
Encrypted original privacy file
Virtual link file (A user run this file.)
Client
Server storage

Fig 2. The Diagram of *Pandora* Process

In the Fig 2, as stated above *Pandora* consists of client and server. Client consists of personal information detector with process preparing VLF creation. Personal information detector works to detect files containing personal information in client PC. To encrypt the personal information, process preparing VLF creation generates a key. Process of preparing VLF creation encrypts the detected personal information by using key. The process creates metadata that detected through personal information detector. Then metadata of detected files that encryption key, file name, execution program, hash value and time stamp are recorded at mapping table in client PC. After metadata is recorded on mapping table in client PC, Personal information in client PC will delete to overwrite. Encrypted a file and created metadata is sent to backup server. The metadata, needed to create a VLF, is sent to the server. Because a VLF is created on the server. Metadata is index value in order to access a personal information in server. Server consists of server manager. A VLF is generated by server manager. Then, the VLF is stored in a shared area that is a specified area of server storage. Client connects to the area using network. Client used the area as local storage. Therefore, the user may use VLF in the shared area to access to the personal information file.

Fig 3 is in the operation flow of the VLF execution when a user's run a VLF. A user signs a client daemon program before run a VLF. This process authenticates the user that access a personal information file in server. The process prevents to open the personal information by third party. If a user executes a VLF in shared area, the VLF requests the encrypted origin personal information to server by reading metadata in VLF. Server provides the encrypted origin personal information file through share area. Received the origin personal information file is decrypted by using a key recorded Mapping table. Then a user works provided personal information. However, the VLF's process is changed by user.

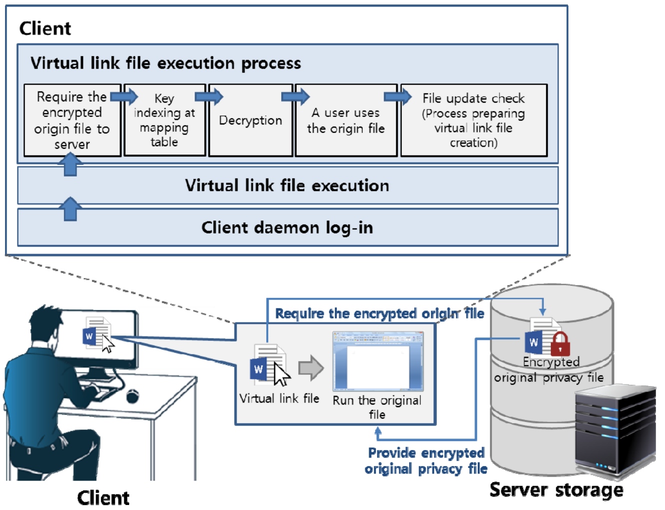
Client
Virtual link file execution process
Require the encrypted origin file to server
Key indexing at mapping table
Decryption
A user uses the origin file
File update check (Process preparing virtual link file creation)
Virtual link file execution
Client daemon log-in
Require the encrypted origin file
Virtual link file
Run the original file
Encrypted original privacy file
Provide encrypted original privacy file
Client
Server storage

Fig 3. The operation flow on the basis of privacy aware file virtualization

If the user revises the origin personal information file, the VLF would check mapping table for comparing previous hash value with hash value of revised file. Mapping table in client PC is updated in case both value is different. In case the hash values is same, the VLF deletes by overwriting to origin personal information. Finally, the VLF process is terminated.

B. Implementation

The system proposed this paper implements in a client PC having personal information and a storage server managing personal information that received from client PC. Format object of personal information detection in client PC is text file format. Personal information encryption is AES using 128bit key. The system uses SHA-1[7] hash in order to check personal information's update. The database that metadata of personal information sent to the server is implemented as SQLite3 [8]. It is facile for user to access personal information stored server. Shared area managing VLF is built up NFS [9] (Network File System). Server managing VLFs, personal information is easy to users when a user uses a personal information stored storage server. Process of preparing virtual link file creation generates 128bit key in order to encrypt the personal information detected from the personal information detection. Then process of preparing virtual link file creation encrypt the personal information using AES. After the encrypt process, the metadata of personal information is recorded on SQLite3 database and is sent to server. Server creates a VLF through metadata of personal information. Server send the VLF to client PC. Particular area in storage server is set up NFS. Then, a user uses the area in opening a personal information. When a user executes a VLF, an encrypted personal information in NFS area is decrypted. A user work on the decrypted personal information.

