Design and Implementation of TPM-Enhanced Privacy Protection System

Hye-Lim Jeong, Sung-Kyu Ahn, Mun Sung Kim*, Ki-Woong Park†
System Security Lab. Daejeon University, Daejeon, Korea
Dept. of Business Consulting Daejeon University, Daejon, Korea*
Dept. of Computer and Information Security Sejong University, Seoul, Korea†
hyello13@gmail.com, yiimfn@gmail.com, k4022@daum.net, woongbak@sejong.ac.kr

***Abstract*—** Software-based key management in conventional secure storage systems can cause a key-leakage problem. For example, malware may access encrypted documents by deriving a corresponding key in an abnormal way. In this paper, we present a TPM (Trusted Platform Module)-enhanced privacy protection system, termed TPS, which enhances the security of documents containing personal information by performing hardware-based security operations which are *Seal/Unseal* operations and *PCR* value of TPM. TPS encrypts the personal information documents through the *Seal* for later decryption with TPM *Unseal*. By comparing the *PCR* value when the *Seal* was performed and the *PCR* inside the TPM, TPS provides a function that the encrypted files can be decrypted only when *PCR* inside the TPM is matched to the *PCR* value when the *Seal* was performed. Consequently TPS can protect the documents by blocking file accesses in the abnormal state of the client system.

Keywords*— *Privacy Protection, TPM, Access Control

Ⅰ. Introduction

Key management in a conventional secure storage system can lead to key-leakage problems, which can lead to leakage of encrypted document by malware. For example, malware may access encrypted documents by deriving a corresponding key in an abnormal way. This study is an extension of our previous work [1], in which we focused on data management scheme based on privacy aware file virtualization. Our objective in this paper however, is to enhance a security of documents containing personal information by performing TPM (Trusted Platform Module)-based security operations. In this paper, we present a TPM-enhanced privacy protection system, termed TPS, which enhances the security of documents containing personal information by performing hardware-based security operations which are *Seal/Unseal* operations and *PCR* value of TPM. The *PCR* is used as a value for the status integrity verification of the client system. The *Seal* operation encrypts the *PCR* value and the document containing personal information. The *Unseal* operation performs the decryption if the comparison between the *PCR* of the encrypted document and the *PCR* inside the TPM is matched. TPS encrypts documents containing personal information through *Seal*, which is a function of TPM. Therefore, the integrity of the client PC can be guaranteed. If the hard disk containing the encrypted document through the *Seal* operation is moved to another computer system, the decryption of the document cannot be performed due to the difference of the *PCR* value. A document containing personal information applied *Seal* encryption is transmitted to the server. Then a Virtual Trusted File (named VTF) is created on the client PC. VTF is a link file pointing the encrypted file stored on the server. Therefore the client can access the document stored in the server by clicking the VTF. Following that the original document on the client PC is deleted in a secure way. The document which is requested to be browsed can be decrypted through the *Unseal* operation by comparing the *PCR* of the document encrypted with the *Unseal* operation and the *PCR* inside the TPM. TPS can protect the documents by blocking file accesses in the abnormal state of the client system. Consequently, we propose a personal information protection system which is more secure than our previous work.

This paper is organized as follows: Section 2 describes related work. Section 3 describes the overall architecture of TPS. Section 4 concludes this paper.

Ⅱ. Related work

This section describes our contributions in comparison to previous related works [2, 3, 4] and the TPS proposed in this paper. Then we describes overall basic function provided by TPM. As a related work, 'A Secure Cloud Backup System with Assured Deletion and Version Control'[4] are explained in this paper. FadeVersion proposed in this related work is a secure cloud backup

†: Corresponding Author

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (2016R1A4A1011761) and by the Ministry of Education (2015H1C1A1035859).

