Implementation of Huffman Algorithm on Digital Signature to Prove The Ownership Portable Document Format

R. Reza El Akbar[1], Rohmat Gunawan[2], Firman Setiawan[3]
reza@unsil.ac.id[1], rohmatgunawan@unsil.ac.id[2], 147006136@student.unsil.ac.id[3]

Informatics Department Engineering Faculty Siliwangi University[123]

Abstract. Portable Document Format (PDF) is one type of file that is frequently exchanged over the internet. Proving ownership of PDF documents is important so that they are not used by irresponsible people. So that proof of ownership of PDF documents can be done, this research proposes applying digital signatures. Digital signatures are generally inserted in the body in the internal structure of the PDF. In this research try to insert a digital signature in the xref pdf section. Before the copying process is carried out, the Digital signature is compressed first with the Huffman algorithm to optimize data storage capacity in each xref line and the inserted data is not easily perceived. The experimental results in the study showed that the size of the digital signature data that had been compressed using the Huffman algorithm decreased by an average of around 16.25% compared to before it was compressed. Digital signature data entered in the xref section is not easily perceived and does not increase the size of the pdf file.

Keywords: Digital Signature, Huffman, PDF.

1 Introduction

The internet is one of the commonly used digital archive distribution media. Some document formats that are often searched for and exchanged via the internet include: PDF, DOCX, XLSX, PPTX, EPUB, ODT, TXT / RTF [1]. According to the survey report [1], in February 2014, the percentage of Portable Document Format (PDF) usage reached 77.3%. Based on these data PDF is one of the most widely document formats exchanged via the internet. The increasing use of the PDF format has raised various problems related to file security aspects. Copyright protection against PDF is one problem that needs to be considered, especially to prove ownership of documents. Proof of ownership of important documents is done so as not to be misused by irresponsible parties. One way to prove ownership of files is by applying digital signatures. Digital signature is one method of file security that can be used to detect unauthorized data modifications and authenticate file ownership identities [2], [3]. In addition, recipients can use digital signatures as evidence to third parties that the signature is in fact, produced by the claimed signatory[4], [5], [6].

Several studies related to the technique of proving ownership of pdf files have been done before, including: digital signature insertion in the xref pdf section [7], the use of watermaking inserted at xref pdf [8]. In study [7], insertion is done by overwriting 3 bytes character 20 ASCII or space characters in each row of xref. Inserted digital signature data cannot be perceived and does not add to the size of the file as long as the xref capacity is still sufficient. But in research

ICONISTECH-1 2019, July 11-12, Bandung, Indonesia

DOI 10.4108/eai.11-7-2019.2298023

[7], it still cannot handle data insertion that exceeds the capacity of xref. Whereas in research [8], digital signatures are inserted in each row of xref as much as 5 bytes which have previously been compressed using the RC4 algorithm.

In the present study, digital signature strings are inserted in the xref section as in research [7], [8], but before the insertion process is done, digital signature data is compressed first using the huffman algorithm. Compression can compress string size compared to the original data [9], [10], [11], [12]. This data compression process is to minimize the size of the digital signature that will be inserted and produce digital signature data that is not easily perceived.

2 Propose Method

The method proposed in this study consists of two main parts, namely the embed process and the extract process. The embed process consists of several stages as shown in Figure 1.

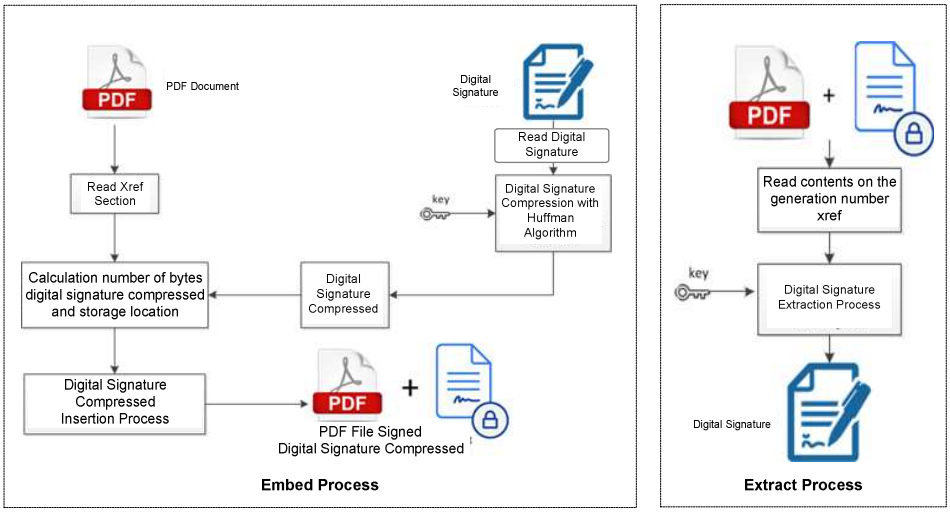

Fig.1. The proposed digital signature insertion method

2.1 Embed Process

1. Read *xref section*

Reading the Xref Section in a PDF file is a first step that must be done. This stage is done to find out the number of xref entries. The process carried out at this stage is as follows:

a. Read the generation number in the xref. Generation Number is one part of the internal PDF file structure. Generation numbers are plain text and can be perceived when PDF files are accessed with a text editor.

