A Hybrid Cryptosystem of Image and Text Files

Using Blowfish and Diffie-Hellman Techniques

Bibin Jaimon
B4E028

Guided by:

Prof. Jobin Joy

CSE Department

Govt College of Engineering Kannur

February 13, 2018

Outline

1. Introduction
2. Background Knowledge
3. Proposed Algorithm
4. Results and Analysis
5. Conclusion
6. References

Introduction

- New algorithm of encrypting and decrypting images and text files.
- Combines the concepts of Diffie Hellman and Blowfish algorithm.
- First encrypt a file using a secret key generated by blowfish algorithm.
- Then using Diffie-Hellman protocol a shared private key is generated.

Background Knowledge I

Hybrid cryptosystem

- Combines the convenience of a public-key cryptosystem with the efficiency of a symmetric-key cryptosystem.
- Encryption integrity is maintained by confusion and diffusion.

Diffie Hellman protocol

- Two computer users generate a shared private key with which they can then exchange information across an insecure channel.

Background Knowledge II

Blowfish algorithm

- A symmetric block cipher that can be used as a drop-in replacement for DES or IDEA.
- Takes a variable-length key, from 32 bits to 448 bits.
- Ideal for both domestic and exportable use.

Symmetric key and Asymmetric key

- Symmetric algorithms: use the same key for both encryption and decryption
- Asymmetric algorithms: use different keys for encryption and decryption.

Related Work I

Image Encryption and Decryption Approach using Pixel Shuffling.[1]

- Encryption and Decryption of an image by pixel shuffling.
- Uses Arnold Cat Map and generate Pseudo-random number using Henon Map.
- XOR operation between the pixel value and the key value generated by the Henon Map.

Related Work II

Performance Analysis of a Proposed Symmetric Cryptography Algorithm

- Design algorithm to merge both RSA algorithm and Diffie-Hellman Algorithm.
- Algorithm is M*N times complex to break using even the latest version of Brute Force attack.
- M and N are corresponding complexities imposed by the Diffie-Hellman and RSA algorithms.

Proposed Algorithm I

1) Both the users agree upon a prime p and another number g that has no factor in common.
2) User 1 takes a private key x and calculates a key R1= (g^x) mod p.
3) User 1 generates a secret key and cipher and then encrypts the file using the secret key and cipher generated by blowfish algorithm.

```
[
keyGenerator = KeyGenerator.getInstance(”Blowfish”);
secretKey = keyGenerator.generateKey();
]
cipher = Cipher.getInstance(”Blowfish”);
```

Proposed Algorithm II

4) User 2 takes a private key y and calculates a key R2= (g^y) mod p.

5) Both users share R1 and R2 with each other through the insecure channel. So these values become public.

6) User 1 calculates final key k1= $(R2^x)$ mod p. User 2 calculates k2=$(R1^y)$ mod p.

7) If the values of k1 and k2 match then only user 2 gets the permission of decryption. So user 1 sends the secret key for Blowfish to user 2. Then user 1 sends the encrypted file to user 2.

8) User 2 decrypts the file using Blowfish algorithm.

Block Diagram I

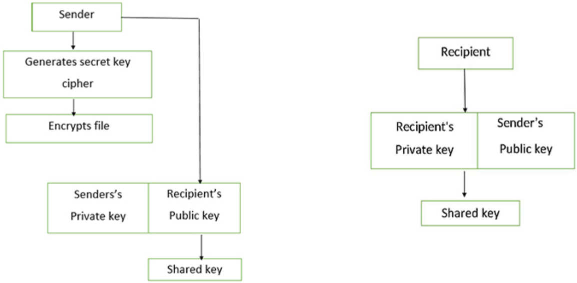


Figure: Encryption process and shared key generation

Block Diagram II

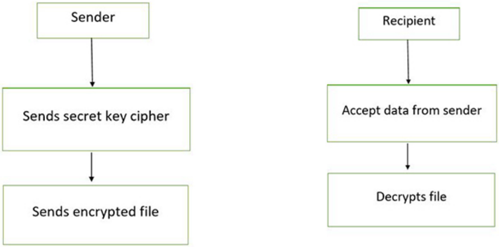
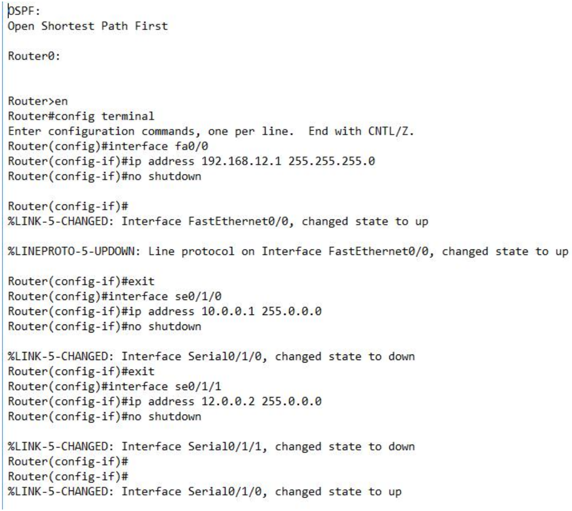


