International Journal of Security and Its Applications
Vol. 12, No. 3 (2018), pp.29-46
http://dx.doi.org/10.14257/ijsia.2018.12.3.04

# KAMIES: Security Optimization of KASUMI Algorithm by Increasing Diffusion Level

Muhammad Salman*, Rizki Yugitama, Amiruddin and Riri Fitri Sari

*Department of Electrical Engineering, Faculty of Engineering*
*Universitas Indonesia, Kampus Baru UI, Depok 16424, Indonesia*
*(muhammad.salman, rizki.yugitama, amiruddin43, riri@ui.ac.id)*

## Abstract

*In this paper we present KAMIES, a variation of KASUMI algorithm we implemented by applying the F function of MISTY1 and S-Box of AES. To compare the security level of KASUMI and KAMIES we have conducted some testing on two components i.e. F functions which consists of FI, FL and FO; and S-Boxes i.e.* $S_7$ *and* $S_9$ *of KASUMI and* $S_8$ *of AES. Methods used for testing the F function include Bit Independence Criterion (BIC) and Strict Avalanche Criterion (SAC), whereas methods for testing the S-Boxes include Avalanche Criterion (AC), SAC, BIC, XOR Table, Linear Approximation Table (LAT) and Nonlinearity. The results obtained from this study showed that the application of F Function of MISTY1 and S-Box of AES has an influence on the increase of the security level of KASUMI algorithm. This is based on the results of SAC and AWD testing on KASUMI and KAMIES algorithm. The result showed that KAMIES algorithm has a better diffusion level than KASUMI algorithm. Thus, KAMIES algorithm is more secure than KASUMI algorithm.*

***Keywords:*** *Avalanche Weight Distribution (AWD), Bit Independence Criterion (BIC), Block Cipher, KAMIES, KASUMI, MISTY1, Strict Avalanche Criterion (SAC)*

## 1. Introduction

Enhancing information and communication security [1] is always a challenge that accompanies technological advances. The most common technique used for such communication or data security is block or stream ciphers in cryptographic approach. An example of block cipher algorithm is KASUMI. KASUMI [2] is a block cipher encryption algorithm created by Secure Algorithms Groups of Experts (SAGE) [3] which is part of the European Telecommunication Standards Institute (ETSI). KASUMI algorithm forms the basis of A5/3 algorithm which is the security algorithm of Universal Mobile Telecommunication System (UMTS), an international standard for 3G mobile communication system. However, there have been several attacks ever directed to the KASUMI algorithm such as impossible [4], Boomerang and rectangle attacks [5]. KASUMI algorithm is based on MISTY1 algorithm [2].

MISTY1 [2] algorithm is a block cipher encryption algorithm which has high level of security, and fulfilled the specification from 3rd Generation Partnership Project (3GPP). The F function of MISTY1 has been tested by Akleylek [6] using LAT, XOR Table, and SAC test. The result showed a good performance. However, MISTY1 algorithm has experienced several attacks such as the impossible differential [7], slide, and integral attacks [8].

AES [9] is the standard encryption algorithm published by NIST on FIPS PUB 197 dated November 26, 2001. AES has a good confusion property, which is influenced by one of the most important components of the algorithm, S-Box. Based on Kavut and Yucel test [10],

Received (August 25, 2017), Review Result (November 18, 2017), Accepted (April 25, 2018)

[*] Corresponding Author

ISSN: 1738-9976 IJSIA

AES test results fulfilled the properties of AC with error value of 0,0352, SAC with error value of 0,125, BIC with error value of 0,1341 and maximum input value for XOR Table equals to 4 and minimum nonlinearity value of 112. This showed that the S-Box of AES has a possibility to be applied to KASUMI algorithm to increase the security level.

Based on the abovementioned description, it is necessary to increase the security level of KASUMI algorithm to avoid or at least to minimize such previously described attacks. Based on the good performance of the F Function of MISTY1 and S-Box of AES, in this research, we have modified KASUMI algorithm by applying the F Function of MISTY1 and S-Box of AES to increase its security level. Our modification algorithm of KASUMI is called, hereinafter, KAMIES, an acronym taken from KASUMI, MISTY1, and AES. For the ease of reading and understanding, we summarize the notations and symbols used in this papers as in Table 1.

**Table 1. Notations and Symbols used in the Paper**

| No. | Notations and Symbols | Meaning |
|---|---|---|
| 1 | = | Equals |
| 2 | ⊕ | XOR operation |
| 3 | \|\| | concatenation |
| 4 | <<<n | Left rotation of n bits |
| 5 | ROL() | Left rotation of 1 bits |
| 6 | ∩ | AND operation |
| 7 | ∪ | OR operation |
| 8 | $f_i$( ) | The round i-th function |
| 9 | *FI*() | Subfunctions on KASUMI and MISTY1 with 16-bit input and 16-bit outputs using 16-bit subkeys |
| 10 | *FL*() | Subfunctions on KASUMI and MISTY1 with 32-bit input and 32-bit output using 32-bit subkeys |
| 11 | *FO*() | Subfunctions on KASUMI and MISTY1 with 32-bit input and 32-bit output using 48-bit subkeys |
| 12 | *K* | Key with size 128-bit |
| 13 | $KL_i$, $KO_i$, $KI_i$ | Subkey used in round i-th |
| 14 | $S_7$[] | S-box that maps 7-bit inputs to 7-bit output |
| 15 | $S_8$[] | S-box that maps 8-bit inputs to 8-bit output |
| 16 | $S_9$[] | S-box that maps 9-bit inputs to 9-bit output |
| 17 | $\#\{x\}$ | The number of $x$ |

The rest part of this paper is structured as follows. Section 2 describes the cryptographic primitives and measurement. In Section 3 we described in detail the research methodology. Section 4 presents the test result and analysis, and we concluded the paper in Section 5.

## 2. Cryptographic Primitive and Measurement

In this section, we present a brief description of related cryptographic primitive and measurements which include block cipher, confusion and diffusion, hamming weight and distance, avalanche effect, Strict Avalanche Criterion (SAC), Nonlinearity, Bit Independence Criterion (BIC), Linear Approximation Table (LAT), Avalanche Weight Distribution (AWD), and the algorithm of KASUMI, MISTY1, and AES.

### 2.1. Block Cipher

Block cipher [11] [12] is a function that maps $n$-bit blocks of Plain text into n-bit blocks of Cipher text where $n$ is the block length. An $n$-bit block cipher is a function $E : V_n \times \mathcal{K} \rightarrow V_n$, with $K \in \mathcal{K}, E(P,K)$ is an invertible mapping (encryption function for K) from $V_n$ to $V_n$ denoted by $Ek\ (P)$ and $x$ is an operation. The inverse mapping is a decryption function denoted by $Dk\ (C)$, where $C\ =\ Ek\ (P)$ is the Cipher text $(C)$ resulted from the encryption of the Plain text $P$ and the key $K$.

### 2.2. Confusion and Diffusion

According to Shannon [13], two basic techniques used to obscure redundancy in Plain text are confusion and diffusion. Confusion is obscuring the relationship between Plain text and Cipher text so that the characteristics or patterns in the Plain text are not found in the Cipher text. A simple way to get the confusion property is by substitution, which is to replace a Plain text symbol with another symbol to form the Cipher text. Diffusion is to remove the characteristics or patterns of Plain text by spreading the patterns across the Cipher text. The simplest way to get the diffusion property is by transposition, which is to change the position of Plain text elements in such a way as to produce a Cipher text.

### 2.3. Hamming Weight and Distance

Hamming Weight [14] is the number of non-zero bits (bit 1) contained in a word. For example, Hamming Weight from a word 11110 is 4 since the number of bit 1 is 4, whereas Hamming Weight of a word 00001 is 1. Hamming Distance [14] $d(z_i, z_j)$ between two word $z_i$ and $z_j$ with the same length is the number of different symbols between $z$ and $z'$. The Hamming Distance of two codewords is the weight of bits difference. An example of calculating the Hamming Distance of a binary word with a five-bit length is given as follows: 11110 $\oplus$ 00001= 11111. Since the result obtained is five bit 1, then the Hamming Distance of the two codewords is 5, which means there are five bits difference between the two codewords.

