Data Puzzle –
Enhanced Mobile Data Security in Cloud Storage

A Project Report
Presented to
The Faculty of the College of
Engineering

San Jose State University
In Partial Fulfillment
Of the Requirements for the Degree
Master of Science in Software Engineering
Master of Science in Computer Engineering

By
Meng-Huan Lee 011414559
Xiaoqian Ma 010769460
Zhemin Su 010832107
Ziyu Ye 011499956

[05/2017]

APPROVED

____

[Advisor’s Name], Project Advisor

ABSTRACT

Data Puzzle – Enhanced Mobile Data Security in Cloud Storage
By
Meng-Huan Lee, Xiaoqian Ma, Zhemin Su, Ziyu Ye

The number of security issues on mobile phones has increased significantly in recent years as mobile phone technology continues to develop. While mobile phones have been popularized in daily life and everyone uses mobile phones frequently, the vulnerability and threats of mobile phones also hang over our everyday use of them. (Zhang & Costa, 2016) It is noteworthy that most people prefer to back up their mobile phone data onto online cloud storage and services offered by third parties.

The cloud storage exists privacy issues. The government has the right to access personal data and personal accounts, making data stored in cloud less secure than user expectations. In the first half year of 2015, there are 17577, 12002 and 5940 requests by US law enforcement agency for accessing user data submitted to Facebook, Google, and Microsoft, respectively. And these tech giants, which own enormous number of user data, approved 66%, 78%, and 80% of the requests.

In addition to government data access request, another security threat is hacking. On December 2016, Yahoo admitted that over 1 billion user accounts were hacked, which includes username and passwords. Usernames and passwords are not good enough for protecting information stored on cloud.

Our project proposed a mechanism with an app to protect personal data on cloud storage. User can sign in with their own cloud storage account and upload their file in any format through their mobile app. This mechanism would slice the file into fragments, and

add data redundancy for recovering lost data. These data fragments will be encrypted and uploaded to the cloud storage. The outcome of the mechanism provides a secure, private and anonymous cloud storage for user data.

Acknowledgments

The authors are deeply indebted to Professor Sheng-Liang Song for her invaluable comments and assistance in the preparation of this study.

Table of Contents

List of Figures

List of Tables

Chapter 1. Project Overview

1.1 Introduction

1.1.1 Project Literature Search

The number of security issues on mobile phones has increased significantly in recent years as mobile phone technology continues to develop. While mobile phones have been popularized in daily life and everyone uses mobile phones frequently, the vulnerability and threats of mobile phones also hang over our everyday use of them. (Zhang & Costa, 2016) It is noteworthy that most people prefer to back up their mobile phone data onto online cloud storage and services offered by third parties.

The vulnerability of cloud security has been exposed frequently recently years. “It turns out that Google Drive has been incontinent, dribbling out private data courtesy of a security hole concerning files with embedded URLs.” (Vaas, 2014) Moreover, governments have right to access private data in personal accounts, making data stored in cloud even less secure than users can image. Figure 1 shows the number of requests from US law enforcement to access user data from top technical companies in the US. Figure 2 displays how many of those requests succeeded. From the two figures, we can conclude the users’ data is not entirely protected from governments. Companies may disclose user information to governments or other companies.

