Journal of the ACS, Vol. 9 , May 2018

A Robust Satellite Imagery On-Board Security System

Dr. Mohamed Ahmed Hussein Ali

Computer Science Department

High Institute for Computers and Information Systems
Al- Shorouk Academy, Cairo – Egypt
e.mail: drmah2006@yahoo.com

Abstract

Advanced information systems are needed to ensure the rapid, accurate exchange of vital information related to Satellite of remote sensing safety. There is a great demand of satellite image security system for providing secure storage and transmission of satellite images. As the demand to protect the sensitive and valuable data from satellites has increased a secure system for satellite imagery is implemented in this paper.

Secure the data comes from satellites to ground stations by encrypting the data and put authority. This done by developing a software using encryption code impeded on raspberry pi as a complete system. Some encryption techniques are considered to implement include AES, DES, RSA, Twofish and blowfish. A proposed method for satellite image security by combining Encryption algorithms for more security and preventing information leakage is presented. The experimental results demonstrate the efficiency of the proposed scheme, which fulfils the strict requirements concerning alterations of satellite images.

Key Words: *Satellite images, Cryptography, Encryption, RSA, AES, DES, Public-Key encryption , Twofish and blowfish.*

1. Introduction

The rapid development, advancement, and growing use of satellite imagery and information technologies have made the security of data storage and transmission essential to prevent unauthorized, and illegal, so protection method must be applied for satellite image by encrypting them with a robust high quality encrypting algorithm fits with the situation of satellite communication standard and its higher priority. The term encryption refers to the practice of obscuring the meaning of a piece of information by encoding it in such a way that it can only be decoded, read and understood by people for whom the information is intended.[1]

In this paper an encryption system is developed in which combination of examined selected encryption algorithms with proposed encryption algorithm apply on satellite image to increase the complexity and difficulty on the hackers.

2.Security and encryption techniques

A secure computing environment would not be complete without consideration of encryption technology.

2.1 Encryption:

Encryption is to generate an alternate form of the original data. The data is passed through a series of mathematical operations that; the sequence of these operations is called an algorithm.[1] To help distinguish between the two forms of data, the unencrypted data is referred to as the plaintext and the encrypted data as ciphertext. The security of encryption lies in the ability of an algorithm to generate ciphertext that is not easily reverted to the original plaintext. [1,2,3]

2.2 Keys:

In the quest for a more secure method of protecting information, the introduction of a key adds another level of security. A key is a piece of information that allows only those that hold it to encode and decode a message. Keys come in many different forms such as passwords or numbers generated by an algorithm. It is a series of numbers or symbols that are used to encode a message so that it can only be read by someone in possession of that key or a related key. A key allows both the sender and the recipient of the message to understand how the message has been encrypted and assures them that nobody else knows how it has been encrypted. It is the key that enables the recipient to properly decode the message. [3,4,5]

2.3 Symmetric and Asymmetric Encryption:

There are two general categories for key-based encryption - symmetric and asymmetric. Symmetric encryption uses a single key to encrypt and decrypt the message. To use symmetric encryption, the sender encrypts the message and, if the recipient does not already have a key, sends the key and ciphertext separately to the recipient. The recipient then uses the key to decrypt the message. This method is easy and fast to implement but has weaknesses; for instance, if an attacker intercepts the key, they can also decrypt the messages. Furthermore, single key encryptions tend to be easier for people to crack, which means that the algorithm that is used to encode the message is easier for attackers to understand, enabling them to more easily decode the message.

