"Xboard: A Secure Keyboard"

Tanmay Joshi
Dept Computer Science and Engineering
MIT ADT University Pune Maharashtra
ktanmayjoshi@gmail.com

Himanshu Gupta
Dept Computer Science and Engineering
MIT ADT University Pune Maharashtra
himanshunotesmit@gmail.com

kiran Bidua
Dept Computer Science and Engineering
MIT ADT University Pune Maharashtra
kiran.bidua@mituniversity.edu.in

Rohan Bhange
Dept Computer Science and Engineering
MIT ADT University Pune Maharashtra
bhangerohan24@gmail.com

Avishi Bafna
Dept Computer Science and Engineering
MIT ADT University Pune Maharashtra
avishibafna@gmail.com

Abstract— X Board is designed to meet today's need for a light-weight, Internet-scale data protection solution for mobile devices. It also meets today's need for a light-weight, Internet-scale data protection solution for mobile devices.

The X Board Keyboard locally encrypts your text and injects it into any app on your phone. When you receive a message you just need to tap to decrypt!

The messages sent or received don't make any sense visually as long as they are decrypted, hence it can easily secure you from some of the androids vulnerabilities which attackers use to exploit.

The messages are end to end encrypted with AES 256 cryptographic algorithm.

Xboard is a secure keyboard with robust features as follows:

1. Flexible: Works in all of your favorite messaging apps like WhatsApp, Signal, Messenger, etc.
2. Cryptography: Secure all your messages using AES(Advanced Encryption Standard) algorithm.
3. Security: You can rely on this app with its end to end encryption feature.
4. Usability: Simple to use, user friendly UI design for all age groups.
5. One tap Encryption: Encrypt or Decrypt all your messages that you "Send" or "Receive" in a single tap of a button.
6. CIA Triad: Follows the CIA Triad of confidentiality, integrity and availability which is considered as the core underpinning of information security. Our project focuses on creating a secure keyboard which deals with the issues of privacy, data breach.

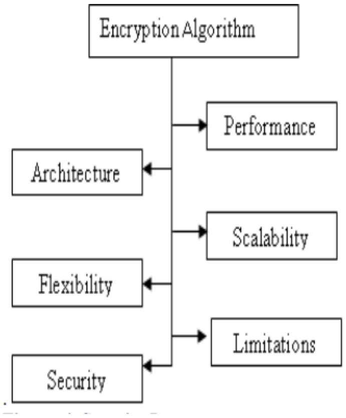
Keywords—Encryption, Decryption, AES Algorithm, Symmetric/Asymmetric Method, End to End Encryption, CIA Triad.

I. INTRODUCTION

X Board is designed to meet today's need for a light-weight, Internet-scale data protection solution for mobile devices. The X Board Keyboard locally encrypts your text and injects it into any app on your phone. When you receive a message you just need to tap to decrypt! The messages sent or received don't make any sense visually as long as they are decrypted , hence it can easily secure you from some of the androids vulnerabilities which attackers use to exploit. The messages are end to end encrypted with AES 256 cryptographic algorithm.

1. SECURITY STANDARDS

Several symmetric algorithms exist but there are some parameters to evaluate them. To select appropriate algorithm for a particular application we need to know its strength and limitation. There are some parameters need to consider these parameter include Architecture, Performance, Flexibility Security, Scalability and Limitations. Below figure shows the parameters for encryption Algorithm:



The security parameters are as follows:

1. Architecture:-Architecture includes structure and mathematical operations that an algorithm performs for encryption and decryption. Characteristics and how they are implemented. It also includes key used in the algorithm (secret key or public key) for encryption and decryption.

2. Performance:-Performance includes time required for Encryption and decryption, Memory required Software hardware performance and computational cost.

3. Security:-Security measures strength of the algorithm form attacks. It includes required element, possesses and property of algorithm. Security of an encryption algorithm depends on the key size used to execute the encryption. Length of key is measured in bits .

4. Flexibility:-Flexibility defines whether the algorithm allows some modification or not. Some time we need to modify the algorithm because of the requirements.

5. Scalability:-We test the algorithm for different size of the file or data. So scalability is one of the important elements for algorithms. Scalability depends on certain parameters such as Memory Usage, Encryption rate, Software hardware performance; Computational efficiency.

6. Limitations:-Each and every algorithm has some drawback for an encryption algorithm we already known attacks or weakness of the algorithm. Limitation defines how fine the algorithm works for the resources available to it, how often is vulnerable to different types of attacks.

