Open Access Journal of Information Systems (OAJIS)

Available at http://is.its.ac.id/pubs/oajis/index.php

The Third Information Systems International Conference

Seamless Document Tracking Using FUSE and SAMBA File System

Henning Titi Ciptaningtyas[a], Royyana Muslim Ijtihadie[a], I Gede Adhiarta Wiandana[a] a*

[a]Department of Informatics,Faculty of Information Technology, Institut Teknologi Sepuluh Nopember,
Kampus ITS, Sukolilo, Surabaya 60111, Indonesia

Abstract

Digital document is important aspect in the business. It contains critical company data such as supplier data, production data, financial data, client data, etc. Digital document in the company can be freely accessed using media data share. Sharing the confidential digital document is prone to leakage. The leak can cause company loss. To prevent leakage of digital documents especially from the insider, we need to insert specific information into each digital document that is copied by employee. The digital document is secured by using FUSE to insert additional information in a transparent way and Samba File System as a medium to share data. When there is document leakage, the company can track who and when is the document copied. Based on the experimental result, insert additional information into digital document can track the document leakage.

© 2015 Published by ISICO

Keywords: Digital Documents; FUSE; Samba; Information Leakage; Additional information.

1. Introduction

Almost every company has digital information to support their business activity. The digital data that is classified as confidential or important should remain secret only for the internal use in the company. Examples of the confidential data are as follows: supplier data, production data, financial data, client data, etc. However, some companies have lack of information security policy or even have no security policy at all. These security holes can be used by irresponsible parties from both inside and outside the company to take opportunity and disseminate the confidential information. The leakage data could cause a big loss.

* Henning Titi Ciptaningtyas. Tel.: +62-81 332 791 684; fax: +62 591 3804.
E-mail address: henning.its@gmail.com; henning@its-sby.edu.

PT. XXX, as our research partner, is an international company which has sea fish processing as their core business. They have some information system such as ERP, but they have no security policy about digital data dissemination. To exchange information among departments, company staff used document sharing systems. They login to the LAN (Local Area Network) using department account, create a file in their computer then upload it to the File Server.

The other employee having account to the File Server can copy, change or delete the documents in the File Server via File Copy Station without any further authentication and authorization process. The digital document in PT. XXX has highly risk to leakage. If there is an indication of document leakage, the company will hard to know the person who spread the document. Our research contribution is to trace the source of document leakage. When the person copy the data, the username and time timestamp are hashed and embedded in the document. The server also save the data.

1.1. Digital Documents

Document is a tool of information exchange between one person to another or from one group to another. The documentation process includes a variety of activities that begins with how a document is created, controlled, manufactured, stored, distributed and duplicated. Digital document is any electronic information that is created, forwarded, sent, received or stored in the form of analog, digital, electromagnetic, optical, etc., but not limited to text, sound, pictures, maps, etc. [1].

Digital documents are common in everyday life, organizations and businesses. So, digital data security and policy is important. There are a variety of techniques in securing a digital document including encryption, Digital Right Management, etc. The security techniques are varies according to the user requirements.

In our study case, PT. XXX has no digital data security and less security policy. To deal with the problem, we create the intermediate system to secure the digital data. Digital document used in this study is the 2007 version of Microsoft Office document that consists of Microsoft Office Word, Excel and PowerPoint, and also PDF documents.

1.2. FUSE

FUSE (File system in User space) is a mechanism on the operating system that allows users to create a file system without changing the code in the kernel [2]. FUSE module only provides a bridge kernel interface. FUSE module can run on Linux, Unix, MAC OS. Implementation of FUSE in the python programming language is Fusepy. Fusepy is a Python module that provides a simple interface for FUSE implementation and Mac FUSE. Fusepy is created using ctypes. Fusepy can execute FUSE commands such as open, write, read, getattr, readdir, mkdir, chmod, etc.

1.3. Samba

Samba is open source software and free software that provides services for data sharing and printer services to users [3]. Because it is open source, Samba configuration can be changed according of the user requirements. Samba uses the rules SMB / CIFS for data transmission from the server towards the user and vice versa. Samba allows integration between the Linux server and Microsoft Windows users to exchange data and documents. Samba configuration is in the data file named smb.conf. Configuration in Samba is as follows: security configuration, usage history configuration, directory access configuration, IP and port configuration, database configuration, etc.

1.4. MySQL

MySQL is a multithreaded and multi-user database management system (DBMS) software. MySQL is the implementation of a relational database management system (RDBMS). MySQL uses SQL as the basic language to access data that is divided into three parts, namely DDL, DML, and DCL. In this research, MySQL is used as a repository of digital data copy history.

1.5. Hash Function

Hash Function is a useful function to compress or minimize a long string, such as sha1, MD4, MD5 [4]. In this research, MD5 hash function is used to maintain the document integrity.

