ISSN: 2456-1363
International Journal of Scientific Research & Growth
A Multidisciplinary Peer Reviewed and Refereed International Journal
Volume-3 Issue-2 June- 2018
www.ijsrg.com
editor@ijsrg.com

Multi Layer Data Security Through Data Obscuring

[1]Harleen Virdi, [2]Manish Kumar Mukhija
[1]M. Tech Scholar, Modern Institute of Technology & Research Centre, Alwar, Rajasthan, India
[2]Assistant Prof., Modern Institute of Technology & Research Centre, Alwar, Rajasthan, India
Email- harleenvirdi.12@gmail.com, mukhijakumar@gmail.com

Abstract
Data is the most significant entity and basic unit of communication in all form whether it is in electronic form, digital form, verbal form or written form. In today's world of advance communication, everybody prefers to save and retrieve data quickly from any geographic coordinates. To facilitate this, mobile cloud computing and smart phones play a significant role. They supply user to use the techniques for economical storage and retrieval of data using platform, software system and infrastructure being provided by third party. Hence in this work we will be working over three layers data security hence providing shelter to the data by obscuring the main content. This will be done by encrypting the data at sender, receiver and also at the cloud where image will be saved for transferring over the internet in encrypted form , hence providing security at multi layer. We shall be discussing about multi layer security mechanism for the image contents in mobile devices.
Keywords: *- Multi layer data security, Data obscuring, mobile cloud computing.*

1- INTRODUCTION

I. What is encryption?

In cryptography, encryption is the path toward encoding messages or information with the end goal that exclusive approved gatherings can read it. Encryption does not of itself keep away from catch endeavor but instead denies the message substance to the interceptor. In an encryption plot, the proposed correspondence information or message, referred to as plaintext, is encoded using an encryption count, delivering figure message that must be perused if decoded. For particular reasons, an encryption plot, for the most part, uses a pseudo- irregular encryption key made by estimation. It is on an basic level possible to decipher the message without having the key, be that as it may, for an overall plot encryption framework, broad computational resources and capacity are required. An approved recipient can without much of a stretch decode the message with the key gave by the originator to recipients, however not to unapproved.

II. Why we need encryption?

While transferring the data through internet from one point of contact to another or it is in motion, it is moved from one point to another. Data need to be secured in each manner, whether it is being transferred from one network to another network or from one local device to another one, from one cloud service to another, for all these effectives measures are required to be followed for data protection. While data is in transit, it has high risk of being exposed and need protection. For this, there are different approaches, like we can either protect data by encrypting it or by encrypting the connections itself which also plays a major role in protecting data and nowadays a popular tool for the same. Along with encryption, there are many other methods through which data can be protected while in transit

III Encryption Methods for Mobile

As it is said that privacy need to exist along with security. And is one of the ways to provide this security to our mobile app to encryption. Types of encryption methods include:

Symmetric Key: In this of encryption there is a common key that is shared by sender and receiver and is used for the message encryption and decryption.

Asymmetric Key In this type of encryption, there is a pair of unique keys, where one key to used to encrypt the data at sender side and another is used to decrypt the data at receiver side. This is also known as public key encryption as in this one key is kept public while other on is kept private.

Cryptographic Hash Function In this type of encryption technique, a hash value or a checksum or a message digest is produced for a specific data object**.** These hash functions are applied in information security to check the data integrity, authentic user control, and other mechanisms of security. These cryptographic hash functions generate a checksum value for each data object and that can be changed when the data is changed whether intentionally or unintentionally. By comparing previous and current checksum of data object, its integrity can be evaluated.

Digital Signature It is different from digital signature. In this a message, software or digital document's integrity and authenticity is validated using mathematical technique.

File Encryption Software This is a program which uses algorithms and adjoining to protect the contents of computer files.

2- SECURITY IN MOBILE DEVICES

Mobile data security is defined as an effort to secure data on mobile devices such as tablets and smart phones. Generally, mobile security is something which enterprises work on to safeguard sensitive information whose security can be compromised because of its use on several mobile devices. Mobile data security becomes more and more important each day as mobile devices have gone from communication tools to cameras for still and video photography and mini computers for emailing, searching, maps etc.