Litreture Review

Several papers have been reviewed and observed certain aspects to implement the effective approach for encryption and decryption algorithm for security.

In 2010 Ayushi proposed "A Symmetric Key Cryptographic Algorithm ". There are two basic types of cryptography Symmetric Key and Asymmetric Key. Symmetric key algorithms are the quickest and most commonly used type of encryption. Here, a single key is used for both encryption and decryption. There are few well-known symmetric key algorithms i.e. DES, RC2, RC4, IDEA etc. She represents various symmetric key algorithms in detail and then proposes a new symmetric key algorithm.

In 2012 Suyash Verma et al proposed "An Efficient Developed New Symmetric Key Cryptography Algorithm for Information Security". They proposed new encryption algorithm and used block cipher generating mechanism. They proposed evaluation, results by calculation with different plaintexts in the same key (DPSK) mode. By the results they show that, under the same key size and for the same size of the data, proposed algorithm work faster than existing algorithm.

In 2013 Prerna Mahajanet al proposed "A Study of Encryption Algorithms AES, DES and RSA for Security". They implemented three encryption techniques like AES, DES and RSA algorithms and compared their performance of other encryption techniques based on time for encryption and decryption. They also show results of analyses of effectiveness of each algorithm. Based on the text files used and the experimental result.

In 2014 Anjula Gupta et al. proposed "Cryptography Algorithms: A Review". They proposed a study of existing encryption techniques are analyzed to promote the performance of the encryption methods. To sum up, all techniques they used unique ID. They surveyed many papers; found that throughput value of BLOWFISH is greater than all symmetric algorithms. Power Consumption value of BLOWFISH is least.

In 2014 Reema Gupta et al proposed "Efficient Encryption Techniques in Cryptography Better Security Enhancement". They proposed a study of Encryption techniques and discussed with their limitations and procedure .Huffman coding and B2G, G2B is used for encryption. They also discussed various transpositional techniques like Simple columnar, simple row, Route cipher, transposition.

In 2015 Abhishek Joshi et al proposed "An Efficient Cryptographic Scheme for Text Message Protection against Brute Force and Cryptanalytic Attacks" They proposed an efficient cryptographic scheme for text message Protection against Brute force and Cryptanalytic attacks. They show that this technique can also be used for most crucial applications where it requires a significant security of transmitted message and also there is no overhead on the transfer of message and the key when it is used with our proposed technique.

In 2015 Ashraf Odeh et al "A Performance Evaluation of Common Encryption Techniques with Secure Watermark System (SWS)". They demonstrate a fair comparison between the most common algorithms and with a novel method called Secured Watermark System (SWS) in data encryption field according to CPU time, packet size and power consumption. They provides a comparison the most known algorithms used in encryption: AES (Rijndael), DES, Blowfish, and Secured Watermark System (SWS). They apply the same methodology on images and audio data.

In 2016 Sushil Kumar Tripathi "An Efficient Block Cipher Encryption Technique Based on Cubical Method and Improved Key". They presented an efficient block cipher encryption techniques based on improved key. Proposed EES method is based on block level symmetric encryption. The proposed EES method is based on improve cubes. They used a pair of binary inputs are contains by each cell. The Cube can able to provide a various number of combinations. The proposed EES algorithm, performed a series of bit transformations, by using of S-BOX, operation XOR, and operation AND.

SYMMETRIC APPROACH

Symmetric technique has emphasized on improving conventional method of encryption by using substitution cipher. Substitution techniques have used alphabet for cipher text. A single key is used for both encryption and decryption. There are few well-known symmetric key algorithms i.e. DES, AES,RC2, RC4, IDEA etc.

AES (Advanced Encryption Standard)