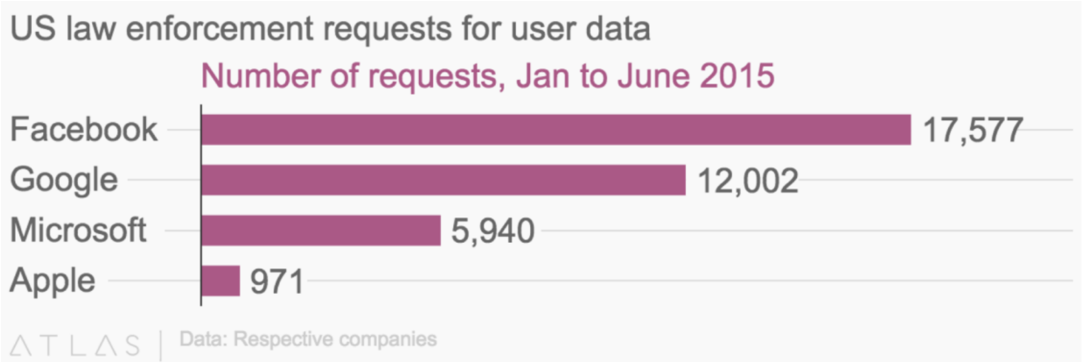

Figure 1: US Law Enforcement Requests for User Data from Jan to June 2015

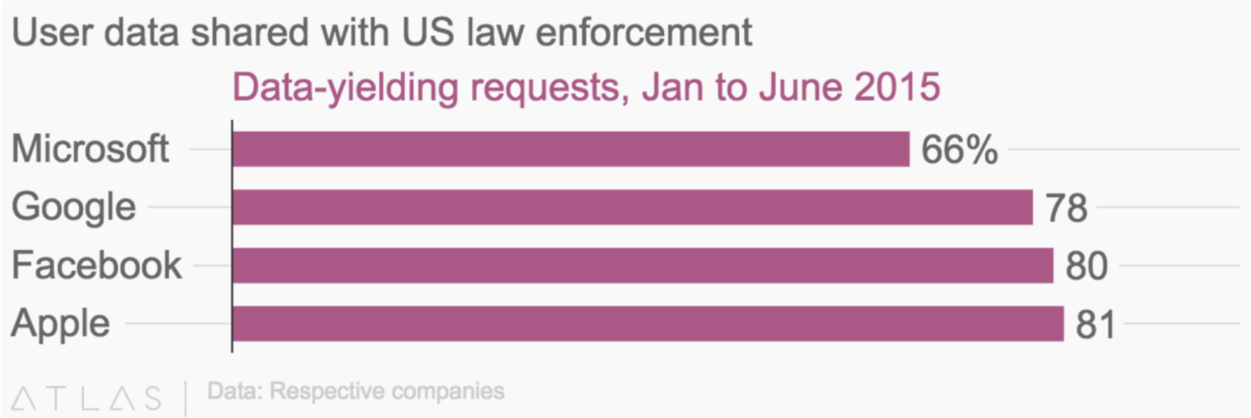

Figure 2: Data-yielding Requests from Jan to June 2015

In addition to government data access request, another security threat comes from hacking. On December 2016, Yahoo admitted that over 1 billion user accounts were hacked which including username and passwords. For protecting personal information, passwords are not enough.

For better protect the user file in the cloud, our project suggests we can encrypt and split the file before upload the file into the cloud.

1.1.2 Project Goals and Objectives

The main goal of our project is to create a file encryption App for users to keep their files safe in Google Drive. Our app assist users to protect their personal files in the more secure cloud.

To solve this problem, our file encryption app helps user to store the encrypted files in the clouds instead of uploading the unencrypted files, reducing the risk of the bugs of cloud enterprises exposed to hackers. If such situations happen unfortunately, the hackers still cannot access the encrypted files easily. As Li and Ma (2017) pointed, “the data privacy issue is paramount in cloud storage system, so the sensitive data is encrypted by the owner before outsourcing onto the cloud, and data users retrieve the interested data by encrypted search scheme.” Therefore, we need to keep data in high confidential and our app should have scalable and flexible encryption. So that advanced features can be easily added if necessary. This enables us to change or update any existing policies and keys. This helps us make a stronger encryption algorithm. What’s more, even in the case if one or more file fragments is lost, the original file can still be recovered by storing redundant data into file fragments.

1.2 Proposed Areas of Study and Academic Contribution

Due to the number of security issues of cloud storage has increased significantly in recent years, this project tends to provide users a secure solution to store their files on the clouds. This project will implement techniques such as distributed storage, symmetric encryption, QR code, error correction on mobile devices and cloud storage.

1.3 Current State of the Art

In this project, we explored different literature for project development purpose. At the beginning, “A Survey Study of Young Generations Mobile Phone Usage and Security Concerns” proved our observation that more and more people rely on mobile phone to store files on cloud drives. The article “Google Drive security hole leaks users’ files” aroused our concern for the cloud storage security. The author of article: “An Efficient Searching Technique for Mobile Cloud on Encrypted Data”, Laddha and Ragha (2016) state if data is encrypted before upload to the cloud, the user data is more secured. Then we proposed an advanced encrypted file app for google drive. For more people to be beneficial from our app, we referenced the information from the article “99.6 percent of new smartphones run Android or iOS.” We found out that Android is more popular than iOS system. We decided to develop our app for the Android system.

For efficiently manage the database, we referenced the database model from Fernández, Summers and Coleman (1975), the authors of the article “An authorization model for a shared database”. There two main choices for the database on the market. One is the NoSQL database; the other is MySQL database. For choosing the proper database, we reference the paper “A comparative study: MongoDB vs. MySQL.” whose authors are Gyorodi, Gyorodi, Pecherle and Olah (2015). Considering the result discussed in the paper and the project’s need, we decided to use the NoSQL database. And for designing an efficient data structure in the database, we referenced the designing principle, Eisner and Severance (1976), mentioned in the article “Mathematical Techniques for Efficient Record Segmentation in Large Shared Databases.” We putted our most frequent reference data as the primary data segment in the data structure. By referencing the author of firebase authorization document, we realized that Firebase has a lot of advantages including the capability of authorization for Google Account.