Asymmetric encryption, also known as Public-Key encryption, uses two different keys - a public key to encrypt the message, and a private key to decrypt it. The public key can only be used to encrypt the message and the private key can only be used to decrypt it. This allows a user to freely distribute his or her public key to people who are likely to want to communicate with him or her without worry of compromise because only someone with the private key can decrypt a message. To secure information between two users, the sender encrypts the message using the public key of the receiver. The receiver then uses the private key to decrypt the message. Unlike with single or shared keys, in the asymmetric key system only the recipient can decrypt a message; once the sender has encrypted the message he or she cannot decrypt it. The private key is never distributed, therefore an attacker cannot intercept a key that decrypts the message. [8,9]

2.4 Block cipher

A block cipher is an encryption algorithm that encrypts a fixed size of n-bits of data - known as a block - at one time. The usual sizes of each block are 64 bits, 128 bits, and 256 bits. So for example, a 64-bit block cipher will take in 64 bits of plaintext and encrypt it into 64 bits of ciphertext. In cases where bits of plaintext is shorter than the block size, padding schemes are called into play. Majority of the symmetric ciphers used today are actually block ciphers. DES, Triple DES, AES, IDEA, and Blowfish are some of the commonly used encryption algorithms that fall under this group. [12]

2.5 Stream cipher

A stream cipher is an encryption algorithm that encrypts 1 bit or byte of plaintext at a time. It uses an infinite stream of pseudorandom bits as the key. For a

stream cipher implementation to remain secure, its pseudorandom generator should be unpredictable and the key should never be reused. Stream ciphers are designed to approximate an idealized cipher, known as the One-Time Pad.

The One-Time Pad, which is supposed to employ a purely random key, can potentially achieve "perfect secrecy". That is, it's supposed to be fully immune to brute force attacks. The problem with the one-time pad is that, in order to create such a cipher, its key should be as long as or even longer than the plaintext.

Clearly, while Top Secret information or matters of national security may warrant the use of a one-time pad, such a cipher would just be too impractical for day-to-day public use. The key of a stream cipher is no longer as long as the original message. Hence, it can no longer guarantee "perfect secrecy". However, it can still achieve a strong level of security.[3]

3. Encryption Algorithms

3.1 Shifting

The simple substitution cipher is a cipher that has been in use for many hundreds of years. It basically consists of substituting every plaintext character for a different ciphertext character. It differs from the Caesar cipher in that the cipher alphabet is not simply the alphabet shifted, it is completely jumbled.

The simple substitution cipher offers very little communication security, and it will be shown that it can be easily broken even by hand, especially as the messages become longer (more than several hundred ciphertext characters).

3.2 RSA

Rivest Shamir Aldeman(RSA) is the most commonly used public key encryption algorithm. RSA computation occurs with integers modulo n = p*q. It requires keys of at least 1024 bits for good security. Keys of size 2048 bit provide best security. Widely used for secure communication channel and for authentication to identity service provider. RSA is too slow for encrypting large volumes of data but it is widely used for key distribution. Following steps are followed in RSA to generate the public and private keys.[13]

- Consider two large prime numbers p and q such that p~=q.
- Compute n=p*q
- Compute φ (pq) = (p-1)*(q-1)
- Consider the public key k1 such that gcd (φ (n), k1) =1; 1<k1<φ (n) such

that GCD is (Greatest Common Divisor)

- Select the private key k2 such that k2*k mod φ (n) =1

Encryption and Decryption are done as follow :

- Encryption :

Calculate ciphertext C from plaintext P such that:

C=P ^k1 mod n

- Decryption :

P=C^k2 mod n=P^k1k2 mod n

3.3 LSB

Least Significant Bit (LSB) is a substitution method popularly used for embedding secret message. It involves the following steps:

- Convert text into binary equivalent.
- Get pixel value of each pixel one by one.
- Replace each bit of ciphertext with last bit of each pixel in image.

As human eye is not very sensitive, after embedding data in a cover file, our eye cannot find difference between original image and data after inserting in the image.

3.4 DES

Data Encryption standard (DES) mainly adopted by industry for security products. Algorithm design for encryption and decryption process has been done with same key. This algorithm processes the following steps.