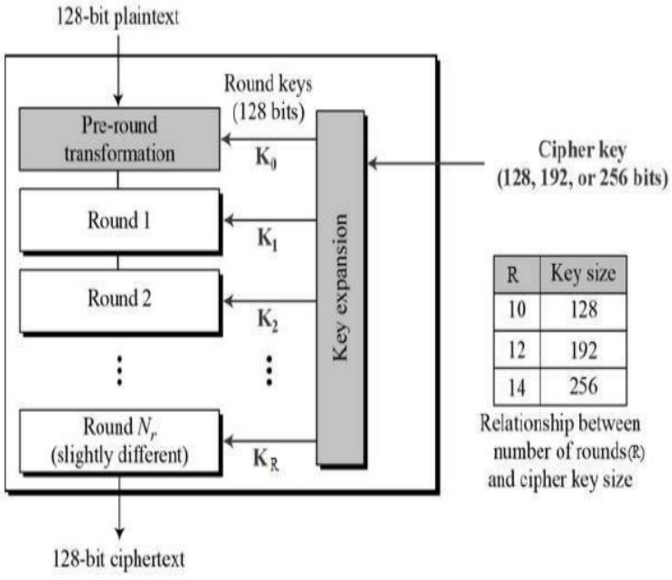
The more popular and widely adopted symmetric encryption algorithm likely to be encountered nowadays is the Advanced Encryption Standard (AES). It is found at least six time faster than triple DES. A replacement for DES was needed as its key size was too small. With increasing computing power, it was considered vulnerable against exhaustive key search attack. Triple DES was designed to overcome this drawback but it was found slow. The features of AES are as follows –

- Symmetric key symmetric block cipher
- 128-bit data, 128/192/256-bit keys
- Stronger and faster than Triple-DES
- Provide full specification and design details
- Software implementable in C and Java

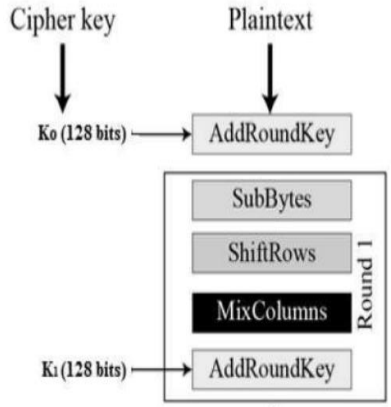
Operation of AES

AES is an iterative rather than Feistel cipher. It is based on 'substitution–permutation network'. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations). Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix − Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

The schematic of AES structure is given in the following illustration-:



Encryption Process: Here, we restrict to description of a typical round of AES encryption. Each round comprise of four sub-processes. The first round process is depicted below –



Byte Substitution (SubBytes): The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.

Shiftrows: Each of the four rows of the matrix is shifted to the left. Any entries that 'fall off' are re-inserted on the right side of row. Shift is carried out as follows –

- First row is not shifted.
- Second row is shifted one (byte) position to the left.
- Third row is shifted two positions to the left.
- Fourth row is shifted three positions to the left.
- The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other.

MixColumns: Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round.

Addroundkey: The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

Decryption Process: The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order –

- Add round key
- Mix columns
- Shift rows
- Byte substitution

Since sub-processes in each round are in reverse manner, unlike for a Feistel Cipher, the encryption and decryption algorithms needs to be separately implemented, although they are very closely related.

AES Analysis: In present day cryptography, AES is widely adopted and supported in both hardware and software. Till date, no practical cryptanalytic attacks against AES has been discovered. Additionally, AES has built-in flexibility of key length, which allows a degree of 'future-proofing' against progress in the ability to perform exhaustive key searches. However, just as for DES, the AES security is assured only if it is correctly implemented and good key management is employed.

VAULT : This is special encrypted vault which can hold your images and videos in encrypted form, its perfectly designed vault inside a keyboard which will help you to keep important images and videos safe at your fingertips. It is has the security of Bio-metrics Lock (fingerprint), the vault uses the same encryption technique the AES Encryption Algorithm it coverts the images into the bytes which helps the AES the process more smoothly.

PASSWORD MANAGER: The Password manager provides easy access to the passwords that are being saved in this manager. The manager does have a large no of accounts that you can choose from the drop down provided when clicked on the add button. Here you can, edit or delete passwords. The manager uses AES Algorithm to make sure your personal and

CONCLUSION

'Xboard' is a robust android application . The key concept is to secure one's integrity through a single android keyboard application using a secure cryptography algorithm. The user with minimum knowledge about phones can be able to operate the application easily. We learn a lot of new things about the cryptography algorithms , Java and Application Development, and were able to successfully implement the project in limited time using team and time management.

REFERENCES

1) A. J. Aviv, K. Gibson, E. Mossop, M. Blaze, and J. M. Smith, “Smudge attacks on smartphone touch screens,” in Proceedings of the Workshop on Offensive Technology WOOT, 2010.

2) K. Mowery, S. Meiklejohn, and S. Savage, “Heat of the moment: characterizing the efficacy of thermal camera-based attacks,” in Proceedings of the Workshop on Offensive Technologies (WOOT), 2011.

