DESIGN OF THE MAIN DATA SEPARATION SYSTEM IN POWER POINT DOCUMENT

Seung Ju Jang College of ICT Engineering, Dong-Eui University Korea sjjang@deu.ac.kr

ABSTRACT

This paper designs an access control system for the PowerPoint (MS PowerPoint) document system. The system designed in this paper is to analyze the PowerPoint document structure and modify the information related with the document, so that it blocks access of the user who does not have rights to access the modified information. This ensures the safe use of PowerPoint documents. The proposed idea in this paper are actually implemented and tested. As a result of the experiment on the proposed idea of this paper, it confirmed that the operation is normal.

Keywords: File Separation, Power Point Document, Security.

INTRODUCTION

Document-related functions have become an indispensable function in our day-to-day business operations. There are many document systems, among which documents related to presentations are often used by many people. In particular, Microsoft PowerPoint is widely used as a presentation document.

Documents stored for presentation may be less important than other documents, but they often contain important data. Therefore, security is important in storing documents for presentations.

In recent years, everything you write in your work is stored as digital documents. In particular, the presentation of the company's technical contents and business related contents is saved as digital documents. The management of such digital documents is becoming more important as the number of users increases.

Most users around the world are using Microsoft PowerPoint to create presentations. Document security for Microsoft PowerPoint is also becoming important. In Microsoft 2003 and earlier versions, the Microsoft PowerPoint file format was stored as a compound document file. 2007 and later versions were saved as Office Open XML (OOXML) files.

Recent studies on document security are under way, focused on security technology through document encryption and centralized management of documents. This paper designs the main data separation system when using the MS-PowerPoint document which is the most used in the presentation of data worldwide. Chapter 2 describes the related studies, Chapter 3 describes the design of the MS-PowerPoint document, and Chapter 4 describes the experiments and results of the designed system.

RELATED RESEARCHES

PowerPoint files are widely used throughout the world as a program to organize presentations. Nevertheless, research on the structure and security of PowerPoint files is not very active. PowerPoint files often contain important content that is edited, saved, and archived [1, 2, 3, 7].

MS PowerPoint files prior to 2007, have internal format content types for storage (for example, text, images, templates, etc.) and use compound document files in external formats. MS PowerPoint files consist of a record tree. If there is another record in the record, it is called the "container." If the record contains data, this record is called "atom." Records cannot have both records and data [4, 5, 6].

The record contains the record header and its data, and the record header contains the type and length of the record. Fig. 1 shows the container containing the other records.

Record Header	Record Header	Data (40 bytes)	Records
This is a container. Type: Document Length: 964 bytes	This is an atom. Type: DocumentAtom Length: 40 bytes		

Figure 1 PPT Data in a Container

MS PowerPoint applications store the basic contents of the file into a PowerPoint document stream. Fig. 2 shows a simple representation of the structure. The example file shown in Fig. 2 is structured as follows:

- 1 MasterSlide object;
- 2 more Slide objects;
- 2 more Picture objects and OLE objects.



<PowerPoint Document Stream>

Figure 1 Structure of PowerPoint Document Stream

Generally, PowerPoint files are executed in the order shown in Fig. 2. Existing studies are mainly focused on forensic research on PowerPoint files. They are conducting researches that are used for forensic analysis, using various information managed within the PowerPoint file. And they are analyzing contents deletion and modification information managed in the PowerPoint file. There are also research papers that examine deleted or edited contents of files using residual information generated by the " allow fast saves " function.

DESIGN of MAIN DATA SEPARATION SYSTEM

In this paper, we design the main data separation system from the PowerPoint file. We design the access control system for separating the main data from PowerPoint file. In the case of a PowerPoint file, it is an OOXML(Office Open XML) document. For this document file, one package is a generic ZIP archive containing the package's content-type, relationship, and part items. The package-relationship ZIP item stores the relationship information between the part and the package, and the part-relationship ZIP item stores the relationship information between the various parts [8, 9].

Fig. 3 shows the actual look within the ZIP structure of the power point file. The power point document in the ZIP file format contains compressed parts and related items in a container called a package.



Figure 3 Internal Structure of ZIP File

Design of Separation System

The MS PowerPoint file stores the document contents in the divided parts rather than in a single part. The slides folder contains the actual data in the PowerPoint file. This folder contains the data that is written in the PowerPoint.

In this paper, we design a system to protect core data in the PowerPoint. We design a structure to separate core data from the original file to protect core files. The following is a look inside the file, after decompressing the PowerPoint file.



Figure 4 Folder Aspect after Decompressing PowerPoint File

After the decompression of the PowerPoint file shown in Fig. 4, the folder where the important data is stored in the folder is shown in Fig. 5.



Figure **5** Main Data Folder within the PowerPoint File

As shown in the Fig. 5 above, the PowerPoint file structure contains data files in the slides folder in the ppt folder. This slides folder should be separated from the PowerPoint file to ensure safe use of PowerPoint.

In the above PowerPoint file, we separate the slides folder from the original PowerPoint file. The structure of the PowerPoint file after the separation is as Fig. 6.