Mobile Data Security becomes more and more important each day as mobile devices have gone from communication tools to cameras for still and video photography and mini computers for emailing, searching, maps etc.

Mobile devices which transmit enterprise data & access corporate networks pose a risk in terms of keeping sensitive information secure. It is necessary to prevent corporate data from getting exposed through users' devices. There are several tools available including encryption, remote wipe, (MDM) mobile device management applications and more

3- CHALLENGES IN SECURITY IN MOBILES

Mobile devices which transmit enterprise data & access corporate networks pose a risk in terms of keeping sensitive information secure. It is necessary to prevent corporate data from getting exposed through users' devices. There are several tools available including encryption, remote wipe, (mdm) mobile device management applications and more.

Data is now more portable than ever. On the one hand the mobile nature of data is opening up opportunities, but on the other hand it is challenging businesses to secure personnel and corporate data.

Mobile data security is defined as an effort to secure data on mobile devices such as tablets and smart phones. Generally, mobile security is something which enterprises work on to safeguard sensitive information whose security can be compromised because of its use on several mobile devices.

4- PROBLEM WITH CURRENT SCENARIO

These days we run over the information infringement. The consistent pace of breaks strengthens the requirement for encryption as a last line of protection. As of late

notwithstanding, one of the most established and best security strategies has been to a great extent consigned to a bit of hindsight in the present new cloud and huge information environments. Data encryption can be made as simple as we need to make or as muddled as we need to. Primary prerequisite is the information compose that is should have been encoded, where it lives and who ought to have the entrance to it. There are different instruments for encryption in the market that are effortlessly accessible, effectively usable and furthermore moderate. In the event that we have a critical application to be scrambled, at that point we need a straightforward information encryption that lies under the application layer without changing our framework working framework or information or capacity or application.

5- RELATED WORK/LITERATURE SURVEY

Vichare et al (2017) proposed an Android Application which will give the highlights of programmed encryption, data encryption/decoding, and cloud backup across the board single mobile application. AES - 256 bit algorithm is utilized for this reason. The sole motivation behind utilizing AES for encryption/ decryption is that it is secure and has high computational complexity. Because of this, the interloper will require similarly more time to decode the information. Portable file system encryption engine that uses NIST certified cryptographic algorithms for Android mobile device offer a comparative performance analysis of our encryption engine under different operating conditions and for different loads including files and database (DB) operations.

Wang et al (2012) suggested that the information must be stored in an encrypted format utilizing cryptography on biometric for the security reasons. The convention is visually impaired as in it uncovers just the personality and no extra data about the client or the biometric to the verifying server or the other way around. As the protocol is based on asymmetric encryption of the biometric data, it captures the advantages of biometric authentication as well as the security of public key cryptography. The client at first selects with the biometric framework which is given by a cloud, once the person is enlisted his/her biometric authentication details are stored in cloud service provider database. The authorization details are also entered at the registration time which is also encrypted.

Teufl et al (2014) referred that deploying Android in security-critical environments is a complex task, as classified information may get compromised while being accessed, processed, and stored by insecure mobile devices. To encourage this task, we have systematically analyzed and assessed different encryption systems of the Android platform, which give the chance to ensure security basic and classified information.

Saraf et al (2014) proposed that fruitful usage of text and image encryption and in addition decryption is possible. Because of the blend of C code, Code Composer Studio, and DSP processor a one of a kind answer for text encryption

Savithri et al (2014) stated that application improvement for transmitting and accepting secret image through encryption and steganography using chaos in android platform is suggested. Key affectability is high for this calculation as unintended receiver won't know about the sort of chaotic map used and beginning conditions considered. MSE and PSNR are satisfactory for ordinary communication. The application created can be effortlessly utilized as different windows help in choosing required objects and achieving secret communication

Kumbharkhane et al(2015) proposed that the objectives of the Android security program portrays the basics of the Android security design and answers the most pertinent questions for system architects and security analysts. And furthermore centres around the

security highlights of Android's core platform and does not talk about security issues that are unique to particular applications.