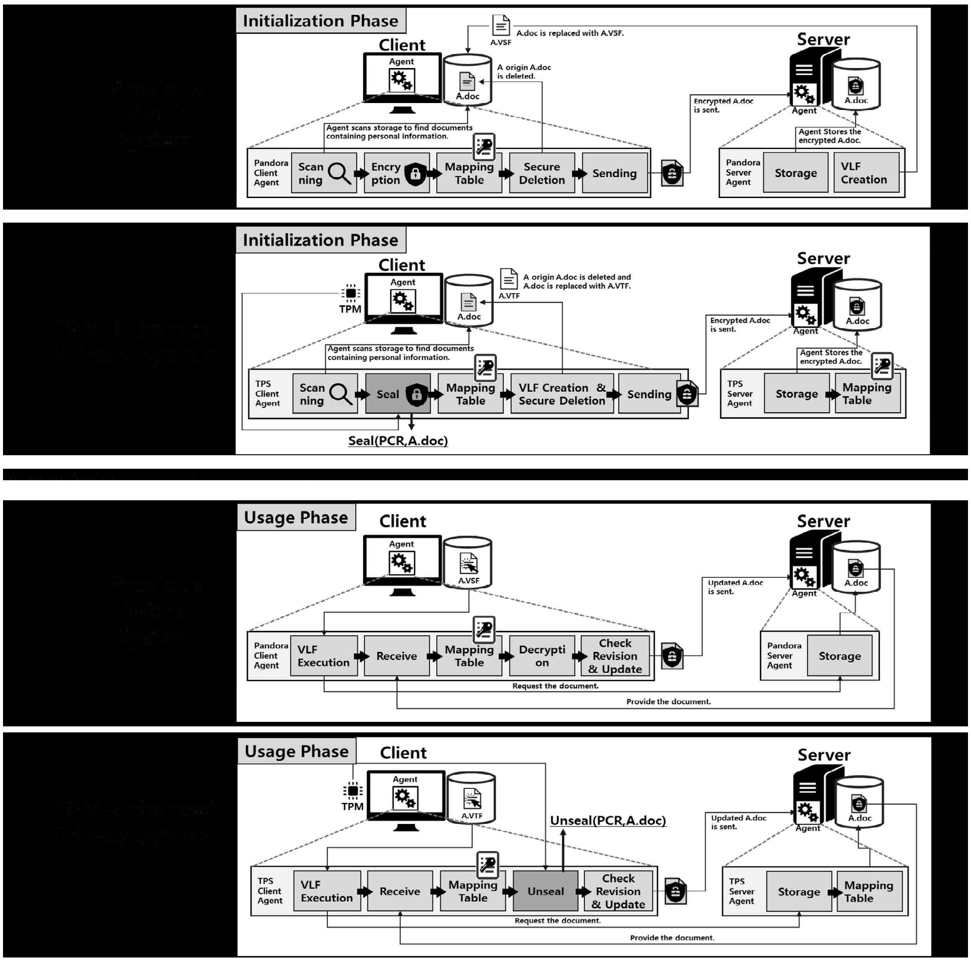

Fig. 1 Architecture of Previous Work System and TPM-Enhanced Privacy System

system that performs the encryption / decryption process during data upload process and guarantees complete deletion of data. However, there is a problem that can occur in the client side before the data backup process in the FadeVersion environment. In this paper, to solve this problem, we implemented a data integrity system based on *Seal/Unseal* operation of TPM to construct a secure personal information protection environment.

TPS uses TPM[5] to encrypt personal information documents. TPM is an abbreviation of Trusted Platform Module. It is a security chip that is designed to perform encryption key management and encryption only in a security chip manufactured by hardware since key management of legacy software encryption is vulnerable to security attack. The *PCR* is used as a value for the status integrity verification of the client system. The *Seal* operation encrypts the document containing the personal information using the value stored in the *PCR*. The *Unseal* operation can decrypt the document only if the *PCR* value in the TPM matches the *PCR* value contained in the encrypted document, and can decrypt only the same PC as the PC on which the *Seal* operation is performed. TPS encrypts the documents containing personal information with *PCR*, which is recorded the state of a system, through the encryption technology of TPM called *Seal/Unseal*. This process is more secure than the previous work, which is our previous work, by ensuring the integrity of personal information.

The TPS proposed in this paper does not require the user to directly upload the personal information document to the storage server. The TPS automatically carries out all processes for protecting personal information so that the user does not have to intervene in the process of protecting personal information. In addition, unlike the method of accessing the document containing personal information stored in the cloud server through the web, the TPS provides the document containing personal information in the same manner as the general document is used to the user through the VTF file.

