Dragon Stream Cipher for Secure Blackbox Cockpit Voice Recorder

Fadira Akmal¹, Surya Michrandi Nasution², Fairuz Azmi³

¹²³Security Laboratory Electrical Engineering Faculty
Telkom University Bandung, Indonesia
¹fadira.akmal@gmail.com ²michrandi@telkomuniversity.ac.id
³worldliner@telkomuniversity.ac.id

Abstract. Aircraft blackbox is a device used to record all aircraft information, which consists of Flight Data Recorder (FDR) and Cockpit Voice Recorder (CVR). Cockpit Voice Recorder contains conversations in the aircraft during the flight. Investigations on aircraft crashes usually take a long time, because it is difficult to find the aircraft blackbox. Then blackbox should have the ability to send information to other places. Aircraft blackbox must have a data security system, data security is a very important part at the time of information exchange process. The system in this research is to perform the encryption and decryption process on Cockpit Voice Recorder by people who are entitled by using Dragon Stream Cipher algorithm. The tests performed are time of data encryption and decryption, and avalanche effect. Result in this paper show us time encryption and decryption are 0.85 seconds and 1.84 second for 30 seconds Cockpit Voice Recorder data with an avalanche effect 48.67%.

1. Introduction
The limited ability of aircraft blackbox in the process of sending information is to be a hassle if an aircraft crash occurs and blackbox is difficult to find, because all the facts cause the occurrence of an accident is in the blackbox. Incident Malaysia airlines flight 370(MH70), Phoenix International Holdings, Inc spends 70 days with dives over 5000 meter into the sea to investigate the crash[5]. From the one example of aircraft accident, aircraft blackbox needs the ability to send information but the information submitted must be maintained its authenticity and secrecy. One technique for securing data is cryptography. According to Rinaldi Munir, Cryptography is a field of science that deals with mathematical techniques related to confidentiality information security, data integrity, and authentication[2].

This Dragon algorithm will be implemented on the recording of Cockpit Voice Recorder by calculating the processing time and avalanche effect. Although the system does not discuss the process of sending data, but the ability of process for data security in a short time is important.

2. Related Works

2.1. Cockpit Voice Recorder
Cockpit Voice Recorder (CVR) is part of the blackbox aircraft that works to record all conversations and sounds during the flight. Cockpit voice recorder is active if the plane engine is switched on, and
start working from the plane engine is turned on until the engine is turned off. This CVR has four channels consisting of three channels used to record conversations crew of the aircraft, between pilots and co pilots, pilots with air traffic control, flight attendants with passengers, while one more channel is used for environmental sound inside and outside the cabin such as noise, rain or lightning. The CVR can record the entire conversation with a duration of 120 minutes from the four channels[1].

2.2. Dragon Stream Cipher
The dragon algorithm includes the type of stream cipher. In 2005 was submitted to the eSTREAM project in the software profile category and has advanced to phase 3. Incorporation of feedback shift register and non-linear function with memory is the basic design of cipher. This state is initialized with 128- or 256-bit key-IV pairs. Each clock of the stream cipher produces 64 bits of keystream, using simple operations on 32-bit words[3].

a. F Function
F function is reversible processing of 192 bits (six 32-bit words) to 192 bits. It takes six 32 bit words as input and produces six 32-bit words as output. These six 32-bit input words are normally defined as a, b, c, d, e, f. And the output words are a’, b’, c’, d’, e’, f’. Also the function has other components called G1, G2, G3, H1, H2, H3. The main operations used in the F function are the xor(+) and the addition modulo $2^{32}$ divide into 3 layers ; pre mixing, s-box and post mixing[3].

$$\text{Input } = \{ a, b, c, d, e, f \}$$

Pre-mixing Layer:
1. $b = b \oplus a$;  
2. $c = c \oplus b$;  
3. $d = d \oplus c$;  
4. $e = e \oplus d$;  
5. $f = f \oplus e$;