- DES accepts an input of 64-bit long plaintext and 56-bitkey (8 bits of parity) and produce output of 64 bit block.
- The plaintext block has to shift the bits around.
- The 8 parity bits are removed from the key by subjecting the key to its Key Permutation.
- The plaintext and key will processed by following:

a. The key is split into two 28 halves

b. Each half of the key is shifted (rotated) by one or two bits, depending on the round.

c. The halves are recombined and subject to a compression permutation to reduce the key from 56 bits to 48 bits. This compressed keys used to encrypt this round's plaintext block.

d. The rotated key halves from step 2 are used in next round.

e. The data block is split into two 32-bit halves.
f. One half is subject to an expansion permutation to increase its size to 48 bits.
g. Output of step 6 is exclusive-OR'ed with the 48-it compressed key from step 3.
h. Output of step 7 is fed into an S-box, which substitutes key bits and reduces the 48-bit block back down to 32-bits.
i. Output of step 8 is subject to a P-box to permute the bits.
j. The output from the P-box is exclusive-OR'ed with other half of the data block.
k. The two data halves are swapped and become the next round's input.

3.5 Blowfish

Blowfish is a symmetric encryption algorithm, Blowfish is also a block cipher. This symmetric cipher splits messages into blocks of 64 bits and encrypts them individually.

The function splits the 32-bit input into four eight-bit quarters, and uses the quarters as input to the S-boxes. The outputs are added modulo 232 and XORed to produce the final 32-bit output.[14]

3.6 AES

Advanced Encryption Standard (AES) algorithm not only for security but also for great speed. Both hardware and software implementation are faster still.

New encryption standard recommended by NIST to replace DES. Encrypts data blocks of 128 bits in 10, 12 and 14 round depending on key size. It can be implemented on various platforms especially in small devices. It is carefully tested for many security applications. The following steps processed in AES algorithm. [7,10]

Following steps used to encrypt a 128-bit block:

- Derive the set of round keys from the cipher key.
- Initialize the state array with the block data (plaintext).
- Add the initial round key to the starting state array.
- Perform nine rounds of state manipulation.
- Perform the tenth and final round of state manipulation.
- Copy the final state array out as the encrypted data.

Each round of the encryption process requires a series of steps to alter the state of array. These steps involve four types of operations. They are:

a. Sub Bytes: This operation is a simple substitution that converts every bite into a different value.
b. ShiftRows: Each row is rotated to the right by a certain number of bytes.
c. MixColumns: Each column of the state array is processed separately to produce a new column. The new column replaces the old one.
d. XorRoundKey: This operation simply takes the existing state array,

3.7 RC4

The RC4 Encryption Algorithm, developed by Ronald Rivest of (RSA), is a shared key stream cipher algorithm requiring a secure exchange of a shared key. The symmetric key algorithm is used identically for encryption and decryption such that the data stream is simply XORed with the generated key sequence. The algorithm is serial as it requires successive exchanges of state entries based on the key sequence. Hence implementations can be very computationally intensive. The RC4 encryption algorithm is used by standards such as IEEE 802.11 within WEP (Wireless Encryption Protocol) using 40 and 128-bit keys. Published procedures exist for cracking the security measures as implemented in WEP. [11]

In the RC4 encryption algorithm, the key stream is completely independent of the plaintext used. An 8 * 8 S-Box (S0-S255), where each of the entries is a permutation of the numbers 0 to 255, and the permutation is a function of the variable length key. There are two counters i, and j, both initialized to 0 used in the algorithm.

The algorithm uses a variable length key from 1 to 256 bytes to initialize a 256-byte state table. The state table is used for subsequent generation of pseudo-random bytes and then to generate a pseudo-random stream which is XORed with the plaintext to give the ciphertext. Each element in the state table is swapped at least once.

The key is often limited to 40 bits, because of export restrictions but it is sometimes used as a 128 bit key. It has the capability of using keys between 1 and 2048 bits. RC4 is used in many commercial software packages such as Lotus Notes and Oracle Secure SQL.

The algorithm works in two phases, key setup and ciphering. Key setup is the first and most difficult phase of this encryption algorithm. During a N-bit key setup (N being your key length), the encryption key is used to generate an encrypting variable using two arrays, state and key, and N-number of mixing operations. These mixing operations consist of swapping bytes, modulo operations, and other formulas. A modulo operation is the process of yielding a remainder from division.[11]

3.8 Comparison between algorithms of encryption techniques:

To develop a robust algorithm for encryption, a comparison between the most common algorithms is made to summarize the powerful of each in both software and hardware. Table 3.1 shows these comparisons.