2. Methodology

2.1. System Architecture

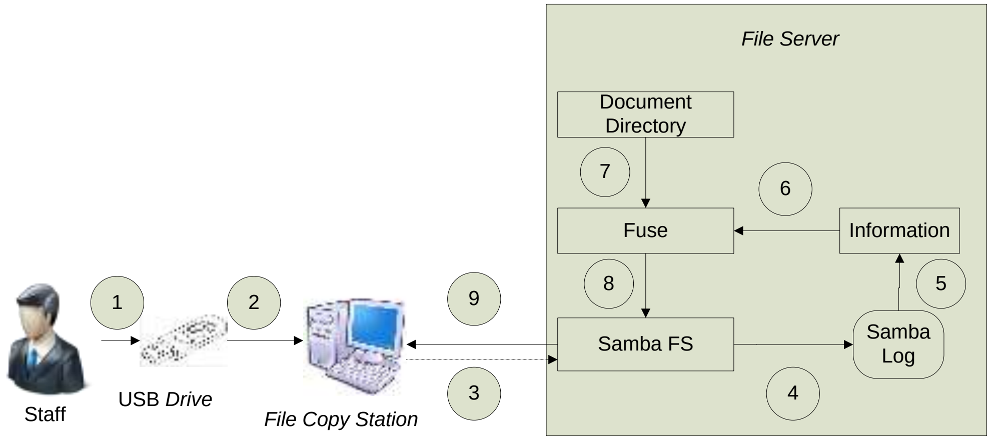


Fig. 1. File System Architecture

System architecture can be seen in Fig. 1. The steps are as follows:

1. Staff uses a USB drive to copy the data from the File Server via File Copy Station.
2. USB drive plugged into the File Copy Station.
3. Staff logs on File Copy Station in order to access the data on file servers via Samba.
4. After successful login, the login history stored in the Samba history data. Then staff can select the document to be copied to the USB drive.
5. All staff’s activities are recorded on the Samba history, including username and the time the document is copied.
6. The information consists of username, timestamp and document hash. They information is stored into the DB.
7. The information will be added to the copied document using File System.
8. The copied document will have additional information given by Samba File System.
9. Documents that have additional information will be sent to the File Copy station via Samba and transferred to USB’s staff.

2.2. Network Architecture

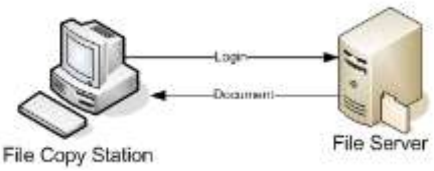


Fig. 2. Network Architecture

Network architecture can be seen in Fig.2. Company staff should login before accessing the File Copy Station (Client). After login, company staff can upload or download the document from the File Server. The File Server act as a document storage and store the information of username, timestamp and document hash in the Database.

3. Implementation

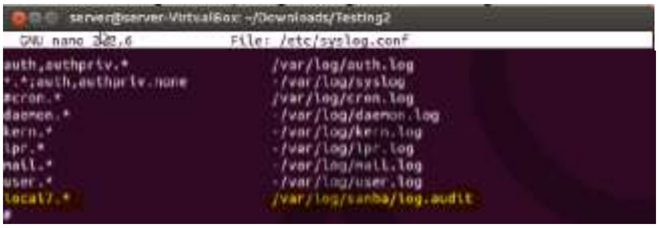
3.1. Server Implementation

The Samba configuration can be seen in Fig.3. The share path is in “/srv/samba/share/”. It saved the username (%u), IP address (%I), machine username (%m) and document name (%S).

```
[share]
        path = /srv/samba/share/
        browsable = yes
        read only = no
        guest ok = no
        valid user = @users
        force group = users
        vfs object = full_audit
        full_audit:prefix = %u|%I|%m|%S
        full_audit:success = pread open
        full_audit:failure = none
        full_audit:facility = local7
        full_audit:priority = notice
```

Fig. 3. Samba Configuration

The modification of *syslog.conf* is in the yellow line as in Fig. 4. The samba history will be saved in *log.audit* file. FUSE implementation has 6 functions: init function (to initialize server), _full_path function (to check the full path), getattr function (to get the dict attribute of the object), readdir function (to read the content of the directory), open function (to open the file), and read function (to read a file).



```
server@server-VirtualBox: ~/Downloads/Testing2
GNU nano 2.2.6              File: /etc/syslog.conf

auth,authpriv.*                  /var/log/auth.log
*.*;auth,authpriv.none           -/var/log/syslog
#cron.*                          /var/log/cron.log
daemon.*                         -/var/log/daemon.log
kern.*                           -/var/log/kern.log
lpr.*                            -/var/log/lpr.log
mail.*                           -/var/log/mail.log
user.*                           -/var/log/user.log
local7.*                         /var/log/samba/log.audit
#
```

Fig. 4. Syslog Configuration

3.2. Additional Information in Documents

To insert information in the Microsoft Office documents 2007 and 2010 versions we need to extract the document. We used core.xml in doc Props directory for this purpose. Core.xml consist of document properties such as document title, document subject, document author, keywords, document description, the last user that saved changes to the documents, revisions, document creation time, time changes in the document, document category and document content. The file system inserts additional information in a transparent manner. The system will read the username from Samba, get the hashed md5 of the document, get current date and time, add these data in the xml_tree of the document, and saved the changes on the original document.