### 2.4. Substitution Box (S-Box)

The term S-Box $n\ \times\ n$ is a function that maps $n$-bit inputs into $n$-bit outputs [15] [16]. Definitively, S-Box is a mapping function $f: \{0,1\}^n\ \rightarrow \{0,1\}^n$, which maps $n$-bit inputs, $X = \{x_1, x_2, \dots, x_n\}$, into $n$-bit output, $Y = \{y_1, y_2, \dots, y_n\}$, in this case $Y = f(x)$, as shown in Figure 1.

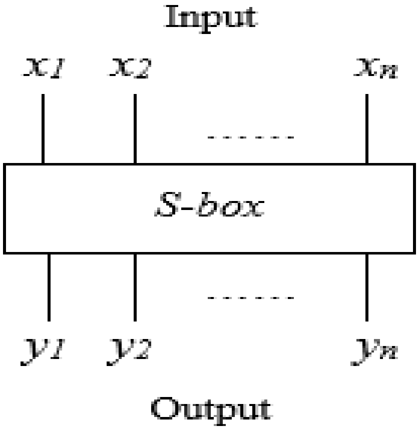

**Figure 1. S-Box Scheme**

### 2.5. Avalanche Effect

The concept of the avalanche [6, 15] was first discovered by Horst Feistel in 1973. He argued that, a function $f\colon Z_2^n \to Z_2^m$ is said to satisfy the avalanche criterion. If one input bit is supplemented, then a half of the total number of output bits will be changed, as in (1).

$$\sum_{x \in Z_2^n} \omega t(f(x) \oplus f(x \oplus c_i^n)) = m2^{n-1} \tag{1}$$

for each i $(1 \le i \le n)$ where $Z_2^n$ is $n$ – dimensional vector space over the finite field, $\oplus$ is XOR operator, $\omega t(.)$ is Hamming Weight function, $c_i^n$ is $n$ - dimensional vector with Hamming Weight = 1 at position $i$-th, $f\ (x)$ is a function $f\colon Z_2^n \to Z_2^m$, and $m$ is the number of bits of result F Function (x) . This means that a half of the output bits change if one input bit is complemented. Based on Eq. (1), we obtained the new formulation of the parameter avalanche, $K_{AVAL}$, as in (2).

$$K_{AVAL}(i) = \frac{1}{m2^n} \sum_{x \epsilon Z_2^n} \omega t\big(f(x) \oplus f(x \oplus c_i^n)\big) = \frac{1}{2} \tag{2}$$

Thus, if the $i$-th $K_{AVAL}$ is less than or more than a half for each $i$, the avalanche criterion is not satisfied. According to [10], an S-Box satisfies the Avalanche Criterion (AC) with relative error of $\pm \varepsilon_A$ if for all $i$ satisfy Eq. (3). The relative error value of $\in_A$ can be calculated using Eq. (4).

$$\frac{1}{2}\ (1 - \in_A) \le K_{AVAL}(i) \le \frac{1}{2}\ (1 - \in_A) \tag{3}$$

$$\in_A = max_{1 \le i \le n} |\ 2K_{AVAL}\ (i) - 1| \tag{4}$$

### 2.6. Strict Avalanche Criterion

Webster and Tavares [17] argued that, a F Function : $Z_2^n \to Z_2^m$ is said to satisfy SAC if each input bit is complemented, it will result in half of the total number of output bits changing. In other words, a F Function : $Z_2^n \to Z_2^m$ is said to meet the SAC criteria if for every i $(1 \le i \le n)$ satisfies Eq. (5). Based on Eq.(1), we obtained new formulation of SAC parameter, $K_{SAC}$ as in (6).

$$\sum_{x \epsilon Z_2^n} f(x) \oplus f(x \oplus c_i^n) = (2^{n-1}, 2^{n-1}, \dots, 2^{n-1}) \tag{5}$$

$$K_{SAC}(i,j) = \frac{1}{2} \omega t\big(f(x) \oplus f(x \oplus c_i^n)\big) = \frac{1}{2} \tag{6}$$

Thus the $K_{SAC}$ can be worth between zero and one, but only $K_{SAC}$ is worth half for each $i$, capable of meeting the SAC criteria. In addition, it can be said that if a function has met the criteria SAC then indirectly the function has met the criteria of avalanche effect and completeness. S-Boxes meet SAC with relative error ± $\in_s$ if for all $i$, it satisfies Eq. (7) and the relative error value $\in_s$ can be calculated by using Eq. (8).

$$\frac{1}{2}\ (1 - \in_s) \le K_{SAC}(i,j) \le \frac{1}{2}\ (1 - \in_s) \tag{7}$$

$$\in_s = max_{i,j} |\ 2K_{SAC}\ (i,j) - 1| \tag{8}$$

### 2.7. Nonlinearity

A defined function $f\colon Z_2^n \to Z_2$ is said to be a linear function if for every $x \in Z_2$, $f(x) = a\,.(x)$ with constant $a\ \in\ Z_2^n$ applies. A function $f$ is said to be an affine function if $f(x) = a.x\ \oplus b$ for constant $a\ \in Z_2^n, b\ \in Z_2$. The linear structure of a Boolean function

$f: Z_2^n \rightarrow Z_2$ can be identified with a vector a ∈Z_2^n/{0} such that $f(x \oplus a) \oplus f(x)$ has the same value (0 or 1) for all $x \in Z_2^n$.

According to Youssef [18], a non-linear function, $\mathcal{NL}_f$ of $f = (f_1 f_2 \dots f_n): Z_2^n \rightarrow Z_2^m$ where $f: Z_2^n \rightarrow Z_2$for every i = 1, 2, ..., m is defined as the smallest Hamming Distance between the set of affine functions with any non-zero linear combinations of its $f$-output coordinates, and formulated as in (9).

$$\mathcal{NL}_f = \min_{b,c,w} \#\{x \in Z_2^n | c.f(x) \neq w.x \oplus b\} \tag{9}$$

where $w \in Z_2^n$, c $\in Z\_2\hat{}m\backslash\{0\}$, $b \in Z_2$, $x \in Z_2$ and $w.x$ are multiplication of points between $w$ with $x$ in $Z_2$, and

$$c.f(x) = \overset{m}{\underset{i=1}{\oplus}} c_i f_i(x), \tag{10}$$

where $c = \{c_1, c_2, \dots, c_m\} \in Z_2^m$.

A cryptographic system that is not susceptible to linear cryptanalysis requires the value of minimum nonlinearity function, $NLM_f$ which approaches the maximum nonlinearity (perfect nonlinearity function). The maximum nonlinearity of a Boolean function is to satisfy or approximate the equation $N_f \leq 2^{n-1} - 2^{\frac{n}{2}-1}$. The minimum value of $NLM_f$ approaching 0 indicates that the function $f$ approximates the affine function and is susceptible to linear cryptanalysis [19].