Figure: When shared keys matches

How are Attacks Intercepted

- File is sent through insecure channel.
- Attacker intercepts the file.
- The attacker won't be able to see the content as the text/image.
- The contents remain encrypted because the attacker don't have the secret key of blowfish encryption.

Results and Analysis I



```
OSPF:
Open Shortest Path First

Router0:


Router>en
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fa0/0
Router(config-if)#ip address 192.168.12.1 255.255.255.0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

Router(config-if)#exit
Router(config)#interface se0/1/0
Router(config-if)#ip address 10.0.0.1 255.0.0.0
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/1/0, changed state to down
Router(config-if)#exit
Router(config)#interface se0/1/1
Router(config-if)#ip address 12.0.0.2 255.0.0.0
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/1/1, changed state to down
Router(config-if)#
Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/1/0, changed state to up
```

Figure: A simple text file with extension '.txt' to be encrypted

Results and Analysis II

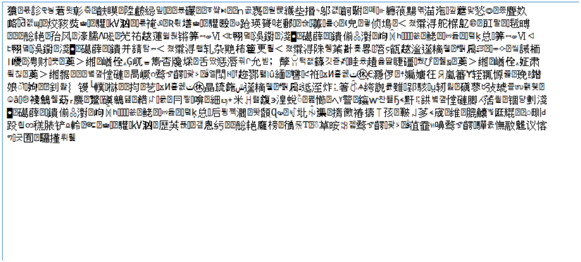


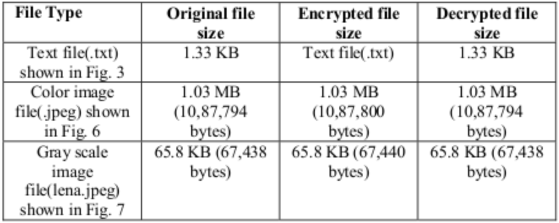
Figure: The encrypted file

- Fig.4. shows the cipher text that is obtained as a result of encryption of the text file shown in Fig. 3.
- When decrypted produces exactly same as the original file.

Results and Analysis III

- In the image files, the content is numeric values in the domain [0, 255].
- The numeric content of encrypted files go out of range and violate the specification for image file format.
- The display is not supported by any digital computers.
- Encrypted file is not readable but the file gets properly decrypted.

Results and Analysis IV



File Type	Original file size	Encrypted file size	Decrypted file size
Text file(.txt) shown in Fig. 3	1.33 KB	Text file(.txt)	1.33 KB
Color image file(.jpeg) shown in Fig. 6	1.03 MB (10,87,794 bytes)	1.03 MB (10,87,800 bytes)	1.03 MB (10,87,794 bytes)
Gray scale image file(lena.jpeg) shown in Fig. 7	65.8 KB (67,438 bytes)	65.8 KB (67,440 bytes)	65.8 KB (67,438 bytes)

Figure: Memory Requirement

Table 1 gives the amount of memory needed to store the different files on which the algorithm is performed.

Conclusion

- Takes the advantage of generating a variable length key using the Blowfish algorithm.
- The file is decrypted only if the key matches.
- Overcomes most of the shortcomings faced by existing algorithms.

References I

[1] A. K. Prusty, A. Pattanaik, and S. Mishra, "An image encryption & decryption approach based on pixel shuffling using arnold cat map & henon map," *2013 International Conference on Advanced Computing and Communication Systems*, Oct 2014.

[2] T. K. Hazra, A. Mahato, A. Mandal, and A. K. Chakraborty, "A hybrid cryptosystem of image and text files using blowfish and diffie-hellman techniques," *2017 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON)*, Oct 2017.

[3] M. Mukhedkar, P. Powar, and P. Gaikwad, "Secure non real time image encryption algorithm development using cryptography & steganography, applications, and challenges," *2015 Annual IEEE India Conference (INDICON)*, March 2016.

References II

[4] S. Hassene and M. N. Eddine, “A new hybrid encryption technique permuting text and image based on hyperchaotic system,” *2016 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, July 2016.

QUESTIONS?

Thank You