Chapter 2. Project Architecture

2.1 Introduction

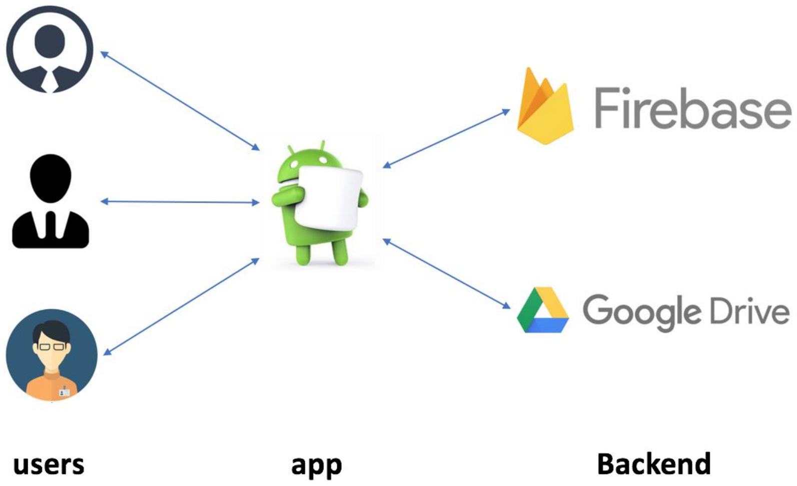

Figure 3: Project Architecture

This project has three main parts: users, app and backend. The user need to login the app using their google account or they can register a new google account on the first page of the app. After user login, the app will use user id as reference to retrieve user data from database, then return ListViews of friends, groups and files as front-end interface. If user want to fetch the encrypted file or upload unencrypted file to the google drive, user require to login their google drive account through our app to gain google drive access authentication. The process of file encryption, decryption, split and merge will be implemented in the user's own smartphone devices to avoid the potential data leaking through network transportation.

2.2 Architecture Subsystems

2.2.1 Private File Mode

Clients who use our product need to install the App in their mobile phone first. After they open the App, the App will request the user to login their Google accounts. Our App has two primary features which are uploading file in private mode and uploading file in team cooperation mode.

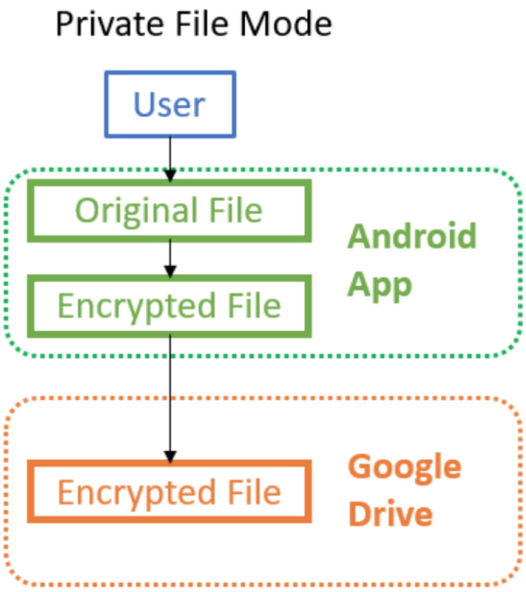

Figure 4: Mobile End File Delivery Structure in Private Mode

For the private file mode, the file uploaded by the user would be encrypted by the Android app and upload to their google drive. The detail of the encryption would be discussed later in technical aspects. The process of the private file uploading mode is shown in Figure 3.

