International Journal of Pure and Applied Mathematics
Volume 117 No. 15 2017, 905-914
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
**url:** http://www.ijpam.eu
Special Issue

ijpam.eu

# DATA SECURITY APPLICATIONS WITH A COMBINATION OF CRYPTOGRAPHIC METHODS AND TINY ENCRYPTION ALGORITHM TWOFISH WEB BASED ON PT JAKARTA SIDOLA COMPUTER

**Ady Widjaja[1],Mujito[2], Chandra Subagja[3]**

**Universitas Budi Luhur**

ady_w168@yahoo.co.id, jitosalemba@gmail.com, chandra.subagja49@gmail.com

Abstract - This data security application designed to secure critical data at PT. Sidola Computer. The data in question is confidential data such as sales data and also data that contains product prices of principal for Sidola that should not be known by other parties, document tender document, system configuration and infrastructure belonging to the customer as well as many other documents are confidential, it is often also the data sent using email. With so many important data often confidential, then the data become vulnerable with data theft, data manipulation or intercepts email. To face these problems then used the application safeguards data by the method of Twofish cryptographic algorithms and Tiny Encryption Algorithm (TEA). The application is built, there is an additional facility, namely the process of login to be able to access it, so that only certain parties who have access rights can use this application. As well as authentication and data integrity protection, because it uses the SHA1 checksum with the method. The programming language used to build the application security data this is the PHP programming language that is web based. The results of testing this binary cryptography, data can be secured in order to avoid double attack cryptanalysis, and be guaranteed the authenticity of the data because the process has a checksum. It's just that for the size of the data results of the process encrypt will be increased from the size of the original data and also the program can not encrypt on some data in the same time.

Keywords: twofish, TEA, SHA1 checksum, encrypt, cryptography.

## 1. Introduction

PT. Sidola Computer is a company engaged in the reseller of hardware, software and IT services. Thus, as a reseller of many confidential data such as sales data and also data that contains product prices of principal for Sidolathat should not be known by other parties, document tender document, system configuration and customer-owned infrastructure as well as many other documents are confidential, it is often also the data sent using email. With so many important data with different file formats, then the data become vulnerable with data theft, data manipulation or intercepts email. To deal with these problems, then implemented data security application with the Twofish cryptographic algorithm and Tiny Encryption Algorithm (TEA) that can secure data with different file formats and apply a special extension. SC and checksums to ensure the authenticity of the data. Implementation this application is restricted to the operating mode on the algorithm Twofish and the TEA is to use mode Cipher Block Chaining (CBC), a big key for the application data security can only be 16 characters and the optimum of the files can be processed by the application is 7168 KB (7 MB). For files with size more than 7 MB can still be processed just takes quite a long time.

## 2. The Cornerstone Of The Theory

Twofish cryptographic algorithm is a block cipher with a block length of 128 bits key length varied [1]. The length of the key to twofish can have a length of 128 bits, 192 start bits and 256 bits, but this implementation is used with key length 128 bits. Tiny Encryption Algorithm (TEA) is a cryptographic algorithm is a simple but fast and is a strong cryptography [11] This TEA designed by Wheeler and Needham. TEA is a cryptographic block cipher known as simple cryptography in the description and implementation. TEA is a block cipher with a block length of 64 bits and a key length of 128 bits.

### 2.1 The Twofish Algorithm

Twofish algorithm is an algorithm that until recently was declared safe because there is still no true kriptanalisis attack – can actually break this algorithm. This algorithm is also not patented encryption tools that use it so that it does not have to pay the costs. At twofish, Input and output data in the-XOR-kan with 8 is K0, ..., K7. The XOR operation is called whitening of input and output. The function F has five types of component operation: the function of rotation to the left by 8 bits, S-Box, MDS Matrix (Maximum Distance Separable), IPM (Pseudo-Hadamard Transform), and two addition modulo 232 is as shown in Figure 1. The algorithm twofish are also a function of G which is the core of the twofish algorithm ari. The function G is done twice in a single process of counting function f. Rounds on twofish is as many as 16 rounds. The process encrypt algorithm twofish as a whole are as in Figure 1 below:

Figure 1: structure of the encrypt algorithm twofish

## 1. S-Box

At twofish, each S-Box has three 8-by-8-bits fixed permutations for the algorithm with 128 bits long keys chosen from two sets of permutations that are possible, q0 and q1. The input of the S-Box of this magnitude is of 32 bits or as many as four words per process S-Box. Among the three 8-by-8-bits fixed permutation, the operation is performed with the XOR S i.e. S0 and S1. S this is obtained from the calculation of the key schedule and its value is fixed for good in the process encrypt and decrypt. The process of calculating the S-Box is as in Figure 2:

Figure 2: process of S-Box algorithm twofish

### b. Permutation Q

Permutation q0 and q1 are the permutations fixed by the value 8-bits. The permutation is the major component of the S-Box on thetwofish algorithm. For 8 bits input x, the result is y resulting from the following process:
A0 = [x/16] and b0 = x mod 16
A1 = a0 b0 XOR
B1 = a0 XOR ROR (b0) XOR (8 a0 mod 16)
A2 = t0 [a1]
B2 = t1 [b1]
A3 = a2 XOR b2
B3 = a2 XOR ROR (b2) XOR (8 a2 mod 16)
A4 = t2 [a3]
B4 = t3 [b3]
y = 16b4 + a4
The value of t0, t3, ... already fixed depending on permutation.
If the permutation q0, t0, ..., t3 is:

Table 1: table of permutation q0

If permutations of q1, the value t0, ..., t3 is:
Table 2: table of permutation q1

**c. Function F**

Function-F works on 64 bits. F-function has three arguments, that is two words input P0 and P1 are each 32 bits, and round number r is used to select K to how that is to be used, is being used is is started from K to 8th, because is 0 to 7 was used for whitening process. Here is the formula of automated calculation of functions F:

T0 = g (R0)
T1 = g (ROL8 (R1))
F0 = (T0 + T1 + K2r + 8) mod 232
F1 = (T0 + 2T1 + K2r + 9) mod 232

**d. The Functions G**

G-function is the core of the twofish algorithm. On the function of this input-G is the Word X with length of 32 bits (4 characters). Further input is broken into 4 parts each of 8 bits (1 byte). Each of the bytes are processed by each key-dependent S-Box him. Each S-Box is a correspondence (bijective) that process the input of 8 bits and produce 8 bits output [3]. Then the four S-Box results are interpreted as a vector with a length of 4 GF (28), and then multiplied by 4 x 4 Matrix MDS (using GF (28) for the calculation). The result of the multiplication of MDS Matrix is then interpreted into 32 bits of output that results from a function-G.

Figure 3: function G in twofish

**e. MDS Matrix**

MDS (Maximum Distance Separable) is a matrix that contains the bytes that are multiplied by the vector of 4 bytes result S-Box. For the process of matrix multiplication in MDS this does not use the usual multiplication but uses Galios Field GF (232) and the primitive polynomial x 8 + x 6 + x 5 + x 3 + 1.

Figure 4: the process of calculating MDS Matrix

### 2.2. Algorithm of Tiny Encryption Algorithm (TEA)

TEA is the fiestel cipher mode of operation that uses XOR, ADD and SHIFT. TEA using 64 bits block size and use 128-bits key and do a 32 round the same process. TEA is a cipher that uses the iteration with variable i, each round has a input y [i-1] and z [i-1] obtained from the previous round. K is for [i] obtained from 128 bits key and using the delta (∂). For delta is obtained from the ratio of gold (golden number ratio) is difficult to ascertain. The value of delta (the golden ratio) was obtained from the following functions: = ∂ (√ 5 – 1) * 231 = 9E3779B9. In the process it uses to encrypt TEA block ciphers with huge block 64 bits, and 64 bits are then in for two became the y and z as input on fiestel network algorithm of TEA, then the input is then processed by the following operations:

y [i] = y + (((ROL4 z) + K0) ^ (sum + z) ^ ((ROL5 z) + K1))
z [i] = z + (((ROL4 y) + K2) ^ (sum + y) ^ ((ROL5 y) + K3))

