Image Encryption based on AES Stream Cipher in Counter Mode

May Thet Khaing, Zin May Aye
University of Computer Studies (Mawlamyine)
maythetkhaing.2008@gmail.com, zinmay110@gmail.com

Abstract

With the first evolution of digital data exchange, information security is becoming more important in data storage and transmission. Because of widely using images in industrial process, it is important to protect the confidential image data from unauthorized access. In this paper, we intend to develop software based image encryption system by applying AES in Counter Mode (AES-CTR) with an explicit initialization vector (IV). IV generation includes incrementing a counter for each packet and linear feedback shift registers (LFSRs). AES-CTR uses the AES block cipher to create stream cipher. AES-CTR uses the only AES encrypt operation for both encryption and decryption, making AES-CTR implementations smaller than implementations of many other AES modes. It is an attractive encryption algorithm for high-speed networking and improving the security of images from unauthorized access.

Keywords: Advanced Encryption Standard (AES), Counter (CTR) Mode, Image, Initialization Vector (IV), Linear Feedback Shift Register (LFSR)

1. Introduction

Today, The World of Information Technology is needed to be secure data and information. Cryptography is a method of storing and transmitting data in a form that only those it is intended for can read and process. It is a science of protecting information by encoding it into an unreadable format. Cryptography has symmetric and asymmetric techniques. Symmetric key algorithms can be divided into stream cipher and block cipher. Stream ciphers encrypt the bits of the message one at a time, and block ciphers take a number of bits and encrypt them as a single unit.

Security is an important issue in communication and storage of images, and encryption is one of the ways to ensure security. Image encryption techniques try to convert an image to another one that is hard to understand. Image encryption has applications in internet communication, multimedia systems, medical imaging, telemedicine, military communication, etc.

AES is very fast symmetric block algorithm especially by hardware implementation [4]. AES is one of the most popular algorithms used in symmetric key cryptography. AES allows for three different key lengths: 128, 192 or 256 bits. Encryption consists of 10 rounds of processing for 128 bit keys, 12 rounds for 192 bit keys, and 14 rounds for 256 bit keys.

Counter mode is a symmetric-key encryption scheme based on any block cipher, e.g., AES. CTR mode is encrypted and XORed with plaintext to produce cipher text. In CTR mode, there is no feedback and preprocessing can be used, in some environments, to increase speed. With CTR mode, both encryption and decryption depend only on encryption.

AES-CTR has many properties that make it attractive encryption algorithm for in high-speed networking. AES-CTR is easy to implement and can be pipelined and parallelized. AES-CTR uses the only AES encrypt operation.

LFSR is used to modify the AES algorithm. LFSR can be easily implemented in hardware or software and is used to create a pseudo-random sequence of numbers for many different applications.

This paper is organized as follows: Section 2 summarizes the related work in image encryption using AES in CTR mode. In Section 3, the background theory is explained. Section 4 presents the implementation of image encryption system. The last section; Section 5 is the conclusion.

2. Related Work

Housley [5] described the use of AES-CTR with an explicit initialization vector, as an IPSec Encapsulating Security Payload (ESP) confidentiality mechanism.

Zeghid et al. [9] analyzed the AES, and added a key steam generator (A5/W7) to AES to ensure improving the encryption performance; mainly for images characterized by reduced entropy. The implementation of both techniques has been realized for experimental purposes.

Mcgrew [8] described Counter Mode and its security properties, reviewing relevant cryptographic attacks and system security aspects. This mode is well understood and can be implemented securely. However, Mcgrew shows that attacks using precomputation can be used to lower the security level of AES-128 CTR below the recommended strength for ciphers if the initial counter value is predictable.

3. Background Theory

In this section, brief discussions about cryptography, AES and Counter Mode operation are presented.

3.1 Advanced Encryption Standard

AES is an encryption standard adopted by the U.S.government. Encryption is the process of transforming the information to insure its security. AES is a symmetric key block cipher algorithm standardized by the U.S. National Institute of Standard and Technologies (NIST) [1]. AES has a fixed block size of 128 bits and a key size of 128, 192 or 256 bits. With AES both the encryption and the decryption are performed using the same key. This is called a symmetric encryption algorithm. Encryption consists of 10 rounds for 128 bit keys, 12 rounds for 192 bit keys, and 14 rounds for 256 bit keys. Each round consists of several processing steps, including one that depends on the encryption key. AES operates on a 4×4 array of bytes. Each round consists of four basic operations Subbytes, Shiftrows, Mixcolumns and Addroundkey.