### 2.8. Bit Independence Criterion (BIC)

BIC was introduced by Webster and Tavares [17]. A F Function: $\{0,1\}^n\{0,1\}^n$ satisfies the BIC if for all $i, j, k \in \{1,2,\dots,n\}, j \neq k$, with changing input bit i resulted in the output of bits j^th^ and k^th^ change independently. To measure the BIC properties, a correlation coefficient between j and k components of the difference string output, called the avalanche vector $a_i^{e_i}$ is needed. A BIC parameter corresponds to the effect of the i^th^ bit changes of the input bits to the j^th^ and kth bits of the avalanche vector $a_i^{e_i}$ and can be formulated as in (11). Overall, the BIC parameter for F Function or S-Box is defined as in (12). The value of BIC is at interval (0,1). If the value is 0 then the avalanche variable is always identical or complement, while if the value is 1 then the avalanche variable is always independent.

$$BIC(a_J, a_k) = \max_{1 \leq i \leq n} |corr\left(a_j^{e_i}, a_k^{e_i}\right)| \tag{11}$$

$$BIC(f) = \max_{\substack{1 \leq i \leq n \\ j = k}} BIC\left(a_j, a_k\right) \tag{12}$$

### 2.9. XOR Table

For an n x n S-Box, the XOR-Table of the s-Box is a matrix that has rows and columns indexed with $0,1,2,\dots,2^n - 1$, and inputs in the table are indexed with (δ, b) , where δ denotes the number of input vectors P modified by δ, and b represents the change of output, where $b = f(P) \oplus f(P \oplus \delta)$. XOR-Table formula is given in (13), where $\delta \in Z_2^n$ and $b \in Z_2^m$.

$$XOR\ f(\delta, b) = \#\{P | f(P) \oplus f(P \oplus \delta) = b\} \tag{13}$$

The number of inputs in the XOR table is always even and the sum of all the values in the line is always $2^n$ [10]. Ideally, the values in the XOR table are zero or two with the exception of inputs (0,0) that are always valued with $2^n$. Due to the number of inputs that is always $2^n$, the ideal input composition is 50% of zero and 50% is two. The high input values in XOR-Table can be used to perform differential cryptanalysis, so the precise condition of the s-Box for resistance to differential cryptanalysis is to avoid high input values.

## 2.10. Linear Approximation Table (LAT)

To test the resistance of an S-Box to linear cryptanalysis we can use LAT-distribution [20] of the S-Box or function $f(x)$: $\{0,1\}^n \rightarrow \{0,1\}$. This can be done by creating a linear function of an S-Box, where each output or any linear combination of output can be formed with linear function. LAT Distribution of an S-Box function is defined as the sum of all input variations $X \in Z_2^n$ which causes the XOR value of input bits operated with α equal to the XOR value of output bits operated with β. Luke O 'Connor in [20] explained the theory of LAT that if there is an S-Box function $\pi : Z_2^n \rightarrow Z_2^n$ which is bijective with the $n$-bit mapping, and if $S_{2^n}$ is the whole set of mappings called symmetric groups. For n-bit vector $\in Z_2^n$, $Xi$ is a notation of the $i^{\text{th}}$ bit of $X$. Thus, the LAT table for the function π, with the notation $LAT_\pi$ is a table of $2^n x 2^n$ such that it applies Eq. (14), where $\alpha \in \{0,1\}$ for $0 \leq \alpha \leq 2^n - 1$, $\beta \in \{0,1\}$ for $0 \leq \beta \leq 2^n - 1$, $X$ is the input value before being substituted to the S-Box $f : \{0,1\}^n \rightarrow \{0,1\}^n$, $\pi(X)$ is the input value after substituted to S-Box $f: \{0,1\}^n \rightarrow \{0,1\}^n$, $\#\{x\}$ = The number $x$. $LAT_\pi(\alpha,\beta)$= The LAT value gives the corresponding parity check number between the linear combination of the input bits (α) and the linear combination of the output bits (β),

$$LAT_\pi(\alpha,\beta) = \#\{X | X \in Z_2^n, \bigoplus_{i=1}^{n} X[i]\,.\,\alpha[i] \; = \; \bigoplus_{i=1}^{n} \pi(X)[i].\beta\,[i] \qquad (14)$$

The LAT criterion testing parameter is based on the result of LAT value in [20] where:

$$LAT_\pi^*(\alpha,\beta) = |\, LAT_\pi(\alpha,\beta) - 2^{n-1}| \qquad (15)$$

The further the LAT value from the ideal LAT value of 128, the more susceptible the S-Box is to linear cryptanalysis. Based on the LAT table, the linear approximation probability is calculated. A probability of less than or more than half can be said to have a correlation between the input and the output. Therefore, it will be easy to be analyzed. Based on this, it can be concluded that the complexity of linear cryptanalysis depends on the input values in the LAT table [21].

## 2.11. Avalanche Weight Distribution (AWD)

The criterion that must be met in the AWD test measured the diffusion properties of the block cipher. The Hamming Weight histogram and its avalanche vector must be random when the Plain text pair $(P_1, P_2)$ are almost identical. Therefore the AWD curve corresponding to all possible almost identical pairs should be binomially distributed near $n/2$. N is the block size of the tested algorithm for block ciphers having the good diffusion properties with block length of $n$ [10]. The probability of finding the number of bits of Cipher text $i$ changes in an $n$-bit Cipher text was calculated using Eq. (16) and (17).

$$B(j) = \frac{\binom{n}{j}}{2^n}, 0 \leq j \leq n \qquad (16)$$

$$\sum_{\mathrm{j=0}}^{\mathrm{n}} \mathrm{B(j)} = 1 \qquad (17)$$

The distortion $(D)$ is a deviation between the AWD test results on the algorithm with the ideal distribution B(j) calculated using $N$ pairs of bright text $(P, P \oplus \Delta P)$ with a fixed $\Delta P$ difference, and corresponds to the Cipher text $(C, C \oplus \Delta C)$. $\Delta C$ from weight $j$ is added to 1 on the array element $AWD(j)$. The distortion between the algorithm and the ideal Binomial distribution is obtained using Eq. (18).

$$D^i = \frac{1}{2N} \sum_{j=0}^{n} |AWD(j) - NB(j)| \qquad (18)$$

where $N$ is the number of samples used, $n$ is the block size of the algorithm being tested, $j$ denotes the Hamming Weight, for $0 \leq j \leq n$. In accordance with the resemblance parameter $R$ to the Binomial distribution, Eq. (19) is given as:

$$R^i = 1 - D^i \qquad (19)$$

If the value of $R^i = 1$ then the AWD of the block cipher algorithm is exactly the same as the ideal Binomial distribution, whereas if the value of $R^i = 0$, then the AWD of the block cipher algorithm does not show any resemblance to the ideal Binomial distribution.

## 2.12. KASUMI

KASUMI [2] is a type of block cipher encryption algorithm and a variance of MISTY1 which was developed by Security Algorithms Group of Experts (SAGE) as the basic A5/3 algorithm used for GSM-based communication encryption algorithms [3]. The structure of the KASUMI algorithm can be seen in Figure 2.

## 2.13. MISTY1

The MISTY1 [2] is a block cipher symmetric algorithm with 64-bit data input and using a 128-bit key. The structure of the MISTY1 algorithm can be seen in Figure 3.

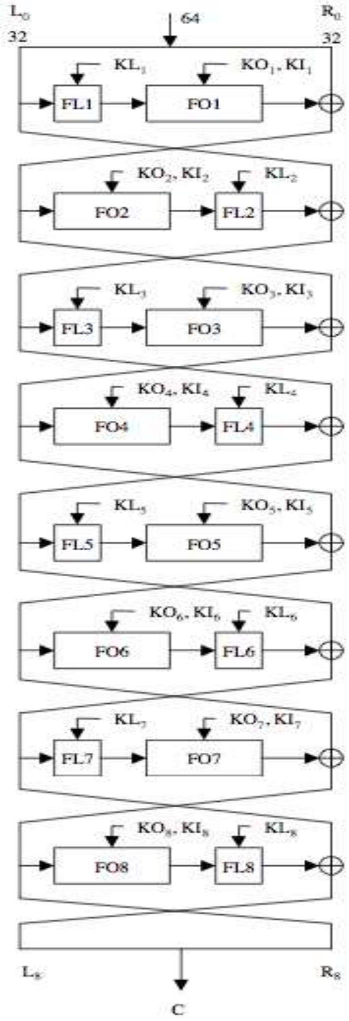

Figure 2. Structure of KASUMI Algorithm [3]

