MIT School of Engineering
Department of Computer Science and Engineering

Mini Project Synopsis

Group ID: 05
Project Title: XBoard: An Secure Keyboard.
Group Members:

Enrolment Number	Roll No.	Name of student	Email Id	Contact Number
MITU20BTCSD035	2203514	TANMAY JOSHI	ktanmay@gmail.com	9082818346
MITU19BTML0023	2196068	ROHAN BHANGE	bhangerohan24@gmail.com	7666465319

Problem statement.

In today’s era everybody is connected with each other through social media apps like Whatsapp, Instagram, etc. The number of smartphone users are increasing exponentially throughout the world, and so the cases of cyber attacks.

The softboards currently available in the market either collect a lot of data of what user is typing.

Even the apps like whatsapp, messenger are end to end encrypted but what about the chat logs which can be visually analysed.

There are many vulnerabilities in android for eg. the features such as accessibility service a feature provided for disabled people can be used to read the sensitive information from the target’s screen.

Hence there is a need of the hour for such application which protects the user from this vulnerabilities, in today's world full of cyber threats to protect one’s integrity.

Abstract.

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.

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.

X Board is designed to meet today’s need for a light-weight, Internet-scale data protection solution for mobile devices.

Beyond its appearance and flexible features it will be equipped with the security feature unlike no other keyboards promises in the market.

Literature survey.

The paper’s present a full-scale usability testing of a generic Android privacy enhancing keyboard (PEK), which can prevent various attacks against touch-enabled devices from inferring user pins or passwords.

They perform an iterative two-round two-stage usability test including pilot usability tests and main usability tests for improving PEK for broad adoption. Based on the findings of the two usability tests in the first usability test, they implement new features in the current PEK.

After the iterative improvement efforts, most users find our app easy to use and install. However, the usability test demonstrates the worrisome phenomena that many users blindly trust their phones for security or are not much concerned with the possible breaches.

These phenomena demonstrate the human factor that contributes to the vulnerabilities of the cyberspace.

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

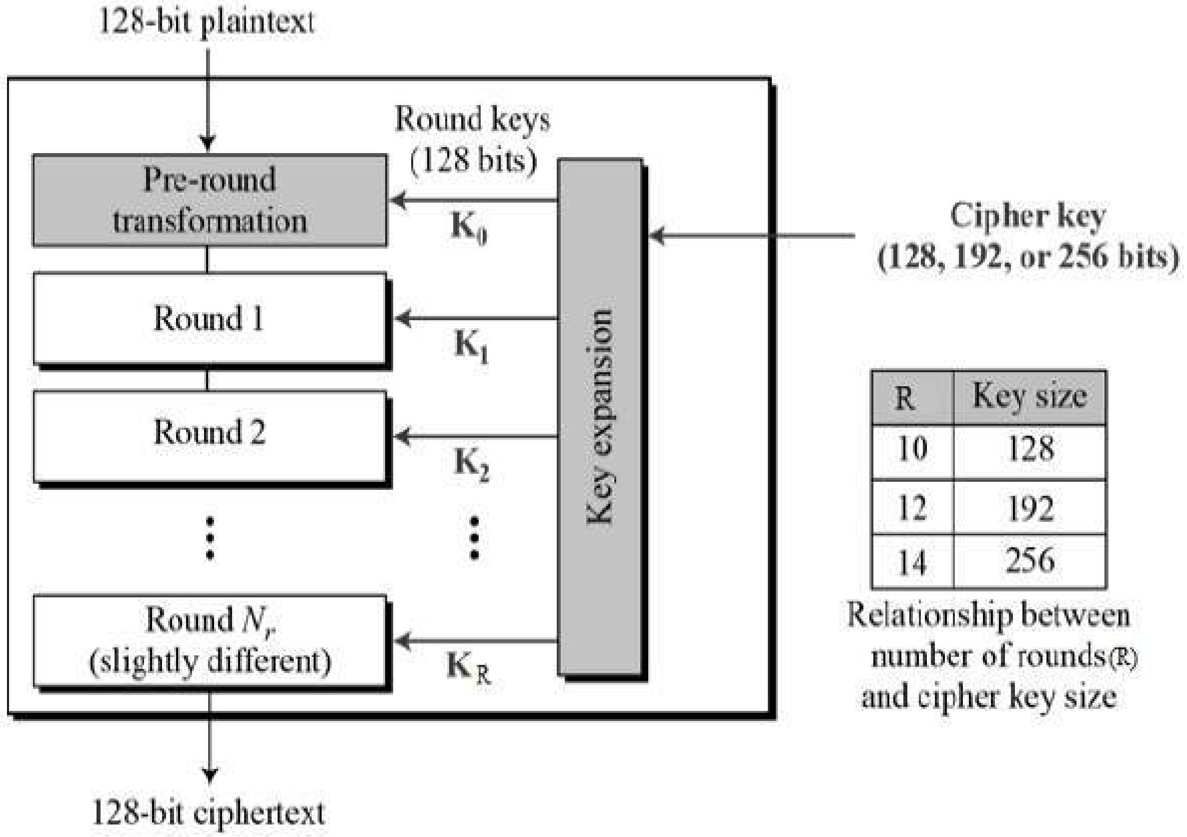
Operation of AES

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 −

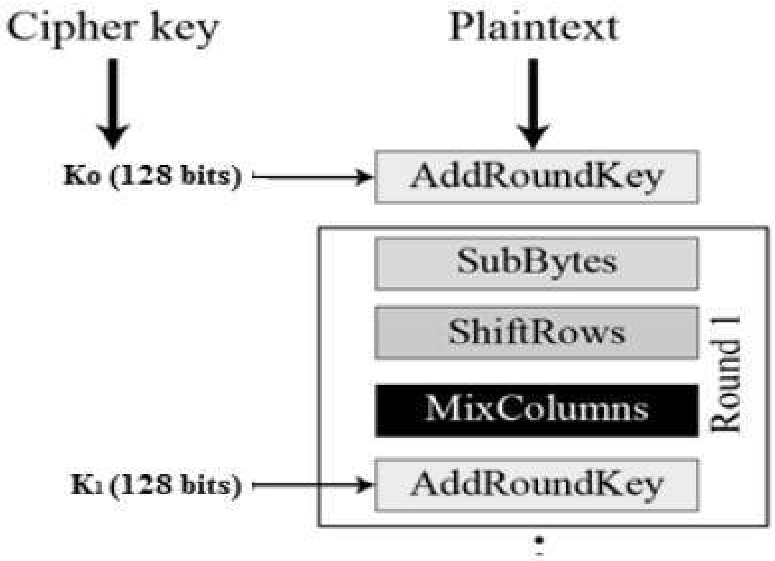


R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

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.

Conclusion.

‘Xboard’ is an robust android application .

The key concept is to secure one’s integrity through a single android keyboard application using secure cryptographic algorithm.

The user with minimum knowledge about phones can be able to operate the application easily.

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.