The i variable indicates the iteration (rounds) to how the process is going on. The value of the sum of the initial round is equal to the value of delta that is 9E3779B9, and for the next round of sum = sum + delta, where this sum will be continually added to the delta suit his iteration. The last process is the merge back the y and z last in split and this became the output ciphertext form. All the process of adding in TEA using modulo 232, so the results of the calculations are nothing more than 32 bits.
The structure of the algorithm the process encrypt TEA is like in Figure 7 below:

Figure 5: structure of algorithms encrypt TEA

### 2.3. Base64 Algorithm

Base64 transformation is one of the algorithms for Encoding and Decoding a data into ASCII format, based on the number of base 64 or could be said to be one of the methods used to perform the encoding (encoding) against the binary data. The resulting characters in the Base64 transformation consists of A. .. Z, a..z and 0 .. 9, as well as the last two characters plus the symbolized IE + and/as well as one of the characters is equal to (=) used for adjustment

and binary data or fulfill the term referred to as charger pad. The actual Base64 encoding technique is simple, if there is one (string) bytes that will be encoded to Base64 then the trick is cited by the Revelation, et al [6]:
1. Broke a string of bytes to per-3 bytes.
2. Combine 3 bytes to 24 bits. With note 1 bytes = 8 bits, so 3 x 8 = 24 bits.
3. the Last 24 bits are stored in the buffer-(unified) was broken up into 6 bits, then it will produce 4 fractions.
4. Each fraction is converted into decimal values, the maximal values of imana 6 bits is 63.
5. Lastly, make the value of the decimal value of the index to choose the constituent characters from the base64 and the maximum is 63 or 64 to index. And so on until the end of the string of bytes that we want to convert. If it turns out that in the process of encoding there is the rest of the divisor, then add the rest of the finisher as the character =. Then sometimes on base64 will appear one or two character = (equal to).

### 2.4. Operating Mode of Ciphaer Block Chaining (CBC)

Operating mode Cipher Block Chaining (CBC) is a block cipher mode of operation that uses the initialization vector (vector initialitation/IV) with a certain size (the size is the same as the one block plainteks). On the mode of operation is plainteks divided into several blocks, each block is then encrypted with the provisions of the first plainteks encrypted blocks first. Before the plaintext is encrypted, the-XOR with the IV. Then, the results of the XOR encrypted to produce the ciphertext. Furthermore, the ciphertext is used as the encoding process IV for the next plaintext block.

### 2.5. Algorithm1 Secure Hash (SHA1)

SHA is one example that can be used for the checksum. SHA is a one-way hash function created by NIST and used with DSS (Digital Signature Standard). By NSA, SHA expressed as a standard one-way hash function. SHA is based on MD4, created by Rivest of MIT. These algorithms receives input in the form of a message with a maximum size of 264 bits and produces a digest that is 160 bits longer than MD5. SHA1 (Secure Hash Algorithm1) was a type of 160 bit hash function which is the replacement of MD5 (Message Digest) that its 128 bit hash. SHA1 and MD5 hash method is the most widely used [9]. SHA1 produces a 40 character SHA1 encryption, thus giving more randomization compared to MD5.

## 3. The Design Of The System And Applications

The design is a process that is done to merancangan the application. The design of the system made in General is to encrypt and decrypt method using Twofish cryptographic algorithms and Tiny Encryption Algorithm-based web.

### 3.1. Twofish

Flowhart the process encrypt algorithm twofish is as shown in the following figure 6 :

Figure 6: Flowchart encrypt twofish

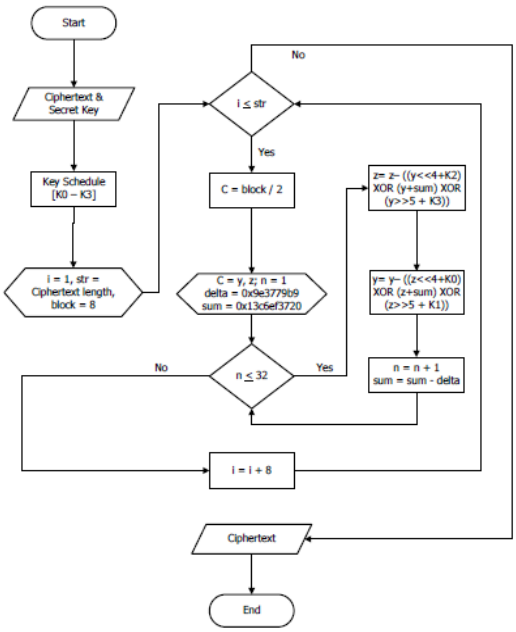
While decryptnya is a process flowchart as shown in Figure 7 below:

Figure 7: Flowchart decrypt twofish

### 3.2. The Tiny Encryption Algorithm (TEA)

Flowchart process encrypt algorithm of TEA is as shown in Figure 8:

Figure 8: Flowchart encrypt TEA
As for the flowchart of the process decryptnya is as shown in Figure 9:




Gambar 9: Flowchart decrypt TEA

### 3.3. The process Encrypt

The process encrypt is starting the process of input, for this there are two input options i.e. input file or input text, and then enter the secret key, then do the process encrypt (encrypt Twofish, encrypt and encode Base64) of TEA, after completed the process to encrypt the file then the file with the the initial extension will be changed to the file with the the extension. sc. As shown in Figure 10 below:

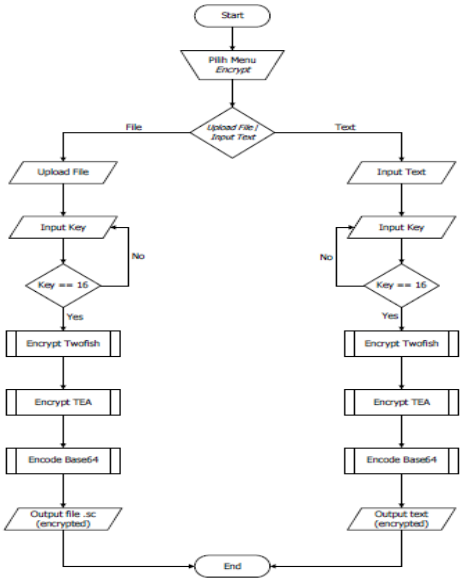



Figure 10: Flowchart process encrypt

### 3.4. Process to Decrypt

Unlike the process encrypt, decrypt process for have the checksum process dmana upon this process the contents of the file (plaintext) that was previously converted into the form of SHA1 will be checked. If the contents of the file to

be in same with SHA1 decrypt stored in the files, then decrypt process will be continued. If different then the process will not run and will appear a message error "the File Has Been Modified." His is like a flowchart in Figure 11 the following:

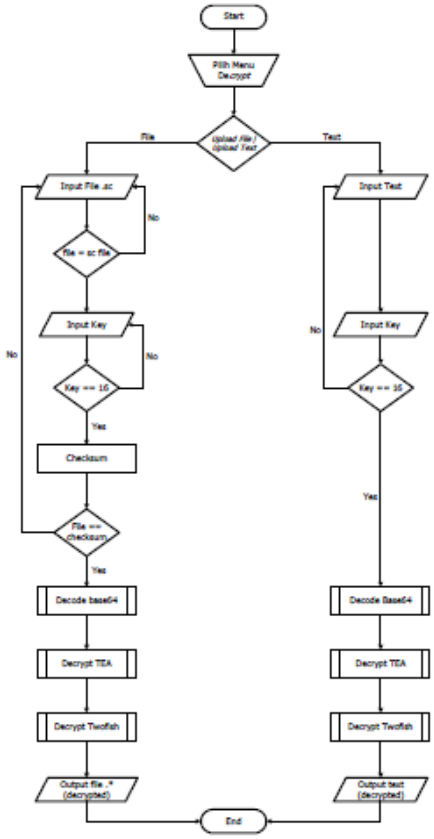



Figure 11: decrypt process Flowchart

## 4. Results And Discussion

In this discussion, conducted testing of applications have been built using xls files. The result of the test is done, the file is successfully carried out the process of securing data, so that the other party cannot see the contents of the real data. The xls file alan performed the process encrypt can be seen in Figure 12 below:

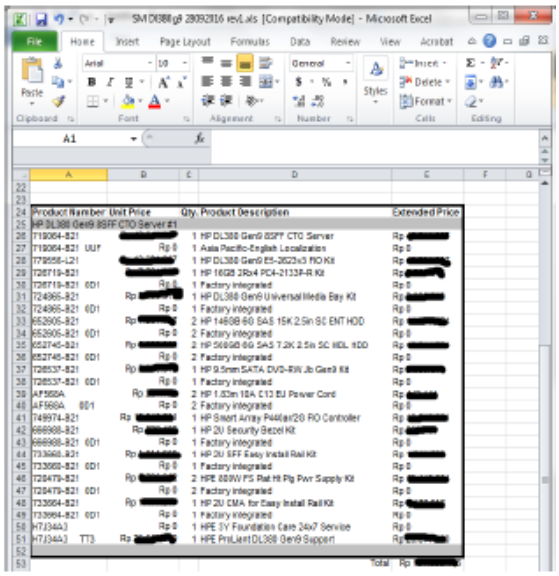


Figure 12: the display of the file before the process encrypt

File encrypt the result with extension. sc can be seen in Figure 14:

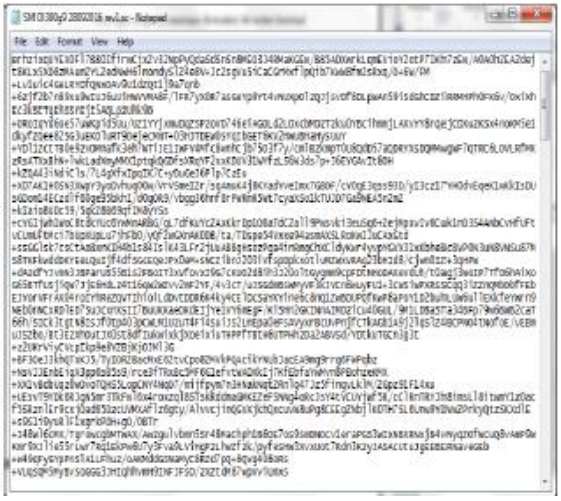
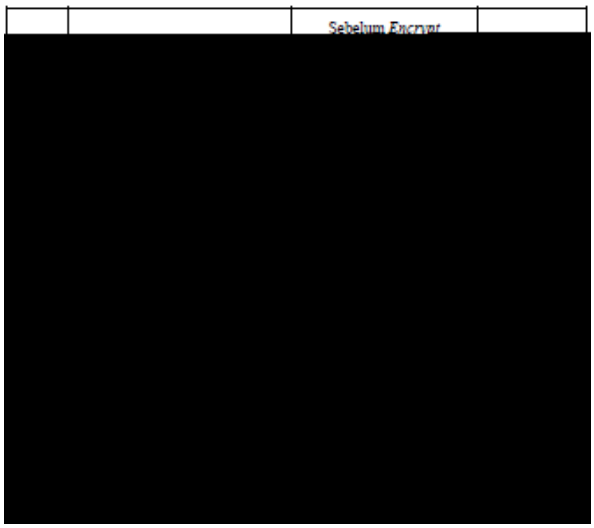


Figure 14: the display of the file after the process encrypt
Following are the results of the testing of multiple files with different formats are obtained as follows:

Table 3: table of results a test process to encrypt




From the results of a test, the file format will be changed to the format of SC. And for the size of the file after the process encrypt will turn into even bigger with an average size of the files that have been through the process encrypt increased approximately 33.3918 percent of the original size file before the process encrypt

Table 4: table of results a test process to decrypt
'

From the results of a test file format. thesc will be changed to the format of the file is performed before beginning the process, encrypt the file size after decrypt process will change to the initial size of the file before you encrypt. The average change in file size will increase of 25.0231 percent (reduced 33.3918 percent), the same as the average in the process encrypt.

Table 5: table comparisonprocessencrypt

Comparative trial results of speed encrypt, to speed the process encrypt different, the algorithm of the program the longest due to a combination of TEA and twofish algorithms, and to encrypt on the TEA for longer compared with at twofish because round of TEA used is 32, while 16 are round twofish

## 5. Conclusions And Suggestions

Based on the results of research and discussion as well as testing systems it can be concluded that the data can be secured well with Twofish cryptographic algorithms and the TEA, the data cannot be opened by the party not entitled to that do not have the key to decrypt file, the application also has a process to authenticate using SHA1 method check sum so only files that have been in the encrypt by applications that can decrypt. As for the input and suggestions in order to increase the performance of applications that have been built, as well as the improvement and development so that the program can encrypt and decrypt with less time for large file sizes, in order to the encrypt and decrypt on the several files at once in the same time for reasons of efficiency and also to be able to do data compression in encrypt so that data size will not change too much compared to the original size the data.