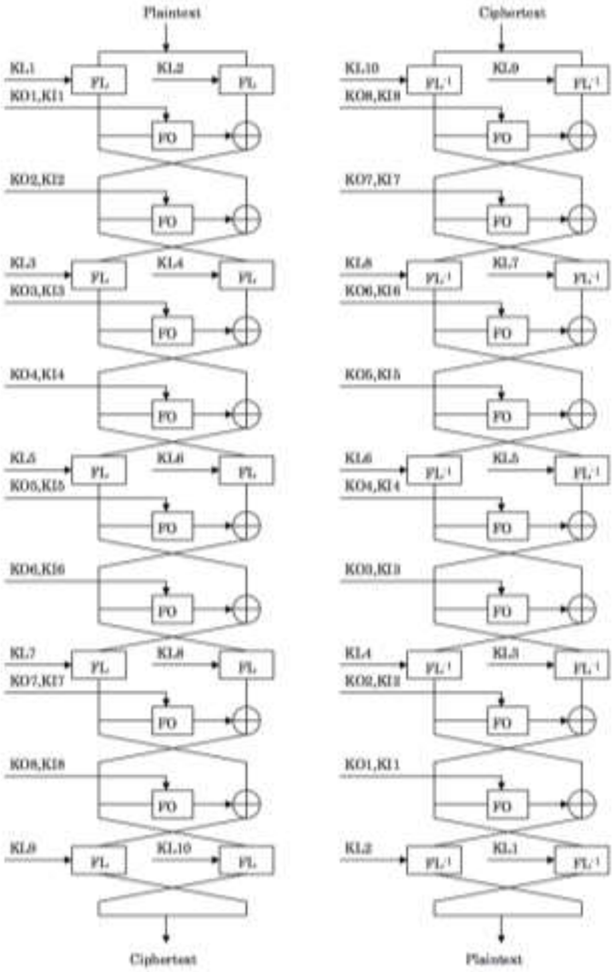

Figure 3. Structure of MISTY1 Algorithm [22]

### 2.14. Advance Encryption Standard (AES)

The Rijndael block cipher algorithm proposed by Joan Daemen and Vincent Rijmen was selected as the Advanced Encryption Standard (AES) in 2001 [9]. This algorithm is a standard block cipher-based symmetric algorithm that encrypts 128-bit input blocks into 128-bit output blocks. The AES algorithm uses various key lengths, *i.e.*, 128, 192, and 256 bits. Based on the key length, AES is grouped into AES-128, AES-192, and AES-256.

## 3. Research Methodology

### 3.1. Data Collection

In this research, we used literature study and experimental method. Literature study reviewed related books, papers, and other sources that can support this research. The experimental method is done by testing the S-Box of both KASUMI and AES algorithm by controlling the variable so that the influence of outside variable can be eliminated. Tests conducted on KASUMI and KAMIES algorithms are AWD and SAC tests. Tests performed on the F function of KASUMI and MISTY1 are SAC and BIC tests, whereas tests on the S-Box of KASUMI and AES are AC, SAC, BIC, XOR Table, LAT and Nonlinearity tests. We used 30.000 Plain text samples where each Plain text is 64 bit long. When the Plain text on the KASUMI and KAMIES algorithm is treated as an independent variable, the key as a controlled variable is made constant with a value of zero. The use of constant value of zero on controlled variable is to eliminate the influence of the controlled variable, since we will test the influence of the change of the independent variable to the dependent variable. The independent variables are taken randomly by using simple random sampling technique using random function in Matlab.

### 3.2. Research Stages

The steps undertaken in this research are:

1. Study the literature of the KASUMI, MISTY1 and AES; the diffusion concept of the KASUMI; the concept of testing AC, SAC, BIC, XOR Table, LAT, and Nonlinearity properties on S-Box; the concept of testing SAC and BIC properties on F function; and the concept of AWD and SAC testing on the block cipher algorithm.
2. Implement F Function of MISTY1 and S-Box of AES on the KASUMI algorithm.
3. Test $S_7$ and $S_9$ of KASUMI algorithm and $S_8$ of AES algorithm with AC, SAC, BIC, XOR-Table, LAT-Table, and Nonlinearity test.
4. Test the KASUMI and KAMIES algorithms using AWD and SAC test.
5. Analyze the results of AC, SAC, BIC, XOR-Table, LAT-Table, and Nonlinearity tests as well as AWD and SAC tests, and compare the AWD and SAC test results between the KASUMI and KAMIES algorithms.
6. Draw conclusions related to simulation of the proposed method on KASUMI.

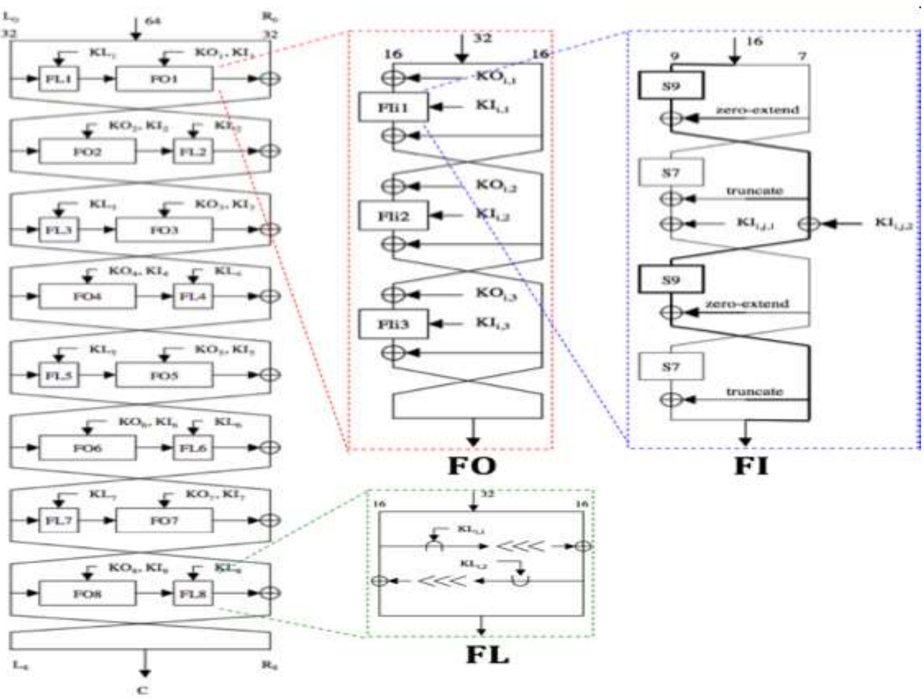

Figure 4. Structure of Original KASUMI Algorithm [3]

### 3.3. Structure of KAMIES Algorithm

Our proposed KAMIES algorithm is a modification of the KASUMI algorithm. The modification lies in the replacement of the F functions (FO, FI, FL) and the S-Box ($S_9$ and $S_7$) of KASUMI with the F functions (FO, FI, FL) of MISTY1 and the S-Box ($S_8$) of AES. Detail comparison of structural difference KASUMI and KAMIES can be seen in Figure 4 and Figure 5.

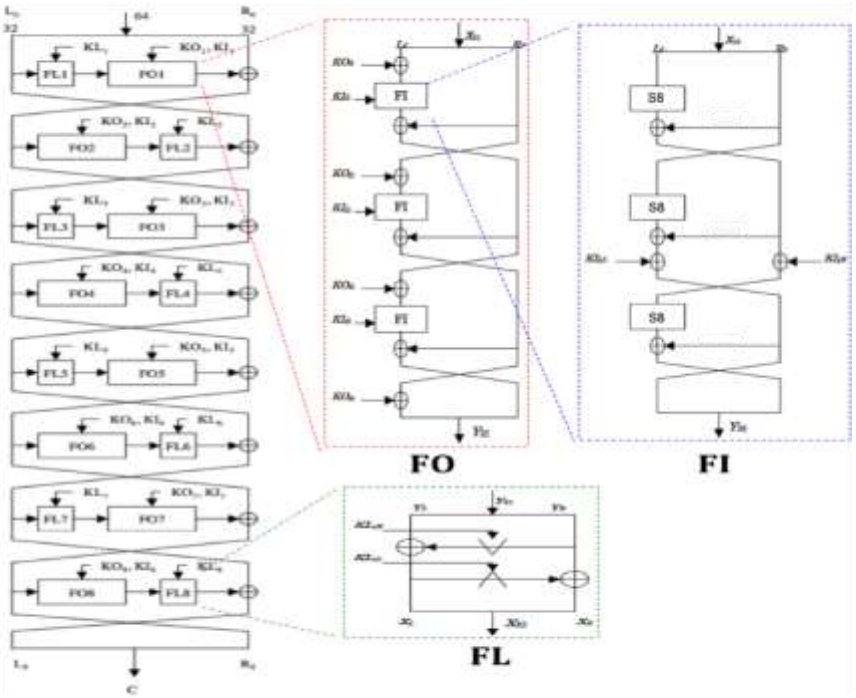

Figure 5. Structure of KAMIES algorithm (modified KASUMI algorithm)

### 3.4. Research Variables

The research variables used in this research are summarized in Table 2.