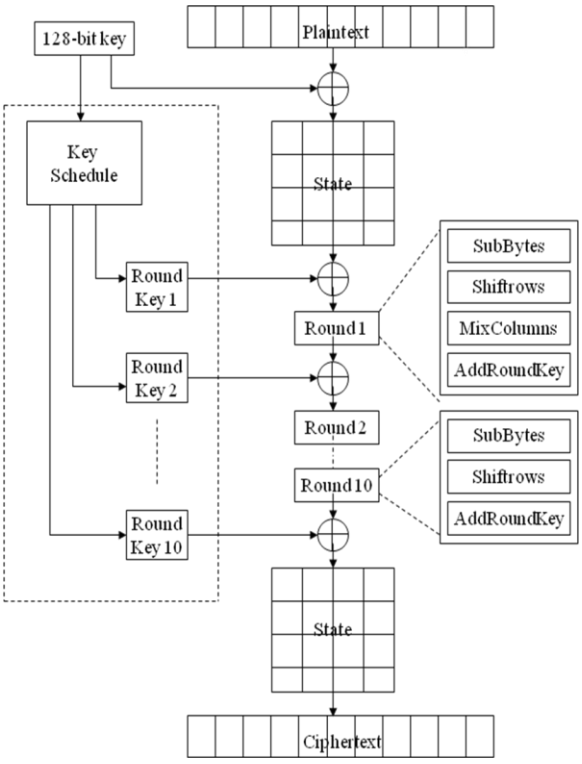

Figure1: Block Diagram of AES Encryption Process

- **AddRoundKey** - each byte of the state is combined with the round key; each round key is derived from the cipher key using a key schedule.
- **SubBytes** – a non-linear substitution step where each byte is replaced with another according to a lookup table.
- **ShiftRows** – a transposition step where each row of the state is shifted cyclically a certain number of steps.
- **MixColumns** – a mixing operation which operates on the columns of the state, combining the four bytes in each column using a linear transformation.

The last round for encryption does not involve the "MixColumns" step. The encryption flow starts with the addition of the initial key to the plaintext. Then the iteration continues for (Nr - 1) rounds (Nr being the total number of rounds). In last round, the MixColumn step is bypassed as shown in Figure 1.

3.2 Counter Mode (CTR)

CTR mode is a symmetric-key encryption scheme based on any block cipher, e.g. AES [6]. It is useful for sending secret data without preserved data integrity. CTR mode has a proven-tight security and it enables the simultaneous processing of multiple blocks without losing the feedback mode advantages. It also gives the advantage of allowing the use of similar hardware for both encryption and decryption [3]. In the CTR mode, the block cipher is used to encrypt a counter value which is incremented for successive encryptions. The encrypted counter values makeup a keystream which is XORed with the plaintext bits to produce the cipher text bits [7]. In CTR mode, there is no feedback. Counter mode encryption and decryption are shown in Figure 2.

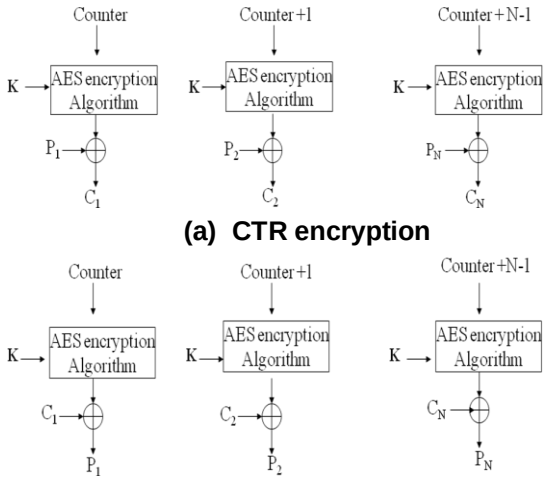

Figure 2: Counter Mode Encryption and Decryption

CTR mode has significant implementation advantages as follows:

- Software efficiency: By eliminating the computational dependency between C_i and C_j, CTR mode encryption enables effective utilization of aggressive pipelining, multiple instruction dispatch per clock cycle, a large number of registers, and SIMD instructions.