Tayde et al (2015) It is indicated that AES encryption and decryption algorithm run faster in android phone. It gives better security of portable from unauthorized access. This application ensures the secure end to end exchange of information without any corrupt data. In future, the work might be reached out by video encryption and building up a more grounded encryption calculation with rapid and less memory usage

6- PROPOSED SYSTEM

The aim of this research is to provide the multi layered security while transferring image from one peer to another, this will be done by providing image data security data at sender side, receiver side as well as at the cloud. Similar solutions were provided in earlier times but they were at two levels, but we are trying to provide solution at three levels.

In our proposed system, the android application 'Snug' will be started with the login page. The user need to login into the app for further usage of the application. New user need to register into the application by entering name, password and mobile number. These login credentials are saved onto the cloud database. After successful registration into the app the user can now login into the app by entering user name and password. After the entered user name and password are verified from the cloud, the user can now use the app. After using the app the user can logout from the application. After the user has successfully login into the application, he/she can now send and receive the image capture through the app to the intended user by entering receiver's user name. This is shown in below figure as 1 level data flow diagram.

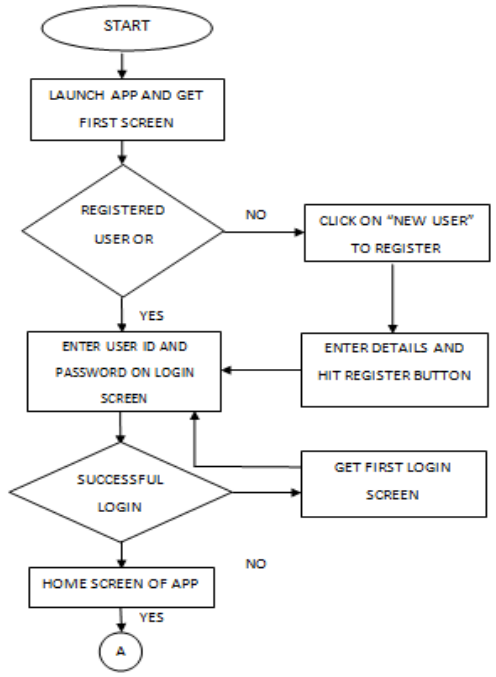

FIGURE1- BASIC FLOWCHART OF PROPOSED WORK

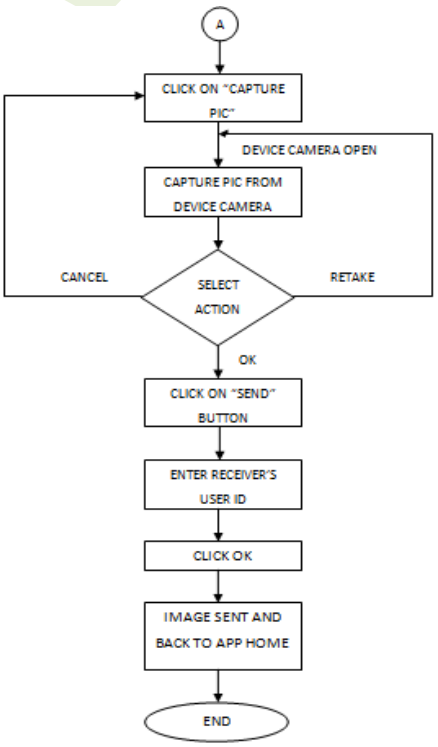

FIGURE2- FLOWCHART OF SENDING IMAGE

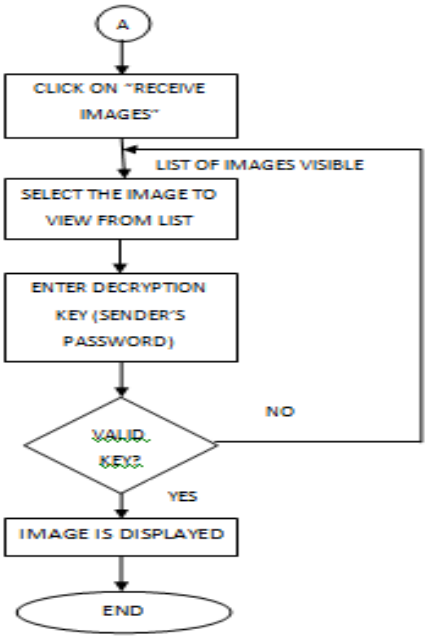

FIGURE 2- FLOWCHART OF RECEIVING IMAGE

6.1- PSEUDO CODE/STEP-WISE ALGORITHM

Step 1: The application's welcome screen will consist of a user ID and password.