**Table 2. Research Variables**

| Test Type | Components to be Tested | Variables | | |
|---|---|---|---|---|
| | | Input | | Output |
| | | Independent | Controlled | Dependent |
| SAC | $S_7$, $S_9$, $S_8$ | Input $S_7$, $S_8$ and $S_9$ | - | Output $S_7$, $S_8$ and $S_9$ |
| | F function | Input *F* function | Subkey | Output *F* function |
| | | Subkey | Input *F* function | Input *F* function |
| AWD | KASUMI | Plain text | Key | Cipher text |
| | KAMIES. | Plain text | Key | Cipher text |
| BIC | $S_7$, $S_9$, $S_8$ | Input $S_7$, $S_8$ and $S_9$ | - | Output $S_7$, $S_8$ and $S_9$ |
| | F function | Input *F* function | Subkey | Output *F* function |
| | | Subkey | Output *F* function | Output *F* function |

### 3.5. Population and Sample

We tested the KASUMI and KAMIES algorithm when the plain text is used as the independent variable. We used 30.000 plain text sample of the total $2^{64}$ Plain text population as in Table 3. The number of samples used in the F function tests (FL, FO and FI) is based on the number of samples used in the test conducted by [12] to the AES algorithm with a sample size of $2^{12}$ as in Table 4.

**Table 3. Population and sample of AWD Test**

| No. | Algorithm | Independent Variable | Population (N) | Sample (n) |
|---|---|---|---|---|
| 1 | KASUMI | Plain Text | $2^{64}$ | 30.000 |
| 2 | KAMIES | Plain Text | $2^{64}$ | 30.000 |

**Table 4. Population and Sample of SAC and BIC Test on F Function**

| No. | F Function | Independent Variable | Population (N) | Sample (n) |
|---|---|---|---|---|
| 1 | FL | Input FL | $2^{32}$ | $2^{12}$ |
| | | Subkey | $2^{32}$ | $2^{12}$ |
| 2 | FO | Input FO | $2^{32}$ | $2^{12}$ |
| | | Subkey | $2^{48}$ | $2^{12}$ |
| 3 | FI | Input FI | $2^{16}$ | $2^{12}$ |
| | | Subkey | $2^{16}$ | $2^{12}$ |

### 3.6. Data Processing and Analysis

For data processing and analysis, we used C ++ programming language for algorithm and Matlab for testing the S-Box and generating sample for Plain text using a laptop with Core i5 2.5 GHz and memory or RAM of 16 GB for testing AC, SAC, BIC, XOR-Table, LAT-Table, Nonlinearity and AWD. The results of the data processing stage will then be used in the data analysis phase. Data analysis is aimed at determining the diffusion level of the S-Boxes and algorithms of KASUMI and KAMIES. Data is used to test AC, SAC, BIC, XOR-Table, LAT-Table, Nonlinearity and AWD properties. The data analysis process is done by observing the percentage matrix of the bit frequency distribution, the correlation of coefficient matrix, and the AWD test matrix for each independent variable used. The S-Box and both KASUMI and KAMIES algorithms are said to meet the SAC criteria if each input in the frequency distribution is 50% with a relative error of 4%, so the value considered to meet the SAC is 48% - 52%. Toz in [12] tested the SAC against the AES algorithm and concluded that AES satisfied the SAC criteria with a relative error of 3.2%. This study is expected to use a relative error of 4% which is close to the ideal of the SAC criteria.

## 4. Result and Discussion

The tests on F functions and S-Box of KASUMI and MISTY1 algorithms are based on the basic theories described in Section 2. The SAC test results are represented by using the bit frequency distribution matrix. It is said to pass the SAC test if the test result shows the interval of SAC value is between 48% - 52%. In other word, the maximum relative error allowed is 4% of the ideal SAC value which is 50%. The BIC test result is represented by using bit correlation coefficient. It is said to pass the BIC test if the test result has a maximum BIC value of 0.02 from the ideal value of 0. The R values for each AWD test result was evaluated. It is said to pass the AWD test if the R value of each AWD test result has a maximum error value of 2% or the minimum R value is 0.98.

### 4.1. S-Box Testing

As summarized in Table 5, the maximum value of the relative error on $S_7$ of KASUMI, and $S_7$ of MISTY1 is 0.01786. The avalanche interval is $0{,}49107 \leq k_{AVAL} \leq\ 0{,}508$. In $S_9$ of KASUMI and $S_9$ of MISTY1, the maximum value of relative error is 0.11112. The avalanche interval produced is $0{,}44444 \leq k_{AVAL} \leq\ 0{,}55556$. The maximum value of the relative error on $S_8$ of AES is 0.035156, so that the avalanche interval is $0{,}482422 \leq k_{AVAL}(i,j) \leq 0{,}517578$. Therefore, it can be concluded that $S_7$ of KASUMI, $S_7$ of MISTY1 and $S_8$ of AES all passed the AC test with avalanche interval value of 48% - 52%. It can be said they satisfied the relative error of 4%. However, $S_8$ of AES has a better AC value than $S_7$ of MISTY1 and $S_8$ of AES.

**Table 5. AC Relative Error Value of S-Boxes**

| Bit *i* | AC Relative Error Value | | | | |
|---|---|---|---|---|---|
| | *$S_7$* of KASUMI | *$S_7$* of MISTY1 | *$S_9$* of KASUMI | *$S_9$* of MISTY1 | *$S_8$* of AES |
| 1 | 0,01786 | 0,01786 | 0,11112 | 0,11112 | 0,01563 |
| 2 | 0,01786 | 0,01786 | 0,11112 | 0,11112 | 0,00391 |
| 3 | 0,01786 | 0,01786 | 0,11112 | 0,11112 | 0,023438 |
| 4 | 0,01786 | 0,01786 | 0,11112 | 0,11112 | 0,015625 |
| 5 | 0,01786 | 0,01786 | 0,11112 | 0,11112 | 0 |
| 6 | 0,01786 | 0,01786 | 0,11112 | 0,11112 | 0,007813 |
| 7 | 0,01786 | 0,01786 | 0,11112 | 0 | 0,015625 |
| 8 | | | 0,11112 | 0,11112 | 0,035156 |
| 9 | | | 0,11112 | 0,11112 | |

**Table 6. SAC Relative Error Value of S-Box**

| No. | S-Box | Value |
|---|---|---|
| 1. | $S_7$ OF KASUMI | 0,125 |
| 2. | $S_7$ MISTY1 | 0,125 |
| 3. | $S_9$ KASUMI | 1 |
| 4. | $S_9$ MISTY1 | 1 |
| 5. | $S_8$ OF AES | 0,125 |

Based on Table 6, the relative error value of $S_7$ of KASUMI, $S_7$ of MISTY1 and $S_8$ of AES of 0.125, thus, the avalanche interval produced is $0{,}4375 \leq k_{SAC} \leq\ 0{,}5625$. In $S_9$ of KASUMI, and $S_9$ of MISTY1, the relative error value is 1, so the avalanche interval produced is $0 \leq k_{SAC} \leq\ 1$. It can be concluded that $S_7$ of KASUMI, $S_7$ of MISTY1, $S_9$ of KASUMI, $S_9$ of MISTY1 and $S_8$ of AES did not pass the SAC test with avalanche value interval obtained is out of the 48% - 52% interval.