Table 3.1 Encryption techniques comparison

Algorithm	Block size	Key size	Speed	Security	Power consumption
Blowfish	64 bits	32->448 bits	Fast	vulnerable	low
DES	64 bits	56 bits	low	low	low
3DES	128 bits	112 ,168 bits	low	Low stronger than DES	high
AES	128 bits	128, 192 ,256 bits	fast	Very high	low
RSA	32 , 64 , 128 bits	>1024 bits	slow	vulnerable	high

4. Proposed encryption and decryption algorithm

A proposed algorithm is provided in this paper. The main idea is a stream cipher with symmetric key algorithm. It's complex combination of both substitution and permutation, and is fast in both software and hardware.

The decryption algorithm is reverse of encryption is simply XORed with the generated key sequence to give the ciphertext.

The key stream is completely independent of the plaintext used. It uses a variable length key from one to file length (plaintext) bit to initialize byte array key, Key-stream should be as long as plaintext.

The Randomness of stream key completely, Non-linear combining functions and a Juggling Algorithm destroy statistically properties in message.

The algorithm is stream of pseudo-random digits that generated independently of the plaintext and ciphertext file, and then combined with the plaintext (to encrypt) or the cipher text (to decrypt). In the most common form, binary digits are used (bits), and the keystream is combined with the plaintext using the exclusive or operation (XOR). This is termed a binary additive stream cipher.

In order to make a stream cipher more difficult to crack, one could use a crypto key which varies in length. This would help to mask any discernible patterns in the resulting ciphertext. In fact, by randomly changing the crypto key used on each bit of data, one can produce ciphertext that is mathematically impossible to crack. This is because using different random keys would not generate any repeating patterns which can give a cracker the clues required to break the crypto key.

4.1 The proposed encryption steps:

The proposed encryotion algorithm can be summarized in the following steps:

- Get the data in array byte form (input[i]) to be encrypted and the selected key.

- Create Key-stream array of byte (arrykey[i]) has same length of input_array where selected key is repeated as long as plaintext.
- Make Randomize substitution transformation by selecting even integer value of element of input[i], its even index(i) and even element of arrykey[i], using combining the exclusive or operation (XOR) and summation.

  input[i]=(byte)(((input[i]+i)^i))^arrykey[i]).
- And selecting odd value of element of input[i], its odd index(i) and odd element of arrykey[i], using combining the exclusive or operation (XOR) and subtraction.

  input[i]=(byte)(((input[i]-i)^i))^arrykey[i]).
- Make permutation transformation by applying array reverse technique at input[i].
- Perform combination of operation XOR with pseudo-random key generation.

  input[i]=(byte)(input[i]^(arrykey[i]^i)).
- Perform array rotation function by applying a juggling Algorithm. Instead of moving one by one, divide the array in different sets where number of sets is equal to GCD of n and d and move the elements within sets.
-

If GCD is 1 ,then elements will be moved within one set only, we just start with temp = input[0] and keep moving input[I+d] to input[I] and finally store temp at the right place, finally to get ciphertext .

The proposed decryption algorithm is reverse steps of encryption algorithm.

4.2 Features of the proposed technique

The features of the proposed algorithm can be summarized as:

- Symmetric stream cipher.
- Variable key length.
- Very quick in software and hardware.

The main advantage of the proposed algorithm is faster than block ciphers, have lower hardware complexity and more suitable for streaming application such as (Satellite Image).

4.3 The complete encryption system

Three tested existed algorithms are integrated with the proposed one to produce the robust encryption technique. The chosen algorithms were AES, Shifting and RC4. The user can select this four algorithms to perform the robust technique. The complete system is embedded on Raspberry Pi to simulate the satellite subsystem and the Personal computer to simulate the ground station.