Step 2: It would be installable on multiple handsets which would enable users across the globe to share encrypted contents wirelessly and independent of their geographical coordinates through HTTP.

Step 3: The user ID and password of the users shall be generated through a registration or sign – up module wherein the user ID would be generated dynamically by the application and password is what the user desires.

Step 4: The sign – up details of every user who registers via this application shall be stored on the cloud database. The cloud database service that we propose to use is from Backendless.com.

Step 5: The cloud database will have a separate user ID and password which we assume to be of administrator account.

Step 6: Once logged in, the application would enable users to send and receive content securely.

Step 7: In the application's settings, there would be an option called "Encrypted Password" in which the user can type in any password of length minimum 6 characters. Once the password is inputted, it will automatically get encrypted using AES algorithm.

Step 8: Once the user sends content through the application, it will get encrypted using AES algorithm along with the merged encrypted password.

Step 9: The sender will be able to send the encrypted content via the application to the receiver. Our application will enable users to send the content to the receiver on the basis of his / her user id using cloud database service.

Step 10: Once the receiver receives the encrypted content, in order to open it, he/she will have to input the password that the sender has set in it. Only upon entry of the correct password will the receiver be able to view the content.

Step 11: As the receiver can receive multiple contents from multiple users across the globe, there would be an option called "Received Contents" in the application which would enable the users to see records containing Timestamp, User ID, Name and Encrypted Content. This feature will allow the users to open any content that they have received so far from other users.

Step 12: The contents which are stored on the cloud are also not viewable by the administrator as they shall be encoded into numeric.

7- EXPERIMENTS AND RESULTS

I- Android with gradle

Gradle is an advanced general purpose build management system based on Groovy and Kotlin. Gradle supports the automatic download and arrangement of conditions or different libraries. It supports Maven and Ivy archives for recovering these conditions. This permits reusing the artifacts of existing form frameworks. Gradle supports multi- project and multi- artifact builds.

II Performance testing

An application is considered to have poor execution in the event that it reacts gradually, indicates uneven activities, stops, crashes, or consumes a lot of power. To stay away from these execution issues, we have to distinguish where our application is making inefficient utilization of resources, for example, the CPU, memory, graphics, network, and device battery. Which helps us to pick up a true understanding of how the execution of the Android application on a given device as the end clients would see them. For our project Snug, we will be getting statistics for the following parameters:

- CPU utilization
- Memory utilization
- Network utilization

As this app is for single user usage, hence we are not considering the stress testing or load testing. But the overall performance of an app is important, hence we are collecting the statistics for how the memory and CPU are behaving while the app is executing and performing. Also, as we have used the "Backendless" cloud services, hence, the network utilization is also an important parameter.