Based on Table 7, it is found that $S_7$ of KASUMI and MISTY1 have the same correlation value of 0.12599. $S_9$ of KASUMI and MISTY1 have the same correlation value of ∞, and this occurred due to the existence of a correlation value on $S_9$ of KASUMI and MISTY1 due to devided by zero. Meanwhile, $S_8$ of AES has a correlation value of 0.13412. From the analysis of BIC criteria on the entire $S_7$ of KASUMI, $S_7$ of MISTY1, $S_9$ of KASUMI, $S_9$ of MISTY1 and $S_8$ of AES, it can be concluded that with the maximum correlation is close to 0 then the value of avalanche variable has the mutual independent character between the output bits.

Based on Table 8, maximum value of XOR test on $S_7$ of KASUMI and $S_7$ of MISTY1 is 8128, which means that there are 8128 pairs of input and output differences which produce a certain maximum output difference of 2 out of 128 possibilities. In $S_9$ of KASUMI and $S_9$ of MISTY1 the maximum input value generated is 130816 which indicates that there are 2 values of a certain difference out of 512 possible output differences, whereas the maximum input value generated on $S_8$ of AES is 32130 which indicates there are 1 values of a certain difference out of 256 possible output differences. The lower the number of inputs, the easier it is to obtain a differential equation. Thus, it can be concluded that differential cryptanalysis is difficult to apply to $S_7$ of KASUMI, $S_7$ of MISTY1, $S_9$ of KASUMI, $S_9$ of MISTY1 and $S_8$ of AES.

**Table 7. Maximum Correlation Value of BIC S-Box Test**

| No. | S-Box | Maximum Correlation Value |
|---|---|---|
| 1 | $S_7$ OF KASUMI | 0,12599 |
| 2 | $S_7$ MISTY1 | 0,12599 |
| 3 | $S_9$ KASUMI | ∞ |
| 4 | $S_9$ MISTY1 | ∞ |
| 5 | $S_8$ OF AES | 0.13412 |

**Table 8. XOR Table Test Results**

| S-Box | Number of Inputs | | | | | |
|---|---|---|---|---|---|---|
| | 0 | 2 | 4 | 128 | 256 | 512 |
| $S_7$ OF KASUMI | 8255 | 8128 | | 1 | | |
| $S_7$ MISTY1 | 8255 | 8128 | | 1 | | |
| $S_9$ KASUMI | 131327 | 130816 | | | | 1 |
| $S_9$ MISTY1 | 131327 | 130816 | | | | 1 |
| $S_8$ OF AES | 33150 | 32130 | 255 | | 1 | |

**Table 9. Extreme Bias Value Based on LAT Test**

| S-Box | Extreme Bias Value |
|---|---|
| *S₇* OF KASUMI | $\pm 8/128$ |
| *S₇* MISTY1 | $\pm 8/128$ |
| *S₉* KASUMI | $\pm 16/512$ |
| *S₉* MISTY1 | $\pm 16/512$ |
| *S₈* OF AES | $\pm 16/256$ |

As summarized in Table 9, the extreme bias values of $S_7$ of KASUMI and $S_7$ of MISTY1 ranges from $-\frac{8}{128}$ to $\frac{8}{128}$. On $S_9$ of KASUMI and $S_9$ of MISTY1 the bias ranges from $-\frac{16}{512}$ to $\frac{16}{512}$. The extreme bias value of $S_8$ OF AES ranges from $-\frac{16}{256}$ to $\frac{16}{256}$. It can be concluded that $S_7$ of KASUMI, $S_7$ of MISTY1, $S_9$ of KASUMI, $S_9$ of MISTY1 and $S_8$ of AES have the LAT values approaching 0, thus, they are all resistant to linear cryptanalysis attacks. Based on the results on Table 10, it can be concluded that the minimum nonlinearity value on $S_7$ of KASUMI and $S_7$ of MISTY1 produced is 56. On $S_9$ of KASUMI and $S_9$ of MISTY1 the minimum nonlinearity value generated is 240, while on $S_8$ of AES the minimum nonlinearity value generated is 112. The minimum $\mathcal{NL}_f$ value is relatively close to the perfect nonlinearity value of $2^{n-1} - 2^{\frac{n}{2}-1}$ in $S_7$ of KASUMI and $S_7$ of MISTY1 is 58.3431, on $S_9$ of KASUMI and $S_9$ of MISTY1 is 244,6862, $S_8$ of AES is 120.

**Table 10. Nonlinearity Minimum (NLM) Value**

| S-Box | NLM | Probability |
|---|---|---|
| *S₇* OF KASUMI | 56 | 72/128 |
| *S₇* MISTY1 | 56 | 72/128 |
| *S₉* KASUMI | 240 | 272/512 |
| *S₉* MISTY1 | 240 | 272/512 |
| *S₈* OF AES | 112 | 144/256 |

In relation to the resulting $\mathcal{NL}_f$ value which is close enough to the ideal value, the number of Plain text satisfying the equation $c.f(x) = w.x \oplus b$ on $S_7$ of KASUMI and $S_7$ of MISTY1 is 72, in $S_9$ of KASUMI and $S_9$ of MISTY1 is 272, while the $S_8$ of AES value is 144. Thus, the probabilities generated in $S_7$ of KASUMI and $S_7$ of MISTY1 is $\frac{72}{128}$, on $S_9$ of KASUMI and $S_9$ of MISTY1 is $\frac{272}{512}$, and on $S_8$ of AES is $\frac{144}{256}$ which is close to $\frac{1}{2}$. Based on the minimum nonlinearity value, the number of vectors, and the probability generated, it can be concluded that $S_7$ of KASUMI, $S_7$ of MISTY1, $S_9$ of KASUMI, $S_9$ of MISTY1 and $S_8$ of AES meet the nonlinearity test, and therefore they are all resistant to linear cryptanalysis.

### 4.2. SAC Testing Algorithm KASUMI and KAMIES

Table 11 shows the results of SAC test on KASUMI and KAMIES when Plain text is treated as independent variable. It indicates that the entire KASUMI and KAMIES algorithm passed the SAC test. The biggest error value is 0.023133333 which was calculated using Eq. (8) and (7), $\in = max_{i,j}|2k_{SAC}(i,j) - 1|$. SAC value interval = $\frac{1}{2}(1-\in) \leq k_{SAC}(i,j) \leq \frac{1}{2}(1+\in)$ = $\frac{1}{2}(1 - 0{,}02313333) \leq k_{SAC}(i,j) \leq \frac{1}{2}(1 + 0{,}02313333) = 0{,}4884333335 \leq k_{SAC}(i,j) \leq 0{,}5115666665$. SAC value interval (%) $= 48{,}84333335\% \leq k_{SAC}(i,j) \leq 51{,}15666665\%$.

**Table 11. SAC Value for each Round with Plain Text as an Independent Variable**

| Round No. | Algorithm | | | |
|---|---|---|---|---|
| | KASUMI | | KAMIES | |
| | SAC Value (%) | | SAC Value (%) | |
| | Min | Max | Min | Max |
| 1 | 0 | 100 | 0 | 100 |
| 2 | 0 | 100 | 0 | 100 |
| 3 | **48,956** | **51,366** | 44,873 | 56,876 |
| 4 | 49,03 | 50,98 | **49,016** | **51,026** |
| 5 | 48,856 | 51 | 49,053 | 51,026 |
| 6 | 48,856 | 51 | 48,966 | 51,163 |
| 7 | 48,983 | 50,986 | 48,966 | 51,163 |
| 8 | **48,983** | **51,156** | **49,036** | **50,153** |

The KASUMI algorithm achieves a good diffusion level since the third round. In the third round, the minimum and maximum SAC interval value obtained is at $48,85\% \leq k_{SAC}(i,j) \leq 51,15\%$, while in full round (eight rounds) the minimum and maximum SAC interval value obtained is $48,84\% \leq k_{SAC}(i,j) \leq 51,15\%$. Meanwhile, the KAMIES algorithm achieves a good diffusion level since in the fourth round. In the fourth round, the minimum and maximum SAC interval value obtained is at $49,01\% \leq k_{SAC}(i,j) \leq 51,02\%$, while in the full round (eight rounds) the minimum and maximum SAC interval value obtained is at $49,03\% \leq k_{SAC}(i,j) \leq 50,15\%$. Table 10 shows that the KAMIES has better diffusion properties than the KASUMI algorithm. The test results show that the maximum relative error value allowed is 4%.