2.2.2 Team Cooperation Mode

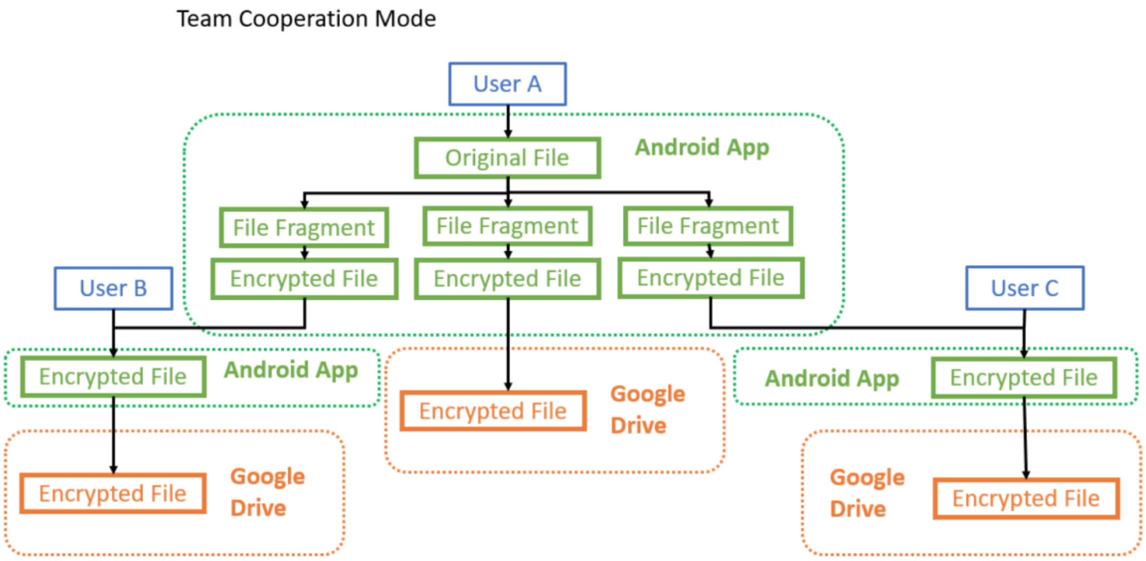

Figure 5. Mobile End File Delivery Structure in Team Cooperation Mode

As for the team cooperation file uploading mode, it provides a safer method to protect files which is shared within certain groups. The process of file sharing in team cooperation mode is shown in Figure 4. The process of how team cooperation mode works can be divided into two stages: hiding the secured files and merging them. To hide the secured files, we break the process further into 3 steps. First, the android will split the files into certain number of file fragments and then encrypt each of them. Second, the user will upload one file fragment to his/her own Google Drive. Finally, he/she will pass the rest of the file fragments to the other group members to have their phone to upload the other fragments to Google. After the final step, everyone in this group get one piece of file. In terms of merging the secured file parts, another 4 steps are involved. First, anyone in this group who wants to access this file needs to send a request to other group members. Second, after all the group members agree, our App will help user to send the file fragments to the requester. Third, the requester can merge the whole file fragments to one completely encrypted file. Fourth, the app will decrypt the file for the requester. The reason why we want to apply this schema lies in that this shared file may be important for a certain group of people, but it is not safe to let a single person to store this file. Therefore, we came up with this idea that everyone can have one piece of file since it is harmless to get one encrypted file fragment. In order to get the whole file, a user need to inform the others that he/she wants to access this file. Only after all of them approved can he/she get access to this file.

Glossary

References

Eisner, M. J., & Severance, D. G. (1976). Mathematical Techniques for Efficient Record Segmentation in Large Shared Databases. *Journal of the ACM*, 23(4), 619-635. doi:10.1145/321978.321982

Fernández, E. B., Summers, R. C., & Coleman, C. D. (1975). An authorization model for a shared data base. *Proceedings of the 1975 ACM SIGMOD international conference on Management of data - SIGMOD 75.* doi:10.1145/500081.500084

Firebase Authentication | Firebase. (n.d.). Retrieved August 10, 2017, from https://firebase.google.com/docs/auth/

Google Drive security hole leaks users' files. (2014, July 09). Retrieved August 07, 2017, from https://nakedsecurity.sophos.com/2014/07/10/Google-drive-security-hole-leaks-users-files/

Gyorodi, C., Gyorodi, R., Pecherle, G., & Olah, A. (2015). A comparative study: MongoDB vs. MySQL. *2015 13th International Conference on Engineering of Modern Electric Systems (EMES)*. doi:10.1109/emes.2015.7158433

Laddha, A., & Ragha, L. (2016). An Efficient Searching Technique for Mobile Cloud on Encrypted Data. *International Journal of Innovations in Engineering and Technology*, 7(4), 505-512.

Vincent, J. (2017, February 16). *99.6 percent of new smartphones run Android or iOS*. Retrieved from https://www.theverge.com/2017/2/16/14634656/android-ios-market-share-blackberry-2016

Zhang, S., & Costa, S. (2016). A Survey Study of Young Generations Mobile Phone Usage and Security Concerns. *2016 17th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*. doi:10.1109/pdcat.2016.075

Firebase Authentication. (2017. Nov 13) Retrieved from https://firebase.google.com/docs/auth/

Firebase Realtime Database. (2017. Nov 10) Retrieved from https://firebase.google.com/docs/database/

Shaked, U. (2015, Apr 24) End-to-end Testing with firebase-server. Retrieved from https://firebase.googleblog.com/2015/04/end-to-end-testing-with-firebase-server_16.html

Unit Testing of Cloud Functions. (2017. Oct 31) Retrieved from https://firebase.google.com/docs/functions/unit-testing

Use Firebase, Google’s unified and cross-platform SDK, to build better apps. Retrieved from https://developer.android.com/distribute/best-practices/develop/build-with-firebase.html

Appendices

Appendix A. Description of Implementation Repository