이름	수정한 날짜	유형
rels	2018-08-30 오후	파일 폴더
🖭 slide1.xml		XML 문서
📄 slide2.xml		XML 문서
🖭 slide3.xml		XML 문서
🕋 slide4.xml		XML 문서
📄 slide5.xml		XML 문서
📄 slide6.xml		XML 문서
📄 slide7.xml		XML 문서

Figure 6 Separation File of Key Data

The flow of the proposed system design is shown in the following Fig. 7.



Figure 7 Process of PowerPoint Separation and Merging

The Fig. 7 diagram is the proposed system structure that separates and merges the main data from the PowerPoint file. We separate the power point module and the data module from the power point file through the separation module. By separating the data module from the PowerPoint file, you can safely protect important data.

Separate data modules from the PowerPoint file can be stored separately for safe data management. If you want to use separate data from the PowerPoint file, you must merge the data module into the PowerPoint file. The merge module allows you to merge the data module into the original PowerPoint file.

In the important data separation system proposed in this paper, the slides folder should be put in the original file, when users want to use PowerPoint. After adding the slides folder to the original PowerPoint file, the internal structure of the PowerPoint file is shown in Fig. 4. By adding the folder containing important data, the user can use it like the original PowerPoint file.

EXPERIMENT

This paper uses MS PowerPoint version 2013 for experiments. Experiments on the design of the main data separation system are performed in the proposed PowerPoint file. We prepare a PowerPoint file for the experiment. Using the prepared PowerPoint file, we first change the extension name of the PowerPoint file from pptx to zip.

The contents of the original PowerPoint file are shown in the following Fig. 8.

```
PPT 보안 테스트(1)

• PPT 보안 테스트

• PPT 보안 테스트
```

Figure 8 Contents of PowerPoint File

We use a test file to test the design of the main document separation system in PowerPoint. The file for testing is shown in Fig. 8 above. We change the file extension from pptx to zip, to separate main document of the PowerPoint. After changing the extension to zip and decompressing it, the inside of the PowerPoint file is the same as the above Fig. 4.

In this compressed file, the PowerPoint folder containing the main document content is ppt / slides. Inside this folder, you can see slides1.xlm through slides7.xml. When the slides2.xml file is extracted and opened, the contents are as follows Fig. 9.

문 물여명기 ⁶ 월 - 중	분 레이아웃 ♥ 에 다시 설정 사 슬락이드 ♥ ♥ ♥ ♥	· · · · · · · · · · · · · · · · · · ·	n [*] №		→ ····································	▲ 도형 재우기 · ● 추가 ☞ 도형 유곽선 · 기· ☞ 도형 효과 · ↓ 선택 ·
클립보드 13	슬라이드	글꼴	5 단락	6	그리기	5 편집
1						
2						
3			제목을	입력하	십시오	
4						
5						

Figure 9 Figure of Opening File after Separating Main Data

You can see that the contents of the original ppt file are stored in xml format in the slides2.xml file. When you open the ppt file after doing so, the screen is shown Fig. 10.

타일(F)	편집(E)	₩2I~>	즐겨찾기(A)	- エイの	도움말(H					
<u></u>	🚮 zum	Google -			Q	🔎 🚖 🚍 2	1호간	1 캡처	- 🔀 번역	▼ 🗾 퀵전송
		<td></td> <td>TOTTY - N</td> <td>o nere onre</td> <td> Onceo</td> <td></td> <td></td> <td></td> <td></td>		TOTTY - N	o nere onre	Onceo				
		<td>dy</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	dy							
	<td>:sp></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	:sp>								
	- <p:< td=""><td>sp></td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></p:<>	sp>	0							
		< p:rvap	NyPr name-"L	비용 개체 :	= 2" id-";	s=/>				
		pid	NVSpPr>							
			a:spLocks not	arp-"1"/s	-					
			NVSpPr>							
			piph idx="1"/	/ ==-						
			nvPr >							
		<td>Pr></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>	Pr>							
	-	< p:sppr/								
		<a:b< td=""><td>odyPr/></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></a:b<>	odyPr/>							
		- a:ls	tStyle/ >							
		- <a:p< td=""><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></a:p<>								
			airPr land	- Cen-US	" emtClea	- "O" dirty		alti ang - "	KO-KR"/>	
			-a:t>PPT							
		<	aira							
			air>							
			- ait - 12 21	a / -	it.	anty	- 0	anceange	en-os /	
		~	/a:r>							
			a:endParaRPr	lang = "er	n-US" smt	Clean="0"	dirty=	"O" altLar	ng="ko-KR	·"/>
		a.p	a.r.							
			<a:rpr lang<="" td=""><td>- "en-US</td><td>" smtClea</td><td>n-"o" dirty</td><td>- "o" -</td><td>altLang-"</td><td>ko-KR"/></td><td></td></a:rpr>	- "en-US	" smtClea	n-"o" dirty	- "o" -	altLang-"	ko-KR"/>	
			<a:t>PPT</a:t>							
			/a:r>							
			airPr lang	- "ko-KR	" smtClea	- "O" dirty	- "0"	altLang = "	en-US"/>	
			<a:t> 2.21</a:t>	테스트 <td>1 : t >-</td> <td></td> <td></td> <td></td> <td></td> <td></td>	1 : t >-					
			/acr>							
			-							
			arra							
			<a:rpr lang<="" td=""><td>- "en-US</td><td>" smtClea</td><td>n-"O" dirty</td><td>— • • •</td><td>altLang-"</td><td>ko-KR"/></td><td></td></a:rpr>	- "en-US	" smtClea	n-"O" dirty	— • • •	altLang-"	ko-KR"/>	
		_	a:t>PPT							
			at the second							
			<a>a:rPr lang	- "ko-KR	" smtClea	n="0" dirty	<u> </u>	altLang = "	en-US"/>	
			<a:t>보안 (</a:t>	테스트 <td>1:t >></td> <td></td> <td></td> <td></td> <td></td> <td></td>	1:t >>					
		-/	airs							
		- <a:p< td=""><td>28-</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></a:p<>	28-							
			aira	-						
			<a:rpr lang<="" td=""><td>- en-Us</td><td>smtClea</td><td>- O dirty</td><td></td><td>altLang = "</td><td>RO-KR-/-</td><td></td></a:rpr>	- en-Us	smtClea	- O dirty		altLang = "	RO-KR-/-	
		-	(aura)							
			49.1 6 20							
			<a:rpr lang<="" td=""><td>- KO-KR</td><td>" smtClea</td><td>- O' dirty</td><td>- "0"</td><td>altLang = "</td><td>en-US"/></td><td></td></a:rpr>	- KO-KR	" smtClea	- O' dirty	- "0"	altLang = "	en-US"/>	
		_								
		~/au								
		a:p	-							