### 4.3. AWD Testing Algorithm KASUMI and KAMIES

From Table 12, it can be seen that AWD test on KASUMI and KAMIES algorithms when Plain text as independent variable indicates that both the KASUMI and KAMIES algorithm passed the AWD test with results is above the minimum AWD value of 0.98. AWD testing results can be seen in Figure 6 for KASUMI algorithm and Figure 7 for KAMIES algorithm.

**Table 12. AWD Test Results**

| Round No. | Algorithm | | | |
|---|---|---|---|---|
| | KASUMI | | KAMIES | |
| | Resemblance | | Resemblance | |
| | Min | Max | Min | Max |
| 1 | 0 | 0,0271 | 0 | 0,0278 |
| 2 | 0,0252 | 0,9935 | 0,0249 | 0,9935 |
| 3 | **0,9850** | **0,9940** | 0,9794 | 0,9939 |
| 4 | 0,9859 | 0,9938 | **0,9855** | **0,9945** |
| 5 | 0,9859 | 0,9932 | 0,9853 | 0,9939 |
| 6 | 0,9863 | 0,9940 | 0,9862 | 0,9936 |
| 7 | 0,9848 | 0,9941 | 0,9856 | 0,9934 |
| 8 | 0,9873 | 0,9936 | 0,9844 | 0,9938 |

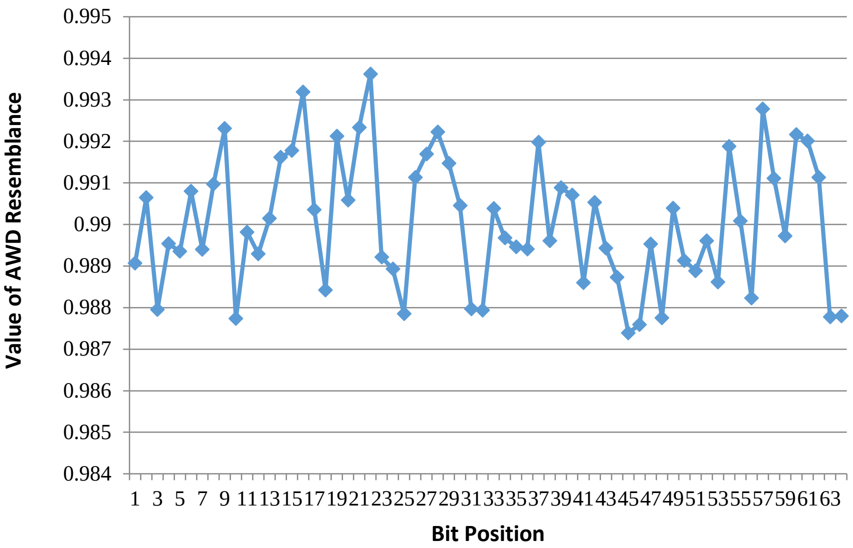

**Figure 6. AWD Resemblance of KASUMI with Plaintext as Independent Variable**

The KASUMI algorithm achieves a good diffusion level since in the third round. In the third round a minimum R value obtained is 0.9850, whereas in full round (eight rounds) a minimum R value obtained is 0.9874. AWD resemblance KASUMI when Plain text is put as independent variable that has a minimum value of 0.9874 at the position of 45th bit change and maximum value of 0.9936 at the position of the 22th bit change. The KAMIES algorithm achieves a good diffusion level since in the fourth round. In the fourth round a minimum R value obtained is 0.9855, while in full round (eight rounds) a minimum R value obtained is 0.9844. AWD resemblance of KAMIES when Plain text is used as independent variable has a minimum value of 0.9843 at the position of the 64th bit change and maximum value of of 0.9937 at the position of the 32th bit change.

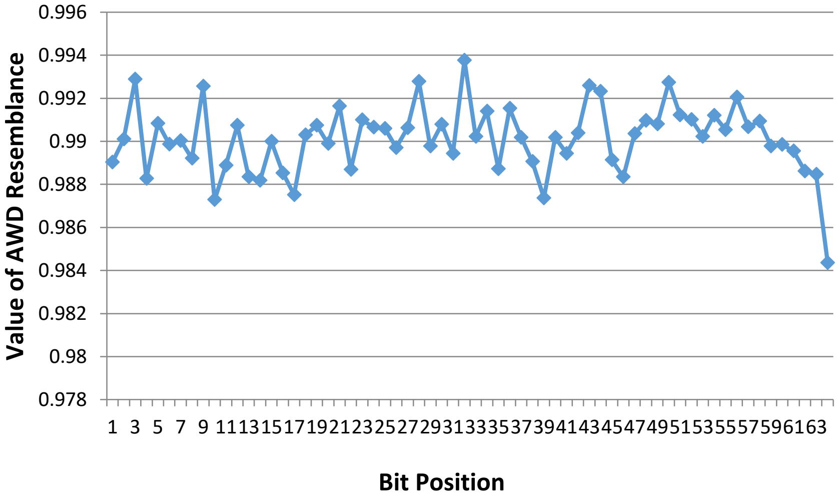

**Figure 7. AWD Resemblance of KAMIES with Plain Text as Independent Variable**

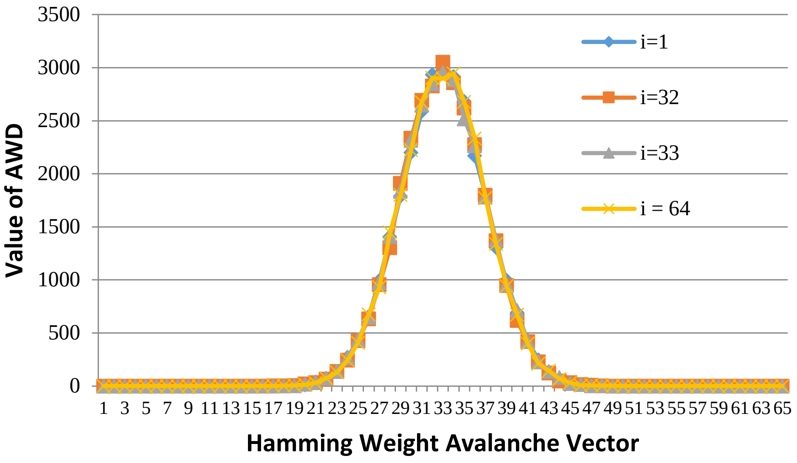

**Figure 8. KASUMI AWD Curve with Plaintext as independent variable**

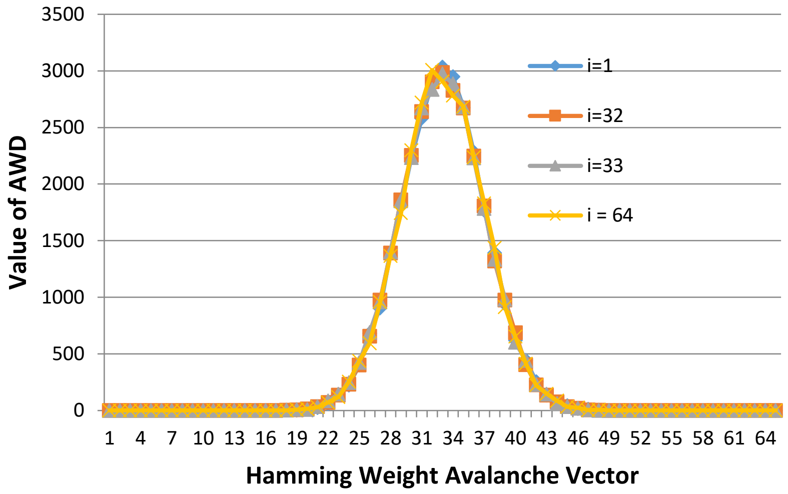

**Figure 9. KAMIES AWD Curve with Plain Text as Independent Variable**

Figure 8 shows the AWD curve of KASUMI algorithm with Plain text as independent variable, where for the position of bit change $i$ = 1, 32, 33, and 64, the resulted curve is similar to the expected Binomial curve. For $i$ = 1, 32, 33, and 64, the peak point on the Hamming Weight avalanche vector j is 32, 33, 32, and 33, respectively. In Figure 9 we can see the AWD curves of KAMIES with Plain text as independent variables for the position of bit change $i$ = 1, 32, 33, 64 the resulting curve is similar to the expected Binomial curve. For i = 1, 32, 33, and 64, the peak point on the Hamming Weight avalanche vector j is 33, 32, 33, and 32, respectively. This indicates that the Plain text elements are equally distributed and do not produce regular patterns.