S-box Layer:
6. $a = a \oplus G_1(a)$;  
7. $b = b \oplus G_2(b)$;  
8. $c = c \oplus H_1(b)$;  
9. $d = d \oplus G_3(c)$;  
10. $e = e \oplus H_2(d)$;  
11. $f = f \oplus H_3(e)$;

Post-mixing Layer:
12. $a' = a \oplus e$;  
13. $b' = b \oplus d$;  
14. $c' = c \oplus b$;  
15. $d' = d \oplus c$;  
16. $e' = e \oplus d$;

Output = \{ a’, b’, c’, d’, e’, f’ \} \quad \ldots (1)

The G and H functions are non-linear process of 32-bit inputs to 32-bit outputs. Storing a 32×32 s-box requires sixteen gigabytes. Virtual 32-bit s-boxes use small of memory in the process from 8×32 32 s-boxes. Therefore the G and H functions are processed from two 8×32 s-boxes, S1 and S2[3].

b. Initialization
Key initialization is partition into eight 128-bit words, W0 to W7. Dragon-128 and Dragon-256 have simple two-phase initialization system that different only in the first phase. The 64-bit memory M is loaded with the constant value 0x0000447261676E, an ASCII representation of ‘Dragon’[3].
c. Keystream Generation[3]

The keystream generation generating a keystream using the previously initialized internal state, and the last value of M as input. The internal state is now divided into 32-bit words. During each round of the keystream generation, B0, B9, B16, B19, B30, B31 are used as the F function inputs a, b, c, d, e, f. Output in each round of the keystream generation is a 64-bit length keystream k, and an updated state B and M, which can be used again to generate more keystreams.

\[
\text{Input} = \{ K, IV \}
\]

1. \( M = \text{tsf000447261676F0E} \)
2. \( W_0 \ldots W_\gamma = \) 128-bit key and IV
\( K \oplus IV' \oplus IV \oplus K' \oplus IV \oplus K' \oplus IV \oplus K' \oplus IV \)
3. \( K \oplus IV \oplus K' \oplus IV \)

Perform steps 3-8 16 times

\[
\begin{align*}
3. & \quad a \| b \| c \| d = (W_0 \oplus W_6 \oplus W_\gamma) \\
4. & \quad e = f = M \\
5. & \quad \{a', b', c', d', e', f'\} = F(a, b, c, d, e, f) \\
6. & \quad t = (a' \| b' \| c' \| d' \| e' \| f') \oplus W_i \\
7. & \quad W_i = W_{i-1}, 1 \leq i \leq 7 \\
8. & \quad W_0 = t \\
9. & \quad M = e' \| f' \\
\end{align*}
\]

\[
\text{Output} = \{ W_0 \| \ldots \| W_\gamma \}
\]...

... (2)

3. System Design

This system will implement the Dragon stream cipher algorithm, this algorithm will encrypt and decrypt frames per frame from recording of cockpit voice recorder files. Dragon algorithm for this paper using 256 bit key and IV, and use symmetric key for encryption and decryption process.

![Figure 1. Dragon encryption process](image)

Encryption process start from divide Cockpit Voice Recorder data .wav into frame and also convert into binary file .txt, frame data will be encrypt one by one with keystream from keystream generation processed, then the frame data can send to ground or another place. This paper only discuss about system how to secure data , not discuss about sending data. System for secure cockpit voice recorder on encryption processing time should not take a long time.
Figure above show the process when frame data on the ground, decryption process will decrypt frame by frame with same keystream on encryption process. After the decryption process is complete, system will concat file frame.txt into CVR data.wav like before. Decryption process may take longer than encryption process, cause there is no process that must be done in the system.

4. Experimental Testing

Experimental testing is performed how long time from encrypt and decrypt cockpit voice recorder files and calculates the avalanche effect of the Dragon algorithm that implemented on the data. This paper testing with 30 seconds cockpit voice recorder duration, with data capacity 840KB.