b. Get the number of lines of generation number. To get the number of rows of xref, pdf has provided the value of the number of rows of xref, shown in figure 2. There is a number 1014 which is the value of the number of rows of xref. To get this value, a byte reading starts from the text section containing the word xref. After finding number 0, the

value shows the number of rows of xref. The results of reading the value of the number of lines of the xref are then stored in a variable.

c. Read digital signature

At this stage, read the contents of the signature digital string to be compressed.

2. Compression of digital signature files with input keys. The next step is to compress digital signature files with the Huffman algorithm with the input key. This compression process is done to compress the digital signature file. The key is the key that is used as input to the compression process with the Huffman algorithm. The key is used to generate dictionary so that it will not produce identical compression even though it contains the same word.

3. Calculation of the number of digital byte signatures and storage locations with the input key. Before the digital signature is inserted, it calculates the size or number of bytes of the digital signature. This is done to find out the number of bytes to be inserted, and compared to the capacity of the available xref, so that all digital signatures can be inserted at the xref location. The steps taken in this process are as follows:

a. Calculates the number of groups of bytes from the digital signature file. For example: the file size of digital signature = 70 bytes, then the number of groups of bytes is 70/7 = 10 groups. From the results of these calculations, it can be seen that, at least 10 lines of storage must be available on the xref to hold all digital signature data.
b. Calculates the size of a compressed signature digital storage location. The size or storage capacity of a digital signature is determined by the number or number of rows of xref
c. Next, calculate the location of digital storage.

4. The process of inserting a digital signature. The process of inserting a digital signature is done by overwriting each xref byte specifically in the generation number section. The stages in the insertion process are as follows:

a. Read digital data signatures that have been grouped and stored in arrays
b. Write each byte generation number with compressed digital signature data.

The stages of the embed process produce a new PDF file that is embedded with a compressed signature digital string.

2.2 Extract Process

1. Read xref in various generation numbers. This process reads the digital signature string compressed in the xref generation number
2. Extract the compressed digital signature string using the Huffman algorithm with the input key.
3. This process will return the signature digital string that has been previously compressed to the original form as it can be perceived.

3 Result and Analysis

3.1 Data Needs Preparation

Before testing the embed and extract process. The first step is to prepare the pdf file document that will be used for the insertion process of the digital signature string and preparing the signature digital string to be used.

Table 1. PDF file used in the experiment

No	PDF File			
	File Name	**URL**	**File Size**	***Xref* (rows)**
1.	File1.pdf	http://jurnal.unsil.ac.id/index.php/jssainstek/article/download/26/28	147 KB	48
2.	File2.pdf	https://erizal.files.wordpress.com/2007/10/kompresi-teks-dengan-menggunakan-algoritma-huffman.pdf	60.1 KB	54
3.	File3.pdf	https://jurnal.unikom.ac.id/_s/data/jurnal/v08-n01/volume-81-artikel-9.pdf/pdf/volume-81-artikel-9.pdf	617 KB	2183
4.	File4.pdf	http://seminar.uny.ac.id/semnasmatematika/sites/seminar.uny.ac.id.semnasmatematika/files/A-12.pdf	276 KB	609
5.	File5.pdf	http://join.if.uinsgd.ac.id/index.php/join/article/viewFile/v2i22/69	683 KB	1109
6	File10.pdf	jurnal.unpad.ac.id/jin/article/download/8536/pdf	13.1 KB	23
7	File7.pdf	http://jurnal.unsil.ac.id/index.php/jssainstek/article/download/494/344	360 KB	934
8	File8.pdf	http://jurnal.unsil.ac.id/index.php/jssainstek/article/download/513/341	594 KB	746
9	File9.pdf	http://jurnal.unsil.ac.id/index.php/jssainstek/article/download/24/26	365 KB	71
10	File6.pdf	http://jurnal.unsil.ac.id/index.php/jssainstek/article/download/477/338	356 KB	616