FIGURE 3- CPU, MEMORY AND NETWORK PROFILING OF SNUG

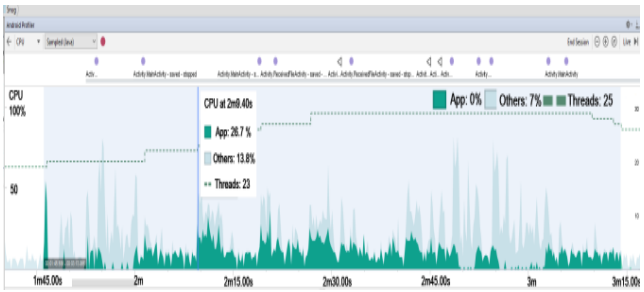

FIGURE 4- CPU PROFILING OF SNUG

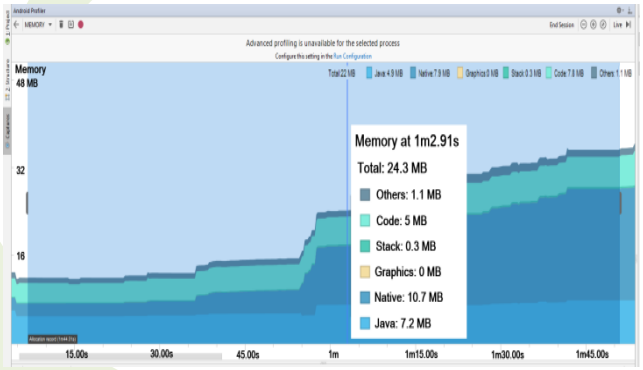

FIGURE 5-MEMORY PROFILING OF SNUG

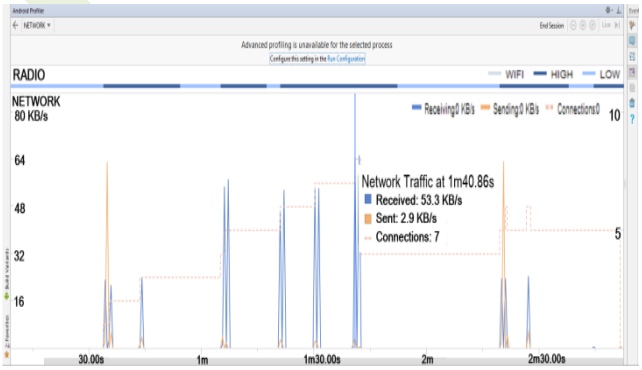

FIGURE 6- NETWORK PROFILING OF SNUG

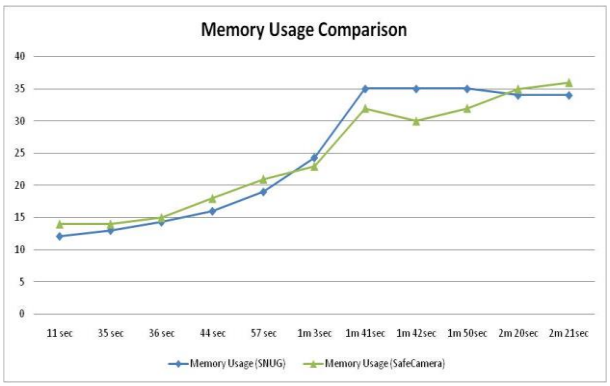

FIGURE 7- COMPARISON BETWEEN OUR APP AND SAFECAMERA

III Results after comparing both apps

After performing CPU, memory and network profiling of both apps Snug and Safe Camera, we have come to some conclusions which are mentioned below:

1. CPU Utilization

By observing the timeline of Snug and Safe Camera, we can conclude that the CPU utilization was similar in both Snug and Safe Camera.

2. Memory Utilization

The memory utilization in app Snug and Safe Camera was comparable. We have captured the memory usage of both apps at different time instances, as shown in table 5.3. and also shown in graphical form in fig. 5.22. Through this we can conclude that the memory utilization of Snug app was more stable as compared to Safe Camera app. Initially both apps were utilizing the memory in similar manner when the app was launched, then over the time period when actions were performed like login, capturing the image, receiving the acknowledgment packet, then the memory utilization of Safe Camera was higher than our app. And then the memory utilization was at peak means when sending and receiving multiple images, our app utilization was more stable as compared to Safe Camera. Hence we can conclude that our app is more stable in terms of memory utilization.

3. Network Utilization

The Safe Camera app do not provide the facility of sending and receiving the captured image over the network, hence there was no network utilization in this app, hence no peaks are shown in the timeline as shown in fig. 5.20.

4. Multi Phased Obscurity

This feature distinguishes our application from others in a way that it does not stores the multimedia images neither at the time of capturing the image nor at the time of fetching the image. While capturing the image from sender's device, our app does not save the captured image; it neither saves it in internal storage nor in external storage. Also while fetching the image at receiver side, the decrypted image is just shown at screen and is not saved in any of the storage of that device.