- Hardware efficiency: CTR model is fully parallelizable. One can be computing blocks C_1; C_2... all at the same time, limited only by the amount of hardware that one throws at the problem.
- Preprocessing: Preprocessing can be used in some environments, to increase speed. That is, one can compute the pad in "spare cycles," even before one knows the plaintext *M*. When *M* is known, it is XORed with the already-computed pad.
- Random-access: The i^{th} cipher text block, C_i, can be encrypted in a random-access fashion. In hard-disk encryption, this is important because bulk data needs to be encrypted quickly. When using CBC mode, properly encrypting the i^{th} block requires one to first encrypt the i - 1 prior block.

3.3 Linear Feedback Shift Register

Today LFSR's are present in nearly every coding scheme as they produce sequences with good statistical properties, and they can be easily analyzed. LFSR is a shift register. A LFSR with a well-chosen feedback function can produce a sequence of bits which appears random in nature and which has a very long cycle [2].

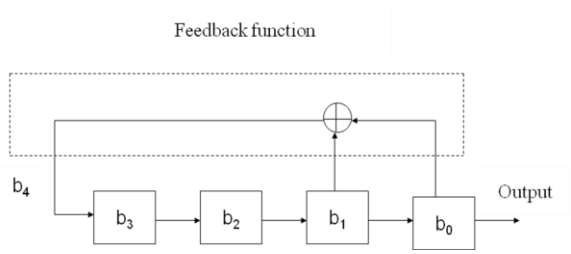

Figure 3: Illustration of LFSR

This system uses 128 bit AES encryption algorithm so it is needed to generate 128 bit LFSR key. To overcome the Key Distribution Problem in AES, LFSR random key stream generator is applied to secure the key. We will use LFSR key to encrypt the image file. A linear feedback shift register (LFSR) sequence is a pseudo-random sequence of numbers that is often created in a hardware implementation of a linear feedback shift register. Equation (1) is the linear function of LFSR where b_m is a linear function of b_0, b_1 …b_{m-1}.

$$b_m = c_{m-1}b_{m-1} + ... + c_2b_2 + c_1b_1 + c_0b_0 \quad (c_0 \neq 0) \qquad (1)$$

However, we are dealing with binary digits because the multiplication and addition are in the GF (2) field, so the value of c_i is either 1 or 0, but c_0 should be 1 to get a feedback from the output. The Equation (2) provides exclusive-or operation in LFSR. In other words, it can be written as

$$b_m = c_{m-1}b_{m-1} \oplus c_2b_2 \oplus c_1b_1 \oplus c_0b_0 \quad (c_0 \neq 0) \qquad (2)$$

Figure 3 shows illustration of an LFSR. An LFSR has a maximum period of 2^m-1 if it has an even number of cells and the characteristic polynomial is a primitive polynomial.

4. Implementation of Image Encryption System

In this section, the main process of image encryption system is described. This process has two phases: the encryption and the decryption.

4.1 Encryption Process of Sender Side

In Figure 4, the user has to select input original image to encrypt. And then the color pixels of image are transformed into the binary data to obtain binary formats. Then, this binary format is encrypted with encryption keys to obtain cipher image.

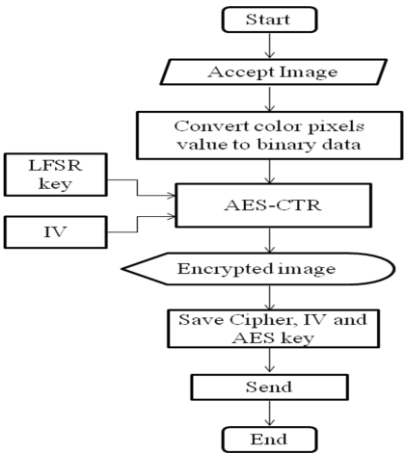

Figure 4: Encryption Process of Sender Side

Csharp language provides random function for the generation of the IV. AES key is randomly generated from LFSR are applied in the encryption process.

After the encryption process is completed, the cipher, IV and AES key are saved into the user specified place. Then, the user can send cipher, IV and AES key to the receiver.

Figure 5: Original Image

This system, input image can be JPEG, PNG, and GIF format. The original input image is shown in Figure 5.

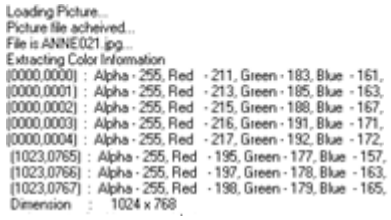

Figure 6: Extract Color Pixels Value of Image

Figure 6 depicts the color pixels value extracted from the received image.