Table 2 Digital signature data

No	Label	Digital Signature Text	Size of Byte
1	String1	Pdf	3 byte
2	String2	Algorithm	9 byte
3	String3	computer network	16 byte
4	String4	information and communication technology systems	48 byte
5	String5	Huffman algorithm implementation on digital signatures to prove ownership of PDF files	86 byte
6	String6	The development of technology is currently growing rapidly along with the development of the internet as a center of communication and rapid exchange of information	164 byte
7	String7	Algortima Huffman was developed by David A Huffman in a journal he wrote as a prerequisite for graduation at MIT by constructing a scheme that contains the frequency with which each symbol appears	196 byte

8	String8	Digital signature is a mechanism to replace the signature manually on a document paper message signature can be done in two ways namely by compressing the message by itself and providing a measure of authentication	214 byte
9	String9	pdf is a document format that is often used for flexible digital document exchange needs compared to other document formats With the increasing use of this PDf format copyright protection of PDFs is one of the issues that need to be considered to prove ownership	262 byte
10	String10	There are several methods applied The method used in this study consists of 2 main parts namely the embed process and the extract process This process consists of several processes Following are the stages carried out in the research and implementation of the RSA algorithm and the Huffman Algorithm to prove ownership of digital documents in PDF format	353 byte

In insertion experiments into documents, digital signature strings are prepared which have different lengths of byte beginning with string1 which has a number of bytes 9 and string5 has a byte length of 123. Strings used only contain letters and numbers, do not contain special characters in them.

3.2 Huffman Compression Testing

Digital strings used for testing have different lengths of byte to determine the success rate of the compression process. Table 1 shows the results of the compression data performed.

Tabel 3 digital signature compressed

No	Before compression		After compression		
	Label	**Size**	**Digital Signature Text**	**Size**	**Compressed (%)**
1	String1	3 byte	Ý	3 byte	0%
2	String2	9 byte	\×cFsÙ	7 byte	22.22%
3	String3	16 byte	Åu—éœ ˜ô¶Lö »	15 byte	6.25%
4	String4	48 byte	Ñl·cd Ñu½ ¶>‘]e š[DPg EÖôÎH §¶|êî› Ž™q–r Y”	40 byte	16.67%
5	String5	86 byte	eõÉd –çt] oK¥éœ òt7Ie ² zA\ ×cFsÙ Ò‹zGD Ôg}2 ¢k`ÏL dË¦wt ¿µ“¥ Û[&9O Eý.—¢ ¤—ÒZ+ “(	75 byte	12.79%
6	String6	164 byte	³< #" Y+ºû$ ·>—KÓ 9"žÛ« ºn:Q— H´Ç- Î¾:M ÛQlÝ1 ,ú#¯Ž Wu³t ÚŒçéœ òtŽMd ®ëì'Ü ú]/Lç “¥ç& 6ÉŸH2 éH¤· 91ét½ "ºË4¶ ^ Î‹- é±ôÆ è ¤› Å<Í' K¥éE² Ý gE ÖÀ	130 byte	20.73%
7	String7	196 byte	\×c AÐÝ% –È-é´ tŽMd ®ëä ¤ <t& h Ý Ò Yl,Þ” [Ò”Ý ¦6Á_I äé¶7g 'H2éKñ“0 ÒŒ¨ÎN 'Ý I± ,= Î‹ -é} 9ž ñÒ +-Ë<t <:->¤ 2Šy, “¦sÁŸ H®·8(·.™Ï' Iq“ $ ¶.:mF sôÚz" Ÿ¤ E? L¸²« é÷É9@	155 byte	20.92%
8	String8	214 byte	“DÖg}2¢k` ÎLdéF ] éd‘ O´eg LîéŒ— ×——— RtÎ y:eDÖ Áž˜ÉÒ È-Ò ë ã¥Öôƒ ¤wE¥’ [ŸKà¾ LzY&Y A4 2¢ k`ÏLd éô† N‘Ýl (·¦{] Óh8Ë¥ °Y%|t ‡Ž]e øÉ–T[7Lç“¥ ’e”I Ò-:Qž RW/H- ¥øÝ- ÑlÝ éd e¦ 2t°^ iœò[A [	170 byte	20.56%