4.1. Encryption and Decryption Time

**Table 1.** CVR encryption data and time

| Testing | 200 frame(s) | 300 frame(s) |
|---------|--------------|--------------|
| 1       | 0.596096     | 0.736012     |
| 2       | 0.894002     | 1.117502     |
| 3       | 0.601053     | 0.751312     |
| 4       | 0.872117     | 1.090142     |
| 5       | 1.406586     | 1.687886     |
| 6       | 0.842201     | 1.01064      |
| 7       | 0.609878     | 0.731853     |
| 8       | 1.11603      | 1.302035     |
| 9       | 0.883208     | 1.057408     |
| 10      | 0.718711     | 0.928767     |

**Table 2.** CVR decryption data and time

| Testing | 200 frame(s) | 300 frame(s) |
|---------|--------------|--------------|
| 1       | 11.889367    | 0.845670     |
| 2       | 0.790354     | 0.790045     |
| 3       | 0.5325493    | 0.673410     |
Tables above showed how long waktu encryption and decryption for cockpit voice recorder in 10 times testing with different frame, 200 frames and 300 frames. We can know that the larger number of frames the longer time we need to process CVR. The longest encryption process exists on the 5th test at 300 frames, while the longest decryption process exists on the 1st test at 200 frames.

Figure 3a. Graph encryption time

Figure 3b. Graph decryption time

Figure 3a showed encryption process between 200 frames and 300 frames, the graph showed time for encryption process relatively stable, although on 5th test time is uphill. Figure 3b describe decryption process 200 frames and 300 frames, on both cases the 1st test always take longer time than other. So we can calculate the average time for encryption and decryption while divide into 200 frames are 0,85 second and 1,84 seconds. We need longer time for 300 frames CVR in encryption and decryption are 1,05 seconds and 0,84seconds.

4.2. Avalanche effect

Avalanche effect is a way to determine whether or not a cryptographic algorithm. By counting avalanche effect, we will know how much the changes that occur in the ciphertext bits due to the encryption process. The higher value of the avalanche effect will make better the level of security algorithms. An algorithm is said to have a good AE value if only one bit changes in the input results in a change of about half the number of bits in its output[4].

\[
Avalanche\ Effect = \frac{\sum \text{bit changed}}{\sum \text{bit total}} \times 100\% \quad \ldots (4)
\]

Here is an avalanche effect table testing of Dragon Algorithm. The first test is on Table 3 with the different audio frame that showed in table and the key and IV are same for each audio frame.

| Key | 00001111 22223333 DDDDDDDD 66667777 88889999 AAAABBBB CCCCCDDD EEEEEFFF | IV | 000011F1 22223333 44445555 68887777 88989999 AAAABBBB CCCCCDDD EEEEEFFF |
|-----|---------------------------------------------------------------------------------|----|---------------------------------------------------------------------------------|
| 4   | 0,672110                                                                        | 5  | 0,723541                                                                        |
| 5   | 0,802844                                                                        | 6  | 0,819987                                                                        |
| 6   | 0,9124202                                                                       | 7  | 1,020                                                                            |
| 7   | 0,600498                                                                        | 8  | 0,885200                                                                        |
| 8   | 1,00603                                                                         | 9  | 0,904258                                                                        |
| 9   | 0,643214                                                                        | 10 | 0,917732                                                                        |
| 10  | 0,629417                                                                        |
Table 3. Avalanche effect testing

| Audio Frame   | Plaintext                        | Ciphertext                      | AE(%) |
|---------------|----------------------------------|---------------------------------|-------|
| Frame1.txt    | 01010010010010010100010100011000 | 10100100100100101001001010011001 | 49.0  |
|               | 00...                             | 00...                            |       |
| Frame2.txt    | 100000001000000010000000100000001 | 100000001000000010000000100000001 | 48.5  |
|               | 00...                             | 00...                            |       |
| Frame3.txt    | 100000001000000010000000100000001 | 100000001000000010000000100000001 | 48.5  |
|               | 00...                             | 00...                            |       |

Table 3 is avalanche effect testing for CVR data divide into 200 frame, the average of avalanche effect for 200 frame is 48.67%. This value including good avalanche effect for security level.