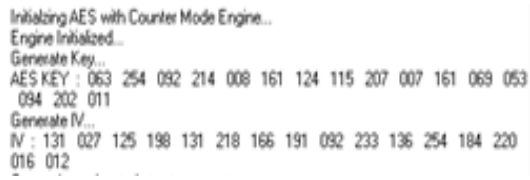

Figure 7: Initializing Key Generator LFSR and IV

For attaining better key security, key generator LFSR is employed for AES key and IV for CTR. Figure 7 describes the key generation from the LFSR.

Figure 8: Convert Color Pixels to Binary Format

The resulting RGB colors are changed into byte format as shown in Figure 8. As soon as, the image is encrypted, the cipher image appears on the screen. Then, the cipher image (Figure 9), IV and AES key are ready to be sent to the receiver through insecure channel.

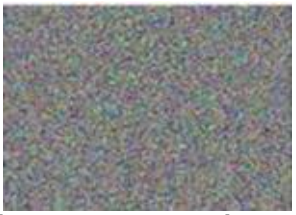

Figure 9: Encrypted Image

4.2 Decryption Process of Receiver Side

In the decryption process, the receiver decrypts his/her receive cipher to obtain original image. To decrypt the cipher, the receiver must use his/her receive keys (AES key and IV). And then the receiver receives the decrypted image with binary format. Finally, this binary format is transformed into the color format to have original image.

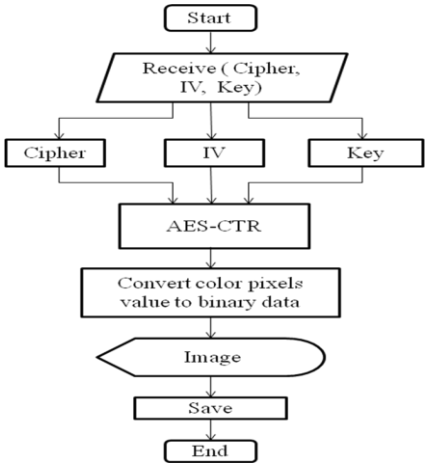

Figure 10: Decryption Process of Receiver Side

In receiver side (Figure 10), the receiver decrypts all the cipher image, IV and AES key. Then, he/she needs to decrypt all encrypted data and image using the appropriate procedure. Finally, the receiver can view the original image (Figure 11) sent by the sender.

Figure 11: Decrypted image

5. Conclusion

In this paper, images are encrypted by using AES stream cipher in counter mode. AES is fast in both hardware and software, and requires little memory. The AES is extended to support key stream generator for image encryption. The key stream generator has an important influence on the encryption performance. AES-CTR can provide high performance confidentiality. AES-CTR mode is applied in this system because it can be parallelized and pipelined in encryption process for it has the advantage of better performance rather than using other modes.

References

[1] National Institute of Standards and Technology, “Announcing the Advanced Encryption Standard (AES)”, FIPS PUB 197, Nov 2001.

[2] Doshi N. A., Dhobale S. B., and Kakade S. R, “LFSR Counter Implementation in CMOS VLSI”. World Academy of Science, Engineering and Technology 48 2008

[3] F. Charot1, E. Yahya2, and C.Wagner1, “Efficient Modular-Pipelined AES Implementation in Counter Mode on ALTERA FPGA”, France.

[4] K.Janvinen, M.Tominisko, J.Skytta, "A fully pipelined memorlyess 17, 8 Gpbs AES-128 encryptor", in *International symposium of Field programmable Gate arrays*, 2003, pp.207-215.

[5] R.Housley, “Using Advanced Encryption Standard (AES) Counter Mode With IPSec Encapsulating Security Payload (ESP)”. Request for Comments 3868, Network Working Group, Standard Track, January 2004.

[6] W. S. HsienHsin, “High Efficiency Counter Mode Security Architecture via Prediction and Precomputation”. School of Electrical and Collage of Computing, Georgia Institute of Technology, Atlanta, GA 30332

[7] C. Mathur, “A Mathematical Framework for Combining Error Correction and Encryption”. STEVENS INSTITUTE OF TECHNOLOGY, Castle Point on Hudson, Hoboken, NJ 07030, 2007.

[8] D. A. McGrew, “Counter Mode Security: Analysis and Recommendations”. Cisco Systems, Inc. November 15, 2002.

[9] M. Zeghid, M. Machhout, L. Khriji, A. Baganne, and R. Tourki, “A Modified AES Based Algorithm for Image Encryption”. Proceeding of world ACADEMY of science, engineering and technology volume 21 May 2007 ISSN 1307-6884.