Figure **10** Open Appearance of Separated Key Files

After separating the main data from the PowerPoint file and saving it as a PowerPoint file and opening the file, it is the same as Fig. 10. You can see that the main data is separated from the PowerPoint file. If you add the separated main data to the PowerPoint file, the result of the experiment is shown in Fig. 8. As a result of this experiment, it is confirmed that the main data can be added to the PowerPoint file, if the user wants to use the data after separating the main data from the PowerPoint file.

CONCLUSION

This paper designs a main data separation system for PowerPoint files which are the most commonly used in the world. PowerPoint files are widely used to generate and store presentation materials. In this paper, we analyze the structure of the PowerPoint file and design a system that separates the main data within the PowerPoint file, based on the analyzed structure. By doing this, you have the advantage of keeping separate important data stored in the PowerPoint file.

The main data separation system in the PowerPoint file utilizes that the structure of OOXML in the PowerPoint file. PowerPoint files in OOXML are stored in compressed format. You can uncompress the file stored in compressed format to find the main data file. If you extract

it from the PowerPoint file, the folder containing the main data is ppt / slides. This folder contains related resources of the PowerPoint.

It locates the ppt / slides folder in the PowerPoint file. It separates the files in this folder from the PowerPoint file. The main xml file is separated from the PowerPoint file and saved. Then, the original compressed file is saved. Then it will be created like the original PowerPoint file. When using the generated PowerPoint file, data cannot be read normally.

REFERENCES

- [1] Youn Ji Hye (2013) "Methods for investigating of edit history about MS PowerPoint files that using the OOXML formats", Department of Information Management & Security Graduate School of Information Management & Security Korea University.
- [2] Youn Ji Hye, Park, Jeong Heum, Lee Sang Jin, (2012) "Methods for investigating of edit history about MS PowerPoint files that using the OOXML formats", Journal of Information Processing Society C, vol 19-C.
- [3] Hong Thai Pham1, Huan Buu Nguyen2, (2018) " Teachers' Perceptions about PowerPoint Use as an ICT Tool for Teaching Vocabulary in Vietnam ", European Journal of Foreign Language Teaching, Volume 3, Issue 4.
- [5] Mark S. O 'Hare, Coto De Caza, (2018) "Secure File Sharing Method and System United States Patent", US 10, 027, 484 B2, CA(US); Rick L. Orsini, Flower Mound ,TX (US); Don Martin, Chesterfield, VA (US)
- [6] Vivek Goyal, Daniel Walsh, David Howells, Miklos Szeredi, (2018) "SUPPORTING SECURITY ACCESS CONTROLS IN AN OVERLAY FILESYSTEM", US 2018 / 0239921 A1.
- [7] Jennifer Kelly, Robert Carr, Raymond Kelly , Seyed Safakish, (2018) "SYSTEM AND METHOD FOR PROVIDING A SECURE NETWORK", US 2018 / 0337975 A1.
- [8] Ida Bagus Ary Indra Iswara1, I Ketut Sudarsana, Nurlaidy Joice Simamora etc, (2018) "Application of Data Encryption Standard and Lempel-Ziv-Welch Algorithm for File Security", *International Journal of Engineering & Technology*, 7 (3.2)783-785
- [9] Jang, Seung Ju, (2018) "Design of the Access Control System for MS-WORD Document System", Journal of the Korea Institute of Information and Communication Engineering, vol. 22, No. 10, pp. 1405-1411.