Ⅲ. Design and Implementation of TPS

TPS consists of the client part and the storage server part. The client is subject to comply with the personal information protection act [6-8] for the documents containing the personal information in the client.

Compared to client of system of the previous work, client of TPS is carried out as scanning documents, *Seal/Unseal*, and VTF creation. The seal keys are used for *Seal* encryption generate keys using the *PCR* value, which is the status register of the client PC. Mapping table in client manages documents sent to server and sealing key of that documents. In this process, the integrity of the document containing personal information is ensured through the *Seal*. The storage server stores the documents including the personal information received from the client and transmits documents upon the requests from the client. The server agent manages received document through mapping table. The overall operation flow on the basis of TPS consists of two phases which are *init phase* and the *Usage phase*.

The previous work has the problem of encryption key management by performing software-based encryption in the process of encrypting documents including personal information. To solve this problem, this paper enhanced key management security through TPM, hardware-based encryption module. As shown in Fig. 1, as for the *init* process, the encryption process in the previous work performs software-based encryption and key management by recording encryption key to the mapping table. Such implementation may result in key leakage problem due to low security of key management. However, in case of TPS, the document name and hash value are stored in the mapping table without storing key in the mapping table since performing seal function which is the encryption of TPM using TPM built-in key. As shown in Fig. 1, as for the usage process, the virtual file VLF in the previous work receives the document stored in the storage server and performs software decryption. However, the virtual file VLF of TPS performs the TPM *Unseal*.

As shown *init phase* in Fig. 1, the client agent performs a process of finding a document containing personal information on the client storage. While the previous work performs a software-based encryption, TPS encrypts documents containing personal information through *Seal*, which is a function of TPM. *Seal* performs the process of encrypting the document detected by the previous step using the derived key from TPM. After that, it stores the document and the sealing key to the mapping table. A VTF file is created and then send a sealed personal documents to the server. The original documents that has been left is securely deleted so that they are not left on the client. The keys generated in all encryption and decryption processes are managed by TPM. Due to *Seal/Unseal* operations using *PCR* in TPM, documents containing personal information can be viewed only if the client system has a normal state.

As shown *usage phase* in Fig. 1, *usage phase* performs VTF execution, document request to server, document provided by server, finding key in client mapping table, *Unseal*, and renewal. The *usage phase* starts by executing the user's VTF file on the client. The VTF file requests the documents to the storage server. The storage server receives the request and finds the file from the mapping table in the server. After that, the storage server sends it to the shared area of the client and the server. The client receives the file from the shared area and the *Unseal* process performs through the corresponding key in the mapping table of the client. Decryption including the integrity verification operation is performed by comparing the *PCR* value including the client PC status through the corresponding process. The system of pervious work is relatively low security, since it does not perform the process. The decrypted document is provided to the user. The client Agent waits for the end of the document and performs the renewal process. The renewal process performs a hash comparison to check whether the document is modified. If the hash result value is different, it performs the same process as performed by *init phase* and sends it to the server.

Ⅳ. Conclusion

This study is an extension of our previous work[1], in which we focused on data management scheme based on privacy aware file virtualization. Our objective in this paper however, is to enhance a security of documents containing personal information by performing TPM-based security operations. To achieve it, we present a TPM (Trusted Platform Module)-enhanced privacy protection system, termed TPS, which enhances the security of documents containing personal information by performing hardware-based security operations which are *Seal/Unseal* operations and *PCR* value of TPM. Consequently, TPS can protect the documents by blocking file accesses in the abnormal state of the client system.

References

[1] Hye-Lim Jeong, Sung-Kyu Ahn, Chang-Hoon Lee, Ki-Woong Park, "Data Management System based on Privacy Aware File Virtualization", ICNGC, 2015.

[2] 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.

[3] 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.1367-1375.

[4] Rahumed, Arthur, et al. "A secure cloud backup system with assured deletion and version control." 2011 40th International Conference on Parallel Processing Workshops. IEEE, 2011, pp.160-167.

[5] Park, Ki-Woong, et al. "THEMIS: A Mutually verifiable billing system for the cloud computing environment." IEEE Transactions on Services Computing 6.3, 2013, pp. 300-313.

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

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

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