9	String9	262 byte	ß½(Ë ¤#º- ,'ÜúK v6AŸL çƒ>"e Òés'Þ še$}% »-'×& ôC\ # ¢j3,¾ 'Ý–I n}$Þ) à¶i:[$'åÒ+ ¬¾d ¦wt»9 äÇ¤wE ¥'[ŸI nÆÈ3Ë £"g?L ç"¥Å Œ eE³ tÓ):] /Lç£. Š"/In ÆÈ3é ×ñ O ŸKñ»9 "Î‹-é t½$¼ ºQ—K- "¥ÒôÎ y:Q–Z I—Lçƒ >–É$} 3»¤2t Šër¢9 1'>™Ý ÒünÖN —ml˜å =Â	209 byte	20.23%
10	String10	353 byte	³<˜ÉÒ dé"š ÉŒê² Lçº<º A}õÑ$ }žN– Iœ÷GÓ L¤ ¥ ôÎz2é –zGÇH ®·*2Ï .—KÓ¿ K ¢Þ— ÁŽyt¶ $¯Ž™ Ï'I,† Hú_ Ñ I–] ¶ >™Ï'I 7Ï"Ï ¥øÝ™ eÑ™èË ¥øÝ™ eÒ+-Ê Œ³Ë¥Ò ôÊMdÆ ú_ Ñ I–ReÐ »«®í¨ ¶n c' Lç"¦Y Á4™tŠ q¢Hú]¦}(· ¦sÉÓ2'bŸ ¤ÇÒ‹ /®K$· 83¢ëz]/Lç" ¢ì Ò æ»3ž Î [L ç"¡ºK - [Ðj æ»3ž Î™ÝÒü nÖN— m l˜å=ôº^'Ñ 5Á_H î‹K$· <ºQoE I/¤·c dÀ	275 byte	22.10%
			AVG		16.25%.

Compression results in Table 3 are testing the compression of the Huffman algorithm by changing the byte length of the digital signature string to be shorter. For example, string 1 has a length of 9 bytes, after going through the compression process to 7 bytes. The result of compression will be a string that cannot be perceived or translated. The average percentage reduction in the compression process is 16.25%.

5 Conclusion

Based on the results of this research, conclusions can be taken as follows:

a. The length of the digital string signature signature that has been compressed using the huffman algorithm decreases in size by an average of 16.25%.
b. The digital signature string that is inserted in the xref section is not easy to perceive and does not increase the size of the pdf file.

References

[1] D. Johnson, "The 8 most popular document formats on the we," 2014. [Online]. Available: http://duff-johnson.com/2014/02/17/the-8-most-popular-document-formats-on-the-web/. [Accessed: 23-May-2018].

[2] M. Warasart and P. Kuacharoen, "Paper-based Document Authentication using Digital Signature and QR Code," *4TH Int. Conf. Comput. Eng. Technol.*, vol. 40, no. January, pp. 94–98, 2012.

[3] N. A. A. S. Al-Maweri, R. Ali, W. A. Wan Adnan, A. R. Ramli, and S. M. S. A. Abdul Rahman, "State-of-the-art in techniques of text digital watermarking: Challenges and Limitations," *J. Comput. Sci.*, vol. 12, no. 2, pp. 62–80, 2016.

[4] National Institute of Standards and Technology, "FIPS 186-4: Digital Signature Standard (DSS)," *Fed. Inf. Process. Stand. Publ.*, no. July, 2013.

[5] Edward J. Delp, "Digital Watermarking and Data Hiding," *Purdue Univ.*, no. 5, pp. 181–183, 2011.

[6] C. Lakmal, S. Dangalla, C. Herath, C. Wickramarathna, G. Dias, and S. Fernando, "IDStack - The common protocol for document verification built on digital signatures," *2017 Natl. Inf. Technol. Conf. NITC 2017*, vol. 2017-Septe, pp. 96–99, 2018.

[7] I. F. Al Shaikhli, A. M. Zeki, R. H. Makarim, and A. S. K. Pathan, "Protection of integrity and ownership of PDF documents using invisible signature," *Proc. - 2012 14th Int. Conf. Model. Simulation, UKSim 2012*, pp. 533–537, 2012.

[8] R. Gunawan and R. Munir, "Watermarking pada Cross Reference (XRef) Portable Document Format (PDF) dengan Enkripsi RC4," no. October 2015, pp. 66–70, 2015.

[9] A. Affandi, Saparudin, and Erwin, "The Application of Text Compression to Short Message Service Using Huffman Table," *J. Generic*, vol. 6, no. 1, pp. 19–24, 2011.

[10] S. R. KODITUWAKKU and U. S.AMARASINGHE, "COMPARISON OF LOSSLESS DATA COMPRESSION ALGORITHMS FOR TEXT DATA," *Indian J. Comput. Sci. Eng.*, vol. 1, no. 4, pp. 416–425, 2010.

[11] M. Sharma, "Compression Using Huffman Coding," *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 10, no. 5, pp. 133–141, 2010.

[12] K. Sailunaz, M. Rokibul Alam Kotwal, and M. Nurul Huda, "Data Compression Considering Text Files," *Int. J. Comput. Appl.*, vol. 90, no. 11, pp. 27–32, 2014.