Here the second test is on Table 4 with the same audio frame1.txt that showed in table and different key but IV are same for each audio frame.

Table 4. Avalanche effect testing

| Key            | Plaintext                        | Ciphertext                      | AE(%) |
|----------------|----------------------------------|---------------------------------|-------|
| 00001111       | 22223333                        | 01010010010010010100010100011000 | 49.0  |
| DDDDDDDDD      | 66667777                        | 1010010011001001010001101000    |       |
| 88889999       | AAAABBBB                        | 00000000......                  |       |
| CCCDDEEEFFFF  | 88889999                        | 1001001110010001101000           |       |
|               | AAAABBBB                        | 00000000......                  |       |
| 00001111       | 22223333                        | 01010010010010010100010100011000 | 49.8  |
| DDDDDDDDD      | 66667777                        | 1010010011001001010001101000    |       |
| 88889999       | AAAABBBB                        | 00000000......                  |       |
| CCCDDEEEFFFF  | 88889999                        | 1001001110010001101000           |       |
|               | AAAABBBB                        | 00000000......                  |       |

Table 4 is avalanche effect testing from 1 frame in CVR, the table showed that one change bit in key can change ciphertext(different avalanche effect from three testing). Value of avalanche effect from different key still produce good performance for security level.

5. Analysis
From the Table 1 above, we can see the comparison of the time required by audio with different frame. The 200 frames has duration of time from encryption and decryption is more efficient than the second with 300 frames. The 200 frames needs 0.85 second for encryption and 1.84 seconds for decryption while the 300 frames is needed 1.05 seconds for encryption and 0.84 seconds for decryption. From that comparison, it can be concluded that the number of frames affects the time of encryption and decryption. The more number of frames so the longer the time of the encryption and decryption process. From test results Table 3 above, we can see the average avalanche effect value from the different frame, the same key and IV are 48.67% and Table 4 we can see the changed value of avalanche effect in every different key 49% - 50%. All result shows that the avalanche effect test produce changes bits is about half the number of bits in its output. Cryptographic algorithm will be difficult to resolve when the key used is not known and the results of the output will be very unique. It means Dragon algorithm has a good performance on security level.
6. Conclusion
From the implementation and test results, we draw some concluding remarks, as follows:

Result in this paper show us time encryption and decryption are 0.85 seconds and 1.84 second for 30 seconds Cockpit Voice Recorder data with an avalanche effect 48.67%. The difference of audio frames can affect to time of encryption and decryption. The larger the number of frames, the longer they need time to encrypt and decrypt the audio. The changes bits after testing avalanche effect value is about half of number of bits in its output, it’s means algorithm has good security system. Dragon algorithm has a good performance avalanche effect range value, 48% - 50% on this system.

7. REFERENCES

[1] Air Transport Series. Product Description. Solid-State Voice/Data&Digital Communications Combined Recording Systems for Air Transport Application.

[2] Rinaldi Munir. 2006. Kriptografi. Bandung : Penerbit Informatika Bandung

[3] Ed Dawson, Matt Henricken, and Leonic Simpson. 2005. The Dragon Stream Cipher: Design, Analysis, and Implementation Issues

[4] Siska Selany. 2015. IEEE Sosemanuk Algorithm for Encryption and Decryption Video on Demand (VoD)

[5] Peter K LeHardy and Christopher Moore. 2014. IEEE Deep Ocean Search for Malaysia Airlines Flight 370

[6] Paul. R. Schoberg. 2003. Secure Ground – Based Remote Recording and Archiving Of Aircraft Black Box Data. Monterey California: AMM: Naval Post Graduate School.

[7] Pardeep Kumar and Hoon Jae Lee. 2009. IEEE A Secure Data Mechanism for Ubiquitous Sensor Network with Dragon Cipher. Dongseo University, Busan, South Korea.