IV. Performance Evaluation

We measure a latency from executing a *Pandora*'s VLF to opening a personal information. When VLF provides a personal information file to the user, it takes the key from a DB and performs a process that decrypts the file. In Fig 4, File execution case is a measured latency that running typical a personal information in the same file system environment. File decryption & execution case is a measured latency that runs by decrypting the encrypted personal information file. VLF execution case is a measured latency that execution of a personal information file by executing a VLF of the system. For the experiments, Result of File execution case is 146.879ms. Result of File decryption & execution case is 172.758ms. Result of VLF execution case is 1019.6ms. In the case of VLF execution, a relatively long latency was measured, because of decryption by indexing to the key from the DB. We consider latency of file decryption & execution to include typing latency of the user's decryption key and network latency to get the file from the server, latency of file decryption & execution and latency VLF execution is similar. Also, VLF as executable file has some process that decryption, indexing etc. VLF is maintained to 9KB regardless of personal information file size. It is effective use of space because the file size of the personal information was more than 9KB.

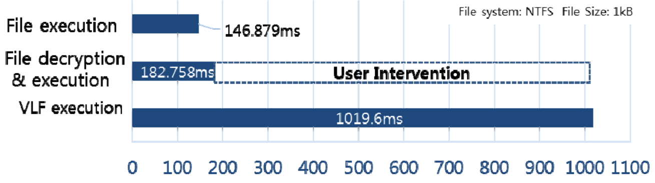

Fig 4. Situation-specific latency

V. Conclusion

We proposed *Pandora* that data management system based on privacy aware file virtualization. *Pandora* encrypts detected personal information file. Then, *Pandora* creates VLF, after send encrypted file to server. A user access a personal information file stored in server by executing VLF. However, *Pandora* has problem, which latency of VLF providing personal information file to user is delayed. The latency is long in comparison with execution latency of a file stored on local, or execution and decryption latency of a file *Pandora* automatically performed all process without user intervention. Therefore, *Pandora* latency is significantly different than the other's latency. Also, *Pandora* is efficient by storing a personal information file at server storage and using VLF of 9KB size. *Pandora* enables to protect personal information through automatic encryption, backup service, VLF.

References

[1] Sophia Yan, K.J. Kwon, “Massive data theft hits 40% of South Koreans”, CNN Money, January 21, 2014

[2] Korea Personal Information Protection Act. Article 24, paragraph 3

[3] Korea Personal Information Protection Act. Article 21, paragraph 2

[4] Korea Personal Information Protection Act. Article 21, paragraph 3

[5] Mowbray, Miranda, and Siani Pearson, "A client-based privacy manager for cloud computing," Proceedings of the fourth international ICST conference on COMmunication system softWAre and middlewaRE. ACM, 2009, pp. 5

[6] Ji, Yi-mu, et al. "A Privacy Protection Method Based on CP-ABE and KP-ABE for Cloud Computing," Journal of Software 9.6, 2014, pp. 1367-137

[7] BONEH, Dan, et al. “Public key encryption with keyword search.” , Advances in Cryptology-Eurocrypt 2004. Springer Berlin Heidelberg, 2004. p. 506-522.

[8] Daemen, Joan, and Vincent Rijmen. "AES proposal: Rijndael." (1999).

[9] Storer, Mark W., et al. "Secure data deduplication." *Proceedings of the 4th ACM international workshop on Storage security and survivability*. ACM, 2008.

[10] Junyan, Lv, Xu Shiguo, and Li Yijie. "Application research of embedded database SQLite." *Information Technology and Applications, 2009. IFITA'09. International Forum on*. Vol. 2. IEEE, 2009

[11] Pawlowski, Brian, et al. "The NFS version 4 protocol." *Proceedings of the 2nd International System Administration and Networking Conference (SANE 2000)*. Vol. 2. No. 5. 2000