5. Cloud Security

By the virtue of this feature, the transmitted multimedia images by the sender users will be stored on the cloud database in an encrypted format so that not even the authorized user is able to view. This feature is implemented with a view that if cloud account is accidentally compromised, his/her confidential stored images are not leaked. These (encoded) password-protected images are decoded only once they are delivered to some recipient's phone and the decoding shall happen on the device itself during runtime by our application. Unlike the existing cloud services which store the (confidential) images as is, leaving them vulnerable to be compromised, our application largely differs from them from this security perspective.

III Benefits of Using Snug

Along with the encoded data that is communicated over the transmission channels, it also ensures the content security by obscuring the data at user's device level.

The reliable and fail-safe cloud service shall not only guarantee the integrity of the stored information but also its security because the multimedia content shall be stored in an encoded manner plus every user's data is enveloped in his/her own individual user account on the cloud.

8- CONCLUSION

TThe application can successfully capture image at the receiver end, and this image is not saved at the sender device.

The image transferred to receiver is saved at the cloud in encrypted format, also not accessible to user.

The image received at the receiver side is only accessible after entering secret key and also after image is being fetched it is not saved at receiver's device either

9- ACKNOWLEDGMENT

The preferred spelling of the word "acknowledgment" in American English is without an "e" after the "g." Use the singular heading even if you have many acknowledgments. Please put the sponsor acknowledgments in this section; do not use a footnote on the first page.

10- REFERENCES

[1] Abhishek Vichare, Tania Jose, Jagruti Tiwari, "Data security using authenticated encryption and decryption algorithm for Android phones", Computing, Communication and Automation (ICCCA) International Conference, 2017, Electronic ISBN: 978-1-5090-6471-7, Print on Demand(PoD) ISBN: 978-1-5090-6472-4, pp 789 - 794, May 2017 13

[2] Zhaohui Wang, Rahul Murmuria, Angelos Stavrou, "Implementing & Optimizing an Encryption Filesystem on Android", Mobile Data Management (MDM), 2012 IEEE 13th International Conference, Print ISBN: 978-1-4673-1796-2 Electronic ISBN: 978-0-7695-4713-8 Print ISSN: 1551-6245 Electronic ISSN: 2375-0324 pp 52-62, July 2012 14

[3] Peter Teufl, Andreas Fitzek, Daniel Hein, Alexander Marsalek, Alexander Oprisnik, Thomas Zefferer, "Android Encryption Systems", 2014 International Conference on Privacy and Security in Mobile Systems (PRISMS) Electronic, ISBN: 978-1-4799-4628-0 Print ISBN: 978-1-4799-4630-3 Print on Demand(PoD) ISBN: 978-1-4799-4627-3 pp 1 – 8, May 2014 15

[4] Kundankumar Rameshwar Saraf, Vishal Prakash Jagtap, Amit Kumar Mishra, "Text and Image Encryption Decryption Using Advanced Encryption Standard", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) ISSN 2278-6856 Volume 3, Issue 3, pp 118 126, May – June 2014 16

[5] Savithri G, K.L.Sudha, "Android Application for Secret Image Transmission and Reception Using Chaotic Steganography", International Journal of Innovative Research in Computer and Communication Engineering, ISSN (Online): 2320-9801ISSN (Print): 2320-9798, Vol. 2, Issue 7, pp 5107-5113, July 2014 17

[6] Minal G. Kumbharkhane, Dr. V. S. Gulhane, "Security Analysis of Android File System", International Journal of Advanced Research in Computer Science and Software Engineering ISSN: 2277 128X Volume 5, Issue 4, pp 148-154, April 2015 18

[7] Suchita Tayde, Asst. Prof. Seema Siledar, "File Encryption, Decryption Using AES Algorithm in Android Phone", International Journal of Advanced Research in Computer Science and Software Engineering ISSN: 2277 128X Volume 5, Issue 5, pp 550-554, May 2015 19

[8] https://en.wikipedia.org/wiki/encryption

[9]https://ico.org.uk/media/for-rganisations/guide-to-data-rotection/encryption-1-1.pdf

[9]http://books.gigatux.nl/mirror/securitytools/ddu/ch09lev1sec1.html

[10] http://www.enlume.com/mobile-data-security/

[11] https://dminc.com/blog/5-encryption-methods-for-mobile/

[12]https://backendless.com/docs/android/doc.html