## BIBLIOGRAPHY

[1] Gehlot, Pumima, Biradar, s. r., Singh, b. p. 2013, Implementation of Modified Twofish Algorithm using 128 and 192-bit keys on VHDL. International Journal of Computer Applications (0975 – 8887) 70-Volume No. 13.

[2] v. Kiran Kumar, G, et. Al. 2015, Design And Implementation Of the Tiny Encryption Algorithm. International Journal of Engineering Research and Applications, ISSN: 2248-9622 vol. 5 Issue 6.
[3] in 2007, Indra, Twofish Algorithm: Implementation and Performance As one of the candidates the algorithm AES (Advanced Encryption Standard). Papers of students Bandung Institute of technology.
[4] Shoeb, Muhammad, Gupta, Vishal K 2013, A Crypt Analysis Of The Tiny Encryption Algorithm In Key Generation. International Journal of Computer Technologies (ISSN: 2278 – 9723) Volume 01-no. 38.
[5] IbnSaalih, Ahmad T, Gunadhi, Erwin, AsepSupriatna, 2013, securing PHP Programming Language script on Using Cryptography Base64. Journal of algorithms of high school Technology Arrowroot, ISSN: 2302-7339, vol. 10, no. 1.
[6] Rev. C, f. P &Rahangiar, a. & de Abreu, f. (2012). The application of Combined Algorithm RC4 and Base64 on E-Commerce security system, Journal of SatyaWacana Christian University.Yogyakarta.
[7] Bruce Schneier, 1996, Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in c. Indianapolis, Wiley Computer Publishing, ISBN 0471128457.
[8] Budiarsyah, financed with British K 2013, implementation and Speed Comparison of functions of several Popular Hash. Papers of students Bandung Institute of technology.
[9] Huda, Muharram 2009, development of the Encryption Hash function on the SHA (Secure Hash Algorithm). Papers of students Bandung Institute of technology.
[10] Chellaprabha.B, "Improving The Security Of Computer Networking Using Steganography Technique", International Journal of Innovations in Scientific and Engineering Research (IJISER), ISSN: 2347-971X (online), ISSN: 2347-9728(print), Vol.3.no.4, pp.33-37, 2016, http://www.ijiser.com/.
Rajesh, M., and J. M. Gnanasekar. "Annoyed Realm Outlook Taxonomy Using Twin Transfer Learning." International Journal of Pure and Applied Mathematics 116 (2017): 547-558.
Rajesh, M. & Gnanasekar, J.M. Wireless Pers Commun (2017),https://doi.org/10.1007/s11277-017-4565-9
Rajesh, M., and J. M. Gnanasekar. "GCCover Heterogeneous Wireless Adhoc Networks." Journal of Chemical and Pharmaceutical Sciences (2015): 195-200.
Rajesh, M., and J. M. Gnanasekar. "CONGESTION CONTROL IN HETEROGENEOUS WANET USING FRCC." Journal of Chemical and Pharmaceutical Sciences ISSN 974: 2115.
Rajesh, M., and J. M. Gnanasekar. "GCCover Heterogeneous Wireless Ad hoc Networks." Journal of Chemical and Pharmaceutical Sciences (2015): 195-200.
Rajesh, M., and J. M. Gnanasekar. "CONGESTION CONTROL USING AODV PROTOCOL SCHEME FOR WIRELESS AD-HOC NETWORK." Advances in Computer Science and Engineering 16.1/2 (2016): 19.
RAJESH, M. "TRADITIONAL COURSES INTO ONLINE MOVING STRATEGY." The Online Journal of Distance Education and e-Learning 4.4 (2016).
Rajesh, M. "Object-Oriented Programming and Parallelism."
Rajesh, M., K. Balasubramaniaswamy, and S. Aravindh. "MEBCK from Web using NLP Techniques." Computer Engineering and Intelligent Systems 6.8: 24-26.