For PDF document, the file system inserts additional information in the document metadata. But for doing so, the PDF document permissions should have writeable permission. The system will read the username from Samba, get the hashed md5 of the document, get current date and time, add these data in the metadata of the pdf document, and saved the changes on the pdf document.

3.3. Hash document using md5

Before information is entered into the copied document, the document is processed using md5 hash. The hash result then merged with username and timestamp of current date and time. The result then will be hashed again. The final data will be inserted into the copied document.

4. EXPERIMENTAL RESULT

4.1. Addition of Information on Documents

The original and copied Microsoft Office documents and PDF documents file size can be seen in Table 1. The average sizes of Microsoft office documents almost twice as much as the original document because the system adds the hash function of original document, while in the pdf document has no difference since it only add the information in metadata. The information added in the document contain username and timestamp, so it can show the source of leakage document.

4.2. Transfer Rate And Time

The transfer rate and time with and without file system can be seen in Table 2. The experimental used document doc1.docx (9,407,110 bytes). The system will transferred 10, 20, 30, 40, 50 and 60 documents respectively. The average speed of copying data without file system is 4,72 MBps, while with file system is 3.02 MBps. The average time of copying data without file system is 1,90 second, while with file system is 2.97 seconds.

The transfer rate for the document with file system security system is lower than the one without file system security because the document which is transferred is bigger. It takes longer time because the file system should record the activity in the server and in the same time embeds additional information and hash the document.

Table 1. Document Size of Original and Copied Document

No.	Document Name	Document Size (Byte)		% Difference
		Original	Copied	
1	doc1.docx	6,719,337	9,407,110	40%
2	doc2.docx	1,830,798	2,335,877	28%
3	doc3.docx	487,483	1,222,085	151%
4	doc4.docx	295,195	849,670	188%
5	doc5.docx	20,065	123,291	514%
6	excel1.xlsx	19,832	77,316	290%
7	excel2.xlsx	15,087	47,950	218%
8	excel3.xlsx	12,790	36,749	187%
9	excel4.xlsx	13,316	43,168	224%
10	excel5.xlsx	17,216	68,595	298%
11	ppt1.pptx	449,621	1,100,024	145%
12	ppt2.pptx	597,389	721,270	21%
13	ppt3.pptx	59,015	133,534	126%
14	ppt4.pptx	49,472	122,132	147%
15	ppt5.pptx	77,867	203,732	162%
16	pdf1.pdf	217,997	217,997	0%
17	pdf2.pdf	697,391	697,391	0%
18	pdf3.pdf	4,035,465	4,035,465	0%
19	pdf4.pdf	5,608,148	5,608,148	0%
20	pdf5.pdf	336,696	336,696	0%

Table 2. Transfer Rate and Time

Document Number	**Transfer Rate (MBps)**		**Time (s)**	
	Without Filesystem	**With Filesystem**	**Without Filesystem**	**With Filesystem**
10	4,112	2,436	21,818	36,820
20	4,168	2,891	43,051	62,074
30	4,699	3,255	52,271	82,674
40	4,991	3,115	71,893	115,187
50	4,867	3,105	92,166	144,462
60	4,786	2,948	112,667	183,581

5. CONCLUSION

The conclusions in this article are as follows:

1. Seamless document tracking system can be implemented using FUSE and Samba File System to secure digital document in a company.
2. The average size of the digital document is almost twice as big as the original document. The average speed of copying data with file system is almost 36% smaller than without file system, while the average time of copying data with file system 56% slower than without file system. Document size and transfer rate are the trade off the security using file system.

Suggestion for further research is to minimize the file size. This may happen if the document that is hashed is not all part of the document, but just the first few pages and the document properties.

References

[1] Adobe, "Adobe PDF", http://www.adobe.com/products/acrobat/adobepdf.html, accessed 14 June 2014.

[2] P. Central, "Python Central," 16 May 2013, http://www.pythoncentral.io/hashing-files-with-python/, accessed 2 July 2014.

[3] Cisco, "Cisco Support Community," https://supportforums.cisco.com/discussion/9551901/what-logging-facility-local7, accessed 18 May 2014.

[4] L. Foundation, http://www.linuxfoundation.org/what-is-linux, accessed 17 June 2014.

[5] S. Team, "Samba," Samba, http://www.samba.org/samba/docs/man/, accessed 25 March 2014.

[6] M. Office, "Microsoft Office," http://office.microsoft.com/en-001/word-help/office-open-xml-i-exploring-the-office-open-xml-formats-RZ010243529.aspx?section=3, accessed 14 June 2014.

[7] SourceForge, http://fuse.sourceforge.net/, accessed 25 March 2014.

[8] "lxml,", http://lxml.de/, accessed 18 June 2014.

[9] B. Hariyanto, Sistem Operasi, Bandung: Informatika, 2009.

[10] Samba, "Samba," https://www.samba.org/samba/docs/man/manpages-3/vfs_full_audit.8.html, accessed 16 Juni 2014].