## 5. Conclusion

In this work, we have presented KAMIES algorithm, a security optimization of KASUMI algorithm. The design on FI function of KAMIES is to replace the F function and S-Box ($S_7$ and $S_9$) of KASUMI with F Function of MISTY1 and S-Box ($S_8$) of AES, respectively. Structural difference of KAMIES algorithm compared to KASUMI algorithm lies on the FI function where in the new FI function there is an S-Box. The original input of 16 bits is divided into two parts with the size of 9 bits and 7 bits converted into 8 bits each. Based on the SAC test, FI and FO function of either KASUMI or KAMIES have a better diffusion level than FI and FO function of MISTY1, whereas FL function of KASUMI, MISTY1 and KAMIES have the same diffusion level. Based on the BIC test, FI and FO function of either KASUMI or KAMIES have a better BIC characteristics than FI and FO function of MISTY1. Based on the S-Box test, the AC result of $S_8$ of AES is better than $S_7$ and $S_9$ of either KASUMI or MISTY1. For SAC and BIC tests, $S_8$ of AES showed better values than $S_9$ of either KASUMI or MISTY1. Comparison of SAC and AWD test results in a full round algorithm showed that the KAMIES algorithm has a better diffusion rate compared to the KASUMI algorithm. It can be concluded that the security level of the KAMIES algorithm is better than the KASUMI algorithm.

## References

[1] F. Qazi, F. H. Khan, K. N. Kiani, S. Ahmed and S. A. Khan, “Enhancing the Security of Communication Using Encryption Algorithm Based on ASCII Values of Data”, International Journal of Security and Its Applications, **(2017)**, pp. 59-68.

[2] M. T. Matsui and Toshio, “MISTY, KASUMI and Camellia Cipher Algorithm Development”, Mitsibishi Electric Advance. Mitsibishi Electric corp., vol. 100, **(2000)**, pp. 2-8.

[3] ETSI/SAGE, “Specification of the 3GPP Confidentiality and Integrity Algorithms”, Document 2: KASUMI Specification, ed, **(1999)**.

[4] K. Jia, L. Li, C. Rechberger, J. Chen and X. Wang, “Improved Cryptanalysis of the Block Cipher KASUMI”, Selected Areas in Cryptography: 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers, L. R. Knudsen and H. Wu, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, **(2013)**, pp. 222-233.

[5] E. Biham, O. Dunkelman and N. Keller, “A related-key rectangle attack on the full KASUMI”, Asiacrypt, vol. 3788, **(2005)**, pp. 443-461.

[6] S. Akleylek, “On the Avalanche Properties of MISTY1, KASUMI, and KASUMI-R”, Cryptography, Middle East Technical University, Ankara, Turkey, **(2008)**.

[7] A. B. On, “Improved Higher-Order Differential Attacks on MISTY1”, IACR Cryptology ePrint Archive, vol. 746, **(2015)**.

[8] Y. Todo, “Integral Cryptanalysis on Full MISTY1”, Journal of Cryptology, vol. 30, **(2017)** July 01, pp. 920-959.

[9] FIPS.197, “Announcing the Advanced Encryption Standard (AES)”, ed., **(2001)**.

[10] S. Kavut and M. D. Yucel, “On Some Cryptographic Properties of Rijndael”, presented at the Proceedings of the International Workshop on Information Assurance in Computer Networks: Methods, Models, and Architectures for Network Security, **(2001)**.

[11] A. J. Menezes, P. C. v. Oorschot and S. A. Vanstone, “Handbook of Applied Cryptography: CRC Press”, **(1996)**.

[12] D. Toz, A. Doğanaksoy and M. S. Turun, “Statistical Analysis of Block Ciphers”, Ulusal Kriptologi Sempozyumu, **(2005)**, pp. 56-66.

[13] C. E. Shannon, “Communication theory of secrecy systems”, Bell Syst. Tech, vol. 28 no. 4, **(1949)**, pp. 656-715.

[14] F. Bavaud, J.-C. Chappelier and J. Kohlas, “An Introduction to Information Theory and Applications”, **(2005)**.

[15] B. Schneier, “Applied Cryptography, Second Edition: Protocols, Algorthms, and Source Code”, C: John Wiley & Sons, Inc., **(1996)**.

[16] K. Kwangjo, “A Study on the Construction and Analysis of Substitution Boxes for Symmetric Cryptosystems”, Electrical and Computer Engineering, Yokohama National University, **(1990)**.

[17] A. F. Webster and S. E. Tavares, “On the design of S-Boxes”, Advances in Cryptology - CRYPT0 '85, LNCS 218, **(1986)**, pp. 523-534.

[18] A. M. Youssef, “Analysis and Design of Block Ciphers”, Department of Electrical and Cornputer Engineering, Queen's University, **(1997)**.

[19] W. S. Meier, “Fast correlation attacks on certain stream ciphers”, Journal of Cryptology, vol. 1, no. 159, **(1989)**.

[20] L. O’Connor, “On Linear Approximation Tables and Cipher secure against Linear Cryptanalysis”, ISRC-QUT Gardens Point, **(1995)**.

[21] M. Mitsuru, “The first experimental cryptanalysis of the data encryption standard”, Advances in Cryptology, **(1994)**.

[22] I. S. ISO/IEC, “Information technology — Security techniques — Encryption algorithms —Part 3: Block ciphers”, ed., **(2005)**.

# Authors

**Muhammad Salman**, a lecturer and researcher at Computer Engineering, Faculty of Engineering, Universitas Indonesia especially in the field of Multimedia, Network and Information Security. He holds a Master's degree in Information Technology from Monash University, Melbourne, Australia and a Bachelor's degree in Computer Engineering from Universitas Indonesia. He is also a board member of ID-SIRTII (Indonesia Security Incident Response Team on Internet Infrastructure).

**Rizki Yugitama,** received his Bachelor's degree from Sekolah Tinggi Sandi Negara (STSN), and Master's in Computer Network Security from the Electrical Engineering Department, Faculty of Engineering, Universitas Indonesia.

**Amiruddin,** a lecturer at Sekolah Tinggi Sandi Negara (STSN), Indonesia. He received Bachelor's degree in Informatics from Universitas Budi Luhur. He gained his Master's in Information Technology from the Faculty of Computer Science, Universitas Indonesia. He is now pursuing a Ph.D degree from the Electrical Engineering Department, Faculty of Engineering, Universitas Indonesia.

**Riri Fitri Sari,** a Professor of Computer Engineering at the Electrical Engineering Department, Faculty of Engineering, Universitas Indonesia (UI). She graduated with a BSc in Electrical Engineering from UI, an MSc in Software Systems and Parallel Processing from the Department of Computer Science, University of Sheffield, UK, and a PhD in Computer Networks from the School of Computing, University of Leeds, UK. Her current main teaching and research area includes Computer Network, Internet of Things (IoT), Cloud Computing, Vehicle Ad Hoc Networks, and ICT implementation.