3) Y. Zhang, P. Xia, J. Luo, Z. Ling, B. Liu, and X. Fu, “Fingerprint attack against touch-enabled devices,” in Proceedings of the 2nd ACM Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM '12), pp. 57–68, Raleigh, NC, USA, October 2012.

4) Y. Abdelrahman, M. Khamis, S. Schneegass, and F. Alt, “Stay cool! understanding thermal attacks on mobile-based user authentication,” in Proceedings of 35th Annual CHI Conference on Human Factors in Computing Systems (CHI), pp. 3751–3763, Denver, CO, USA, May 2017.

5) M. Backes, M. Duermuth, and D. Unruh, “Compromising reflections - or - how to read lcd monitors around the corner,” in Proceedings of the 29th IEEE Symposium on Security and Privacy (S&P), 2008.

6) M. Backes, T. Chen, M. D1rmuth, H. P. A. Lensch, and M. Welk, “Tempest in a teapot: Compromising reflections revisited,” in Proceedings of the 30th IEEE Symposium on Security and Privacy (S&P), 2009.

7) D. Balzarotti, M. Cova, and G. Vigna, “Clearshot: eavesdropping on keyboard input from video,” in Proceedings of the 29th IEEE Symposium on Security and Privacy (S&P), 2008.

8) F. Maggi, A. Volpatto, S. Gasparini, G. Boracchi, and S. Zanero, “A fast eavesdropping attack against touchscreens,” in Proceedings of the 2011 7th International Conference on Information Assurance and Security, IAS 2011, pp. 320–325, Melaka, Malaysia, December 2011.

9) Q. Yue, Z. Ling, X. Fu, B. Liu, W. Yu, and W. Zhao, “My google glass sees your passwords!,” in Proceedings of the Black Hat USA, 2014.

10) Q. Yue, Z. Ling, X. Fu, B. Liu, K. Ren, and W. Zhao, “Blind recognition of touched keys on mobile devices,” in Proceedings of the 21st ACM Conference on Computer and Communications Security, CCS 2014, pp. 1403–1414, November 2014.

11) J. Sun, X. Jin, Y. Chen, J. Zhang, R. Zhang, and Y. Zhang, “Visible: Video-assisted keystroke inference from tablet backside motion,” in Proceedings of the 23rd ISOC Network and Distributed System Security Symposium (NDSS), 2016.

12) L. Zhang, Z. Cai, and X. Wang, “FakeMask: A Novel Privacy Preserving Approach for Smartphones,” IEEE Transactions on Network and Service Management, vol. 13, no. 2, pp. 335–348, 2016.

13) Z. Cai, Z. He, X. Guan, and Y. Li, “Collective data-sanitization for preventing sensitive information inference attacks in social networks,” IEEE Transactions on Dependable and Secure Computing, 2016.

14) L. Cai and H. Chen, “TouchLogger: Inferring keystrokes on touch screen from smartphone motion,” in Proceedings of the 6th USENIX Workshop on Hot Topics in Security (HotSec), 2011.

15) Z. Xu, K. Bai, and S. Zhu, “TapLogger: inferring user inputs on smartphone touchscreens using on-board motion sensors,” in Proceedings of the 5th ACM Conference on Security and Privacy in Wireless and Mobile Networks, pp. 113–124, Tucson, Ariz, USA, April 2012.

16) E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang, “ACCessory: password inference using accelerometers on smartphones,” in Proceedings of the Proceeding of the 13th Workshop on Mobile Computing Systems and Applications (HotMobile '12), no. 9, New York, NY, USA, February 2012.

17) E. Miluzzo, A. Varshavsky, S. Balakrishnan, and R. R. Choudhury, “Tapprints: your finger taps have fingerprints,” in Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services (MobiSys '12), pp. 323–336, Ambleside, UK, June 2012.

18) A. J. Aviv, B. Sapp, M. Blaze, and J. M. Smith, “Practicality of accelerometer side channels on smartphones,” in Proceedings of the 28th Annual

Computer Security Applications Conference (ACSAC '12), pp. 41–50, ACM, Orlando, Fla, USA, December 2012.

19) L. Simon and R. Anderson, “Pin skimmer: Inferring pins through the camera and microphone,” in Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM), 2013.

20) H.-S. Shin, “Device and method for inputting password using random keypad,” United States Patent No. 7,698,563, 2010.