These Encryption algorithms were selected depends on the advantages were needed in the difficulty and complexity of complete encryption technique.

5. Experimental work

A complete program including graphic user interface is implemented to use the encryption technique and its options.

Different sets of satellite images were tested using the proposed system. For more than million trial, the number of cracks to the system were zero.

It is proposed that the encryption program will be on satellite subsystem, and the decryption one will be in the ground station.

6. Conclusion

This paper presented a novel and robust research study of integration and robust implementations of cryptographic algorithms for satellites on-board use. The increasing demand for on-board security measures are the motivation for this research study.

The encryption algorithm used on-board should be robust to harsh environment induced faults and providing high-speed encryption without consuming much power and processing resources. With these constraints a system using the latest encryption algorithms and a novel proposed one has been proposed in this paper.

To protect the valuable information generated by the valuable on-board payloads, the proposed encryption system has been identified as the suitable encryption model to perform the high data rate downlink.

Different kinds of standard strong encryption algorithms have been implemented. A new proposed encryption algorithm was presented. One of the major advantages of the proposed security system of the satellite images is its very

high complexity to crack. Using the proposed encryption system on different satellite images (Optical, SAR and IR) proved that the proposed system is a powerful security system for different satellite payloads.

The research objectives were achieved. A powerful algorithm for encrypting satellite images is implemented with its software interface and imbedded on hardware (Raspberry Pi).

References

[1] Alfred J. Menezes, Paul C. van Orschot and Scott A. Vanstone, *Hand book of Applied Cryptography*, Page # 1, CRC Press, 2009.

[2] William Stallings; Cryptography and Network Security: Principals and Practice, Prentice Hall international, Inc.; 2010.

[3] Douglas R. Stinson, *Cryptography Theory and Practice*, CRC Press, 1995.

[4] Dorothy Elizabeth Robling Denning, *Cryptography and Data Security*, Addison-Wesley Publishing Company, 2010.

[5] Fred Piper and Sean Murphy, Cryptography: A Very Short Introduction, Oxford University Press, 2002.

[6] Sashikala Channalli and Ajay Jadhav, "*Steganography An Art of Hiding Data*", International Journal on Computer Science and Engineering Vol.1(3), 2009, 137.

[7] BENCHIKH, Omar & Bentoutou, Y & Taleb, Nasreddine. (2016). "*Encryption of satellite images using the AES algorithm"*. International Conference of Computing for Engineering and Sciences (ICCES'2016), At Barcelone, (Espagne).

[8] Ferguson, Niels, Bruce Schneier, and Tadayoshi Kohno. "*Cryptography Engineering: Design Principles and Practical Applications"*. Indianapolis: Wiley, 2010.

[9] Buchmann, Johannes, and Jintai Ding. "*Post-quantum Cryptography"*: Second International Workshop, PQCrypto 2008, Cincinnati, OH, USA, October 17-19, 2008: Proceedings. Berlin: Springer, 2008.

[10] Dobbertin, Hans, Vincent Rijmen, and Aleksandra Sowa. "*Advanced Encryption Standard—AES"*: 4th International Conference, AES 2004, Bonn, Germany, May 10-12, 2004: Revised Selected and Invited Papers. Berlin: Springer, 2005.

[11] Paul, Goutam, and Subhamoy Maitra. "*RC4: Stream Cipher and Its Variants"*. Boca Raton, Fla: CRC, 2012.

[12] Rupinder Kaur, Dr. Madhu. "*Effective Symmetric Key Block Ciphers Technique for Data Security"*: RIJNDAEL GoelInternational Journal of Science, Engineering and Technology Research (IJSETR), Volume 3, Issue 7, July 2014

[13] Boukhatem Mohammed Belkaid el. "*Meteosat Images Encryption based on AES and RSA Algorithms Meteosat Image Encryption"*. (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 6, No. 6, 2015

14] Ms NehaKhatri – Valmik, Prof. V. K Kshirsagar. "*Blowfish Algorithm"*. IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 2, Ver. X (Mar-Apr